Reinforcement Learning

Prof. Volkan Cevher
volkan.cevher@epfl.ch

Lecture 3: Linear Programming

Laboratory for Information and Inference Systems (LIONS)
École Polytechnique Fédérale de Lausanne (EPFL)

EE-568 (Spring 2025)

License Information for Reinforcement Learning (EE-568)

Recall: Reinforcement learning setup

- Reinforcement Learning: Sequential decision making in an unknown environment
- Markov decision process: $M = (\mathcal{S}, \mathcal{A}, \mathsf{P}, r, \mu, \gamma)$
- Stationary stochastic policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$, $a_t \sim \pi(\cdot|s_t)$
- State-value function: $V^\pi(s) := \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) | s_0 = s, \pi\right]$
- Performance objective: $\max_\pi (1-\gamma) \sum_{s \in \mathcal{S}} \mu(s) V^\pi(s)$

Challenges:

- Infer long-term consequences based on limited, noisy short-term feedback.
- Unknown transition dynamics P: knowledge only through sampled experience.
- Large state- and action-spaces.
- Non-convex performance objective as a function of π.

Motivation

- Approximate dynamic programming (previous lecture)
  - Attempts to find approximate fixed-point solutions to the (nonlinear) Bellman equation.
  - Pros:
    - + Well-studied setting for tabular MDPs that comes with theoretical convergence guarantees.
      - See Lecture 2.
    - + Deep-learning variants (e.g., DQN [20]) are powerful.
  - Cons:
    - − Does not leverage classical machine-learning tools rooted in *convex* optimization.

Motivation (cont'd)

◦ The linear programming approach (this lecture)

- Introduces the linear programming (LP) approach, i.e., an alternative convex viewpoint.
- Overviews recent scalable algorithms with theoretical guarantees rooted in the LP approach.
- Highlights how historical key limitations have been eliminated.

Revisiting Bellman optimality equation

- We denote $V^\star(s) = \max_{\pi \in \Pi} \ V^\pi(s)$.
- $V^\star$ satisfies the Bellman optimality equation, which can be written as a feasibility problem:

$$
\begin{aligned}
&\min_{V} \ 0 \\
&\text{s.t.} \ \ V(s) = (\mathcal{T}V)(s) := \max_{a \in \mathcal{A}} \left[r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathsf{P}(s'|s,a) V(s') \right], \quad \forall\, s \in \mathcal{S}.
\end{aligned}
$$

  - $\mathcal{T}$ is the so-called Bellman operator
  - The only feasible assignment is $V^\star$
  - The above equality constraints are nonlinear in V due to the maximization over $\mathcal{A}$

Revisiting Bellman optimality equation

◦ We denote $V^\star(s) = \max_{\pi \in \Pi} \; V^\pi(s)$.

◦ $V^\star$ satisfies the Bellman optimality equation, which can be written as a feasibility problem:

$$\min_{V} \;\; 0$$

$$\text{s.t.} \;\; V(s) = (\mathcal{T}V)(s) := \max_{a \in \mathcal{A}} \left[r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathsf{P}(s'|s,a) V(s') \right], \quad \forall \; s \in \mathcal{S}.$$

- $\mathcal{T}$ is the so-called Bellman operator
- The only feasible assignment is $V^\star$
- The above equality constraints are nonlinear in V due to the maximization over $\mathcal{A}$

Remarks: ◦ The Bellman optimality operator is a γ-contraction mapping w.r.t. ℓ_∞-norm:

$$\left\| \mathcal{T}V' - \mathcal{T}V \right\|_\infty \leq \gamma \left\| V' - V \right\|_\infty.$$

◦ The Bellman operator is also monotonic (component-wise): $V' \leq V \;\Rightarrow\; \mathcal{T}V' \leq \mathcal{T}V$.

Solving MDPs via LPs: Primal LP formulation (cont'd)

Derivation:

- We will derive LP to reach the unique solution of $V^\star$.
- Recall: Bellman optimality opterator $[\mathcal{T}V](s) = \max_{a\in\mathcal{A}} \left(r(s,a) + \gamma \sum_{s'\in\mathcal{S}} \mathsf{P}(s'|s,a)V(s') \right)$.
- $V^\star$ is feasible as

$$V^\star(s) = [\mathcal{T}V^\star](s) \geq r(s,a) + \gamma \sum_{s'\in\mathcal{S}} \mathsf{P}(s'|s,a)V^\star(s'),\ \forall (s,a) \in \mathcal{S}\times\mathcal{A}.$$

- For any feasible V, we have $V \geq \mathcal{T}V$. Component-wise monotonicity ($V_1 \geq V_2 \Rightarrow \mathcal{T}V_1 \geq \mathcal{T}V_2$)

$$V \geq \mathcal{T}V \geq \mathcal{T}^2 V \geq \cdots \geq \mathcal{T}^\infty V = V^\star,$$

  implies optimality of $V^\star$.
- Uniqueness follows as $\mathcal{T}$ is contractive.

Relaxation of Bellman optimality condition

- The Bellman optimality $\Rightarrow V^\star$ is the function with the lowest values $V(s)$ among all $V \in \mathbb{R}^{|\mathcal{S}|}$ satisfying

$$V(s) \geq \; r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathsf{P}(s'|s,a)V(s'), \quad \forall \; s \in \mathcal{S}, \; a \in \mathcal{A}. \qquad \text{(Bellman inequality)}$$

- Note that the Bellman inequality constraint is linear in V $\quad \Longrightarrow \quad$ Linear Programming (LP)

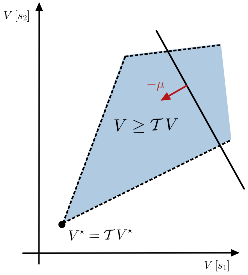


Figure: Graphical interpretation of Bellman inequality

Solving MDPs via LPs: Primal LP formulation

◦ The previous derivation motivates the following LP.

Primal LP

Let $\mu(s) > 0, s \in \mathcal{S}$ be the initial distribution (or any positive weights). Then, the primal LP is given by

$$\begin{aligned} \min_{V} \quad & (1-\gamma) \sum_{s \in \mathcal{S}} \mu(s) V(s) \\ \text{s.t.} \quad & V(s) \geq \; r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathsf{P}(s'|s,a) V(s'), \quad \forall\, s \in \mathcal{S},\; a \in \mathcal{A}. \end{aligned} \tag{P}$$

Remarks:

- ◦ The number of decision variables is $|\mathcal{S}|$, and the number of constraints is $|\mathcal{S}||\mathcal{A}|$.
- ◦ Given $V^\star$, we can determine an optimal (deterministic) policy greedily

$$\pi^\star(s) \in \arg\max_{a \in \mathcal{A}} \left[r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathsf{P}(s'|s,a) V^\star(s) \right]. \tag{1}$$

- ◦ The factor $(1-\gamma)$ in the objective will ensure that the dual variables are in the simplex.

Solving MDPs via LPs: Primal LP formulation (cont'd)

Recall: Primal LP

Let $\mu(s) > 0, s \in \mathcal{S}$ be the initial distribution (or any positive weights). The primal LP formulation is given by

$$\begin{aligned} \min_{V} \quad & (1-\gamma)\sum_{s\in\mathcal{S}} \mu(s)V(s) \\ \text{s.t.} \quad & V(s) \geq r(s,a) + \gamma \sum_{s'\in\mathcal{S}} \mathsf{P}(s'|s,a)V(s'), \quad \forall\, s \in \mathcal{S},\ a \in \mathcal{A}. \end{aligned} \tag{P}$$

Lemma (LP Formulation and $V^\star$)

$V^\star$ is the unique optimal solution to the above LP formulation for any positive weights $\{\mu(s)\}$.

Remark:

- The unique optimizer does not depend on the positive weights $\{\mu(s)\}$.

Solving MDPs via LPs: Dual LP formulation

- From linear programming, we know that the dual LP of (P) is given by the following.
  - See supplementary material, Slide 8. We refer to [19] for a comprehensive treatment.

Dual LP

$$
\begin{aligned}
\max_{\lambda} \quad & \sum_{s\in\mathcal{S}} \sum_{a\in\mathcal{A}} r(s,a)\lambda(s,a) \\
\text{s.t.} \quad & \sum_{a\in\mathcal{A}} \lambda(s,a) = (1-\gamma)\mu(s) + \gamma \sum_{s'\in\mathcal{S},a'\in\mathcal{A}} \mathsf{P}(s|s',a')\lambda(s',a'), \quad \forall\, s\in\mathcal{S}, \\
& \lambda(s,a) \geq 0, \quad \forall\, s\in\mathcal{S}, a\in\mathcal{A}.
\end{aligned} \tag{D}
$$

Remarks:

- The number of decision variables is given by $|\mathcal{S}||\mathcal{A}|$.
- The number of constraints is given by $|\mathcal{S}| + |\mathcal{S}||\mathcal{A}|$.
- The constraints imply the decision variables are probabilities: $\lambda \in \Delta([|\mathcal{S}||\mathcal{A}|])$.
- The solution to the dual LP $\lambda^\star$ corresponds to the state-action *occupancy* of $\pi^\star$.

Occupancy measure

Definition (Occupancy measure)

The occupancy measure for an initial distribution μ and a policy π is defined as follows:

$$\lambda_\mu^\pi(s,a) = (1-\gamma)\sum_{t=0}^{\infty}\gamma^t \mathbb{P}[s_t = s, a_t = a \mid s_0 \sim \mu,\ \pi],$$

where $\mathbb{P}[\cdot \mid s_0 \sim \mu,\ \pi]$ denotes the probability of an event when following policy π starting from $s_0 \sim \mu$.

Interpretation:

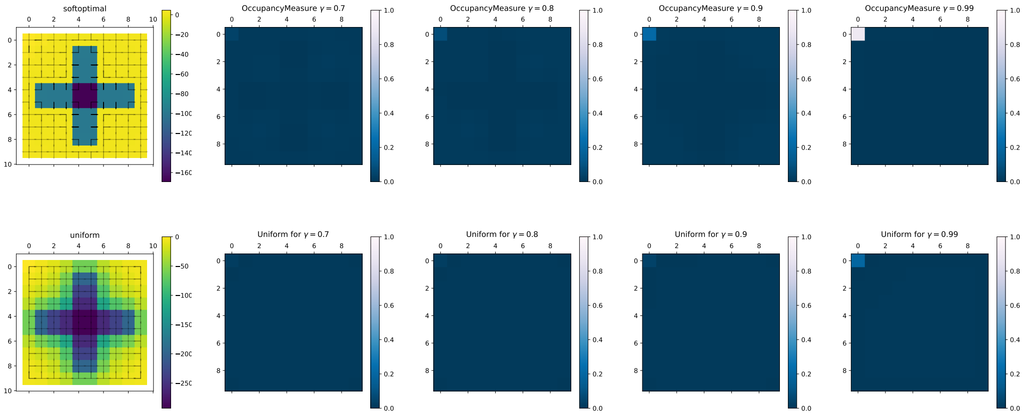
- $\lambda_\mu^\pi(s,a)$ is the normalized discounted visitation frequency of the pair (s,a) when π is played:

$$\lambda_\mu^\pi(s,a) = (1-\gamma)\mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^t \mathbb{1}(s_t = s, a_t = a)\ \middle|\ s_0 \sim \mu,\ \pi\right]$$

- We sometimes drop the subscript μ after specifying a fixed initial distribution.

Visualize an occupancy measure

- Let us consider the policies represented by the arrows in the leftmost column.
- The corresponding occupancy measures varying the discounted factor are depicted just below.
- Notice that increasing γ makes the effect of the initial distribution less and less prominent.



A closer look at the dual LP

- For any policy π and $s_0 \sim \mu$, we defined the occupancy measure $\lambda^\pi(s,a)$ as

$$\lambda^\pi(s,a) := (1-\gamma)\sum_{t=0}^{\infty}\gamma^t \mathbb{P}[s_t = s, a_t = a \,|\, s_0 \sim \mu, \pi].$$

- We can write

$$\begin{aligned}
&(1-\gamma)\mathbb{E}_{s\sim\mu}[V^\pi(s)] && \Rightarrow \text{ primal objective (P)} \\
&= (1-\gamma)\,\mathbb{E}\left[\sum\nolimits_{t=0}^{\infty}\gamma^t r(s_t,a_t) \mid s_0 \sim \mu,\ \pi\right] \\
&= (1-\gamma)\,\mathbb{E}\left[\sum\nolimits_{t=0}^{\infty}\sum_{s\in\mathcal{S},a\in\mathcal{A}}\gamma^t \mathbb{1}(s_t = s, a_t = a) r(s,a) \mid s_0 \sim \mu,\ \pi\right] \\
&= (1-\gamma)\sum_{s\in\mathcal{S},a\in\mathcal{A}}\sum_{t=0}^{\infty}\gamma^t \mathbb{P}[s_t = s, a_t = a \,|\, s_0 \sim \mu,\ \pi]\, r(s,a) \\
&= \sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}}\lambda^\pi(s,a) r(s,a) && \Rightarrow \text{ dual objective (D)}
\end{aligned}$$

A closer look at the dual LP (cont'd)

Recall: Dual LP

$$\begin{aligned}
\max_{\lambda} \quad & \sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r(s,a)\lambda(s,a) \\
\text{s.t.} \quad & \sum_{a\in\mathcal{A}} \lambda(s,a) = (1-\gamma)\mu(s) + \gamma \sum_{s'\in\mathcal{S},a'\in\mathcal{A}} \mathsf{P}(s|s',a')\lambda(s',a'), \quad \forall\, s\in\mathcal{S}, \\
& \lambda(s,a) \geq 0, \quad \forall\, s\in\mathcal{S}, a\in\mathcal{A}.
\end{aligned} \tag{D}$$

Observations:

- The occupancy measure $\lambda^{\pi}(s,a)$ satisfies the constraints in the dual LP.
- By the Markov property, we have (see the supplementary material, Slide 14 for details)

$$\lambda^{\pi}(s,a) = (1-\gamma)\mu(s)\pi(a|s) + \gamma \sum_{s',a'} \pi(a|s)\mathsf{P}(s|s',a')\lambda^{\pi}(s',a').$$

- Summing over a implies feasibility.

A closer look at the dual LP (cont'd)

Recall: Dual LP

$$\begin{aligned} \max_{\lambda} \quad & \sum_{s\in\mathcal{S}}\sum_{a\in\mathcal{A}} r(s,a)\lambda(s,a) \\ \text{s.t.} \quad & \sum_{a\in\mathcal{A}} \lambda(s,a) = (1-\gamma)\mu(s) + \gamma \sum_{s'\in\mathcal{S},a'\in\mathcal{A}} \mathsf{P}(s|s',a')\lambda(s',a'), \quad \forall\, s\in\mathcal{S}, \\ & \lambda(s,a) \geq 0, \quad \forall\, s\in\mathcal{S}, a\in\mathcal{A}. \end{aligned} \tag{D}$$

Observations:

- For any λ feasible to the dual LP, we can define a policy

$$\pi_\lambda(a\,|\,s) = \frac{\lambda(s,a)}{\sum_{a\in\mathcal{A}}\lambda(s,a)},$$

  where we set $\pi_\lambda(\cdot|s)$ arbitrarily when $\sum_{a\in\mathcal{A}}\lambda(s,a) = 0$. Then, $\lambda^{\pi_\lambda} = \lambda$.

- Note that λ is optimal for (D) iff π_λ is an optimal policy [30]. (self-study)

- Optimality of policies does not depend on μ. (LP sensitivity analysis)

Finding the optimal policy

- Primal LP approach:
  - Solve primal LP to obtain for the optimal value function $V^\star$
  - Then construct an optimal policy (deterministic) as the greedy policy

$$\pi^\star(s) \in \arg\max_{a \in \mathcal{A}} \left[r(s,a) + \gamma \sum\nolimits_{s' \in \mathcal{S}} \mathsf{P}(s'|s,a) V^\star(s') \right].$$

- Dual LP approach:
  - Solve the dual LP to obtain an optimal state-action occupancy $\lambda^\star$
  - Then construct the optimal policy (randomized) by

$$\pi^\star(a \mid s) = \frac{\lambda^\star(s,a)}{\sum_{a \in \mathcal{A}} \lambda^\star(s,a)}.$$

- For further reading: See [30] (Section 6.9)

Occupancy measure and value function

Pop quiz: ◦ What is the relation between the occupancy measure and the value function?

Occupancy measure and value function

Pop quiz: ◦ What is the relation between the occupancy measure and the value function?

Answer:

$$(1-\gamma)V^{\pi}(\mu) = \langle \lambda_{\mu}^{\pi}, r \rangle.$$

Occupancy measure and value function

Pop quiz: ◦ What is the relation between the occupancy measure and the value function?

Answer:

$$(1-\gamma)V^\pi(\mu) = \langle \lambda_\mu^\pi, r \rangle.$$

Remark: ◦ It holds that

$$V^\pi(\mu) = \langle \mu, V^\pi \rangle = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \mid s_0 \sim \mu,\ \pi\right].$$

Occupancy measure and value function (cont'd)

Derivation:

$$
\begin{aligned}
V^{\pi}(\mu) &= \mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^t r(s_t,a_t) \mid s_0\sim\mu,\ \pi\right] \\
&= \mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^t \sum_{s,a} r(s,a)\mathbb{1}(s_t=s, a_t=a) \mid s_0\sim\mu,\ \pi\right]
\end{aligned}
$$

Occupancy measure and value function (cont'd)

Derivation:

$$
\begin{aligned}
V^\pi(\mu) &= \mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^t r(s_t,a_t) \mid s_0\sim\mu,\ \pi\right] \\
&= \mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^t \sum_{s,a} r(s,a)\mathbb{1}(s_t=s,a_t=a) \mid s_0\sim\mu,\ \pi\right] \\
&= \sum_{s,a} r(s,a)\ \mathbb{E}\left[\sum_{t=0}^{\infty}\gamma^t \mathbb{1}(s_t=s,a_t=a) \mid s_0\sim\mu,\ \pi\right] \qquad \text{(Linearity of expectation)}
\end{aligned}
$$

Occupancy measure and value function (cont'd)

Derivation:

$$\begin{aligned}
V^\pi(\mu) &= \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \mid s_0 \sim \mu,\ \pi\right] \\
&= \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \sum_{s,a} r(s,a) \mathbb{1}(s_t = s, a_t = a) \mid s_0 \sim \mu,\ \pi\right] \\
&= \sum_{s,a} r(s,a)\ \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \mathbb{1}(s_t = s, a_t = a) \mid s_0 \sim \mu,\ \pi\right] && \text{(Linearity of expectation)} \\
&= \sum_{s,a} r(s,a) \sum_{t=0}^{\infty} \gamma^t\ \mathbb{P}[s_t = s, a_t = a \mid s_0 \sim \mu,\ \pi] && \text{(Dominated convergence theorem)}
\end{aligned}$$

◦ For more details on the dominated convergence theorem, see Slide 11 in the supplementary material.

Occupancy measure and value function (cont'd)

Derivation:

$$\begin{aligned}
V^{\pi}(\mu) &= \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \mid s_0 \sim \mu,\ \pi\right] \\
&= \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \sum_{s,a} r(s,a) \mathbb{1}(s_t = s, a_t = a) \mid s_0 \sim \mu,\ \pi\right] \\
&= \sum_{s,a} r(s,a)\ \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \mathbb{1}(s_t = s, a_t = a) \mid s_0 \sim \mu,\ \pi\right] && \text{(Linearity of expectation)} \\
&= \sum_{s,a} r(s,a) \sum_{t=0}^{\infty} \gamma^t\ \mathbb{P}[s_t = s, a_t = a \mid s_0 \sim \mu,\ \pi] && \text{(Dominated convergence theorem)} \\
&= \frac{\sum_{s,a} r(s,a) \lambda_{\mu}^{\pi}(s,a)}{1-\gamma} = \frac{\langle \lambda_{\mu}^{\pi}, r\rangle}{1-\gamma}. \quad \square
\end{aligned}$$

◦ For more details on the dominated convergence theorem, see Slide 11 in the supplementary material.

Some more compact notation

- With the following definitions, we can compactly write the primal and dual LP in matrix form.
- We will use the following matrix notation.
  - Write the transitions P in matrix form, i.e., P is a $(|S||A| \times |S|)$-matrix and the entry in row $(s,a) \in \mathcal{S} \times \mathcal{A}$ and column $s' \in \mathcal{S}$ is given by

$$P_{(s,a),s'} \triangleq \mathsf{P}(s'|s,a).$$

  - E is a binary matrix of dimensions $|S||A| \times |S|$, defined by

$$E_{(s,a),s'} \triangleq \begin{cases} 1 & (\text{if } s = s'), \\ 0 & (\text{else}). \end{cases}$$

  - Write $r, \ \lambda \in \mathbb{R}^{|S||A|}$ for the (column) vectors with entries $r(s,a)$, $\lambda(s,a)$ at index $(s,a) \in \mathcal{S} \times \mathcal{A}$, respectively.
  - Write $\mu, \ V \in \mathbb{R}^{S}$ for the vectors with entries $\mu(s), \ V(s)$ at index $s \in \mathcal{S}$, respectively.

Some more compact notation - Visualization

- To simplify the notation, recall the matrices defined on slide 20:
  - $E \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|\times|\mathcal{S}|}$ such that $(EV)(s,a) = V(s)$ (copying $|\mathcal{A}|$ times),
  - $P \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|\times|\mathcal{S}|}$ such that $(PV)(s,a) = \sum_{s'} \mathsf{P}(s'|s,a)V(s')$ (expectation over $s'|s,a$).
  - E is a block matrix, with the $|\mathcal{S}| \times |\mathcal{S}|$ identity matrix vertically stacked $|\mathcal{A}|$ times:

$$E = \begin{bmatrix} I_{|\mathcal{S}|} \\ \vdots \\ I_{|\mathcal{S}|} \end{bmatrix}.$$

  - P is a block matrix, with the $(|\mathcal{S}| \times |\mathcal{S}|)$-matrices P_{a_i}

$$P_{a_i} = \begin{pmatrix} \mathsf{P}(s_1|s_1,a_i) & \cdots & \mathsf{P}(s_{|\mathcal{S}|}|s_1,a_i) \\ \vdots & & \vdots \\ \mathsf{P}(s_1|s_{|\mathcal{S}|},a_i) & \cdots & \mathsf{P}(s_{|\mathcal{S}|}|s_{|\mathcal{S}|},a_i) \end{pmatrix},$$

vertically stacked for $i = 1, \ldots, |\mathcal{A}|$:

$$P = \begin{bmatrix} P_{a_1} \\ \vdots \\ P_{a_{|\mathcal{A}|}} \end{bmatrix}.$$

Some more compact notation - Visualization (cont'd)

- Their adjoints are given by
  - $E^T \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{S}||\mathcal{A}|}$ such that $(E^T\lambda)(s) = \sum_a \lambda(s,a)$ (sum over all a),
  - $P^T \in \mathbb{R}^{|\mathcal{S}|\times|\mathcal{S}||\mathcal{A}|}$ such that $(P^T\lambda)(s') = \sum_{s,a} \mathsf{P}(s'|s,a)\lambda(s,a)$ (total expectation for s' w.r.t. λ).

  - E^T is a block matrix, with the $|\mathcal{S}| \times |\mathcal{S}|$ identity matrix horizontally stacked $|\mathcal{A}|$ times:

$$E^T = \begin{bmatrix} I_{|\mathcal{S}|} & \cdots & I_{|\mathcal{S}|} \end{bmatrix}.$$

  - P^T is a block matrix, with the $(|\mathcal{S}| \times |\mathcal{S}|)$-matrices $P^T_{a_i}$

$$P^T_{a_i} = \begin{pmatrix} \mathsf{P}(s_1|s_1,a_i) & \cdots & \mathsf{P}(s_1|s_{|\mathcal{S}|},a_i) \\ \vdots & & \vdots \\ \mathsf{P}(s_{|\mathcal{S}|}|s_1,a_i) & \cdots & \mathsf{P}(s_{|\mathcal{S}|}|s_{|\mathcal{S}|},a_i) \end{pmatrix},$$

horizontally stacked for $i = 1, \ldots, |\mathcal{A}|$:

$$P^T = \begin{bmatrix} P^T_{a_1} & \cdots & P^T_{a_{|\mathcal{A}|}} \end{bmatrix}.$$

Linear Programming - Summary

Primal LP:

$$\min_{V\in\mathbb{R}^{|\mathcal{S}|}} \quad (1-\gamma)\langle \mu, V\rangle \qquad \text{s.t.} \quad EV \geq r + \gamma P V. \tag{P}$$

- Primal LP over value functions
- $|\mathcal{S}|$ decision variables and $|\mathcal{S}||\mathcal{A}|$ constraints
- $\forall\ V$ primal feasible $\quad \Rightarrow\ V^\star \leq V$
- Optimal value function $V^\star$ is the optimizer
- Optimal policy is the associated greedy policy

Dual LP

$$\max_{\lambda\in\mathbb{R}^{|\mathcal{S}||\mathcal{A}|}} \quad \langle \lambda, r\rangle \qquad \text{s.t.} \quad E^\top\lambda = (1-\gamma)\mu + \gamma P^\top \lambda, \quad \lambda \geq 0. \tag{D}$$

- Dual LP over occupancy measures
- $|\mathcal{S}||\mathcal{A}|$ variables and $|\mathcal{S}| + |\mathcal{S}||\mathcal{A}|$ constraints
- $\forall$ policy π, the induced λ^π is dual feasible
- $\forall$ feasible $\lambda \Rightarrow \pi_\lambda$ has occupancy measure λ
- Optimal policy is the associated random policy $\pi_{\lambda^\star}$

Dynamic programming vs linear programming (exact solutions)

Algorithm	Component	Output
Value Iteration (VI)	Bellman Optimality Operator $\mathcal{T}$	$V^\star$ (control)
Policy Iteration (PI)	(Multiple) Bellman Operator $\mathcal{T}^\pi$ + Greedy Policy	$\pi^\star$ (control)
Linear Programming (LP)	LP solver (Simplex, Interior Point Method)	$V^\star, \pi^\star$ (control)

Dynamic Programming:

- Simple iterative updates.
- Polynomial complexity in $|\mathcal{S}|$ and $|\mathcal{A}|$ and $(1-\gamma)^{-1}$.
- Works better for short horizon problems.

Linear Programming:

- Rich library of fast LP solvers.
- Polynomial complexity in $|\mathcal{S}|$ and $|\mathcal{A}|$ **but not** on $(1-\gamma)^{-1}$.
- Works better for long horizon problems.

The LP approach - Pros and Cons

◦ Why is this useful?

- Defining optimality is simple: no value functions, no fixed-point equations, just the numerical objective.
- Easily comprehensible with an optimization background.
- A disciplined convex optimization template with a rich set of algorithms.

◦ End User License Agreement:

- Number of variables is large.
- Intractable number of constraints.
- Constraints may not be satisfied when working with function approximators.

Beyond exact solutions - A bit of history of approximate linear programming (ALP)

- [Manne 1960] [18]
  - Formulated the primal LP over value functions and showed equivalence to Bellman equations.
- [Borkar 1988] [3] and [Hérnandez-Lerma & Lasserre 1996, 1999] [10, 11]
  - Studied the LP approach to MDPs with continuous state and action spaces.
  - The corresponding LPs are infinite-dimensional.
- [Schweitzer & Seidman 1982] [34]
  - Proposed linear function approximators to reduce the number of decision variables
  - Proposed a relaxation to reduce the number of constraints.
- [De Farias & Van Roy 2003, 2004] [6, 7]
  - Analyzed the reduction [Schweitzer & Seidman 1982] [34].
  - Inspired some follow-up work in RL [Petrik et al. 2009,2010] [28, 27], [Desai et al. 2012] [8], [Abbasi-Yadkori et al. 2014] [1], [Lakshminarayanan et al. 2018] [16].
- We refer to Slide 36 in the supplementary material for more details.

Towards the Lagrangian

- Instead of working solely with the primal or dual LP formulation, we work with an expression combining them.
- Introducing the Lagrangian multipliers vector $\lambda \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$, we can write the Lagrangian as follows:

Primal LP:

$$\begin{aligned} \min_{V \in \mathbb{R}^{|\mathcal{S}|}} \quad & (1-\gamma)\langle \mu, V \rangle \\ \text{s.t.} \quad & EV \geq r + \gamma P V. \end{aligned} \tag{P}$$

Dual LP

$$\begin{aligned} \max_{\lambda \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}} \quad & \langle \lambda, r \rangle \\ \text{s.t.} \quad & E^\top \lambda = (1-\gamma)\mu + \gamma P^\top \lambda, \quad \lambda \geq 0. \end{aligned} \tag{D}$$

$$\Updownarrow$$

Saddle point formulation

$$\min_{V} \max_{\lambda \geq 0} (1-\gamma)\langle \mu , V \rangle + \langle \lambda , r + \gamma P V - E V \rangle. \tag{Saddle-point problem}$$

Minimax optimization

∘ We recap some minimax optimization background in preparation for the so-called REPS algorithm.

Bilinear min-max template

$$\min_{\mathbf{x}\in\mathcal{X}} \max_{\mathbf{y}\in\mathcal{Y}} f(\mathbf{x}) + \langle \mathbf{A}\mathbf{x}, \mathbf{y}\rangle - h(\mathbf{y}),$$

where $\mathcal{X} \subseteq R^p$ and $\mathcal{Y} \subseteq \mathbb{R}^n$.

- $f : \mathcal{X} \to \mathbb{R}$ is convex.
- $h : \mathcal{Y} \to \mathbb{R}$ is convex.

Convex-concave min-max template

$$\min_{\mathbf{x}\in\mathcal{X}} \max_{\mathbf{y}\in\mathcal{Y}} \Phi(\mathbf{x}, \mathbf{y}), \tag{2}$$

where $\Phi(\mathbf{x}, \mathbf{y})$ is convex in $\mathbf{x}$ and concave in $\mathbf{y}$.

Basic algorithms for minimax

◦ Given $\min_{\mathbf{x}\in\mathcal{X}} \max_{\mathbf{y}\in\mathcal{Y}} \Phi(\mathbf{x},\mathbf{y})$, define $V(\mathbf{z}) = [\nabla_{\mathbf{x}}\Phi(\mathbf{x},\mathbf{y}), -\nabla_{\mathbf{y}}\Phi(\mathbf{x},\mathbf{y})]$ with $\mathbf{z} = [\mathbf{x},\mathbf{y}]$.

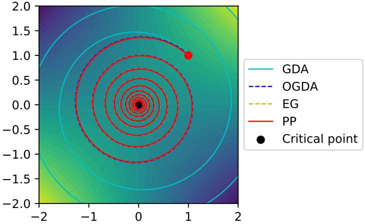


Figure: Trajectory of different algorithms for a simple bilinear game $\min_x \max_y xy$.

◦ (In)Famous algorithms
- Gradient Descent Ascent (GDA)
- Proximal point method (PPM) [33]
- Extra-gradient (EG) [15]
- Optimistic Gradient Descent Ascent (OGDA) [21]
- Reflected-Forward-Backward-Splitting (RFBS) [4]

◦ EG and OGDA are approximations of the PPM
- $\mathbf{z}^{k+1} = \mathbf{z}^k - \eta V(\mathbf{z}^k)$.
- $\mathbf{z}^{k+1} = \mathbf{z}^k - \eta V(\mathbf{z}^{k+1})$.
- $\mathbf{z}^{k+1} = \mathbf{z}^k - \eta V(\mathbf{z}^k - \alpha V(\mathbf{z}^{k-1}))$
- $\mathbf{z}^{k+1} = \mathbf{z}^k - \eta[2V(\mathbf{z}^k) - V(\mathbf{z}^{k-1})]$
- $\mathbf{z}^{k+1} = \mathbf{z}^k - \eta V(2\mathbf{z}^k - \mathbf{z}^{k-1})$

Proximal point method (PPM)

◦ Consider the following smooth unconstrained optimization problem: $\min_{\mathbf{x}\in\mathbb{R}^p} f(\mathbf{x})$

Proximal point method for convex minimization.

For a step-size $\tau > 0$, PPM can be written as follows

$$\boxed{\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}\in\mathbb{R}^p}\left\{f(\mathbf{x}) + \frac{1}{2\tau}\|\mathbf{x}-\mathbf{x}^k\|^2\right\} := \mathrm{prox}_{\tau f}(\mathbf{x}^k)} \tag{3}$$

Observations: ◦ The optimality condition of (3) reveals a simpler PPM recursion for smooth f:

$$\boxed{\mathbf{x}^{k+1} = \mathbf{x}^k - \tau\nabla f(\mathbf{x}^{k+1}).}$$

◦ PPM is an **implicit**, non-practical algorithm since we need the point $\mathbf{x}^{k+1}$ for its update.

◦ Each step of PPM can be as hard as solving the original problem.

◦ Convergence properties are well understood due to Rockafellar [33].

PPM and minimax optimization

PPM applied to the minimax template: $\min_{\mathbf{x}\in\mathbb{R}^d} \max_{\mathbf{y}\in\mathbb{R}^n} \Phi(\mathbf{x},\mathbf{y})$

Define $\mathbf{z} = [\mathbf{x},\mathbf{y}]^\top$ and $\mathbf{V}(\mathbf{z}) = [\nabla_{\mathbf{x}}\Phi(\mathbf{x},\mathbf{y}), -\nabla_{\mathbf{y}}\Phi(\mathbf{x},\mathbf{y})]^\top$. PPM iterations with a step-size $\tau > 0$ is given by

$$\mathbf{z}^{k+1} = \mathbf{z}^k - \tau\mathbf{V}(\mathbf{z}^{k+1}).$$

Derivation:

- For $\tau > 0$, $(\mathbf{x}^{k+1},\mathbf{y}^{k+1})$ is the unique solution to the saddle point problem,

$$\boxed{\min_{\mathbf{x}\in\mathbb{R}^d} \max_{\mathbf{y}\in\mathbb{R}^n} \Phi(\mathbf{x},\mathbf{y}) + \frac{1}{2\tau}\|\mathbf{x}-\mathbf{x}^k\|^2 - \frac{1}{2\tau}\|\mathbf{y}-\mathbf{y}^k\|^2} \tag{4}$$

- Writing the optimality condition of the update in (4)

$$\boxed{\mathbf{x}^{k+1} = \mathbf{x}^k - \tau\nabla_{\mathbf{x}}\Phi(\mathbf{x}^{k+1},\mathbf{y}^{k+1}), \qquad \mathbf{y}^{k+1} = \mathbf{y}^k + \tau\nabla_{\mathbf{y}}\Phi(\mathbf{x}^{k+1},\mathbf{y}^{k+1})} \tag{5}$$

Observation:

- **PPM is an implicit algorithm.**
- For the bilinear problem, PPM is implementable!

Proximal point methods in the Bregman setup

Definition: Bregman distance

Let $\omega : \mathcal{X} \to \mathbb{R}$ be a distance generating function where ω is $1-$strongly convex w.r.t. some norm $\|\cdot\|$ on the underlying space and is continuously differentiable. The Bregman distance induced by $\omega(\cdot)$ is given by

$$D_\omega(\mathbf{z}, \mathbf{z}') = \omega(\mathbf{z}) - \omega(\mathbf{z}') - \nabla\omega(\mathbf{z}')^\top(\mathbf{z} - \mathbf{z}').$$

- The proximal point method in the Bregman setup reads as follows:

$$\boxed{\mathbf{x}^{k+1} = \arg\min_{\mathbf{x}\in\mathbb{R}^p} \left\{ f(\mathbf{x}) + \frac{1}{\tau} D_\omega(\mathbf{x}, \mathbf{x}^k) \right\}}$$

Remarks:

- Choosing the negative entropy as a generating function $\omega(\mathbf{x}) = \langle \mathbf{x}, \log \mathbf{x} \rangle$, we obtain the KL divergence. Such $\omega(\mathbf{x})$ is 1-strongly convex in $\|\cdot\|_1$ norm.
- This choice will allow to avoid projection in the simplex constraints and it improves the dependence on the domain dimension.
- Now, we will see PPM in action on the Lagrangian.

Detour: Primal-dual π-learning given the model

Saddle point formulation

$$\min_{V} \max_{\lambda \in \Delta_{\mathcal{S}\times\mathcal{A}}} (1-\gamma)\langle \mu \,, V\rangle + \langle \lambda \,, r + \gamma PV - EV\rangle. \qquad \text{(Saddle-point problem)}$$

◦ For known dynamics, it can be solved via primal-dual gradient updates:

- $V_{k+1} = V_k - \eta\Big((\gamma P - E)^\top \lambda_k + (1-\gamma)\mu\Big).$
- $\lambda_{k+1} \propto \lambda_k \odot e^{\eta(r+\gamma PV_k - EV_k)}$, where $\odot$ denotes entry wise multiplication.

Detour: Primal-dual π-learning given the model

Saddle point formulation

$$\min_{V} \max_{\lambda \in \Delta_{\mathcal{S}\times\mathcal{A}}} (1-\gamma)\langle \mu \,, V\rangle + \langle \lambda \,, r + \gamma PV - EV\rangle. \qquad \text{(Saddle-point problem)}$$

- For known dynamics, it can be solved via primal-dual gradient updates:
  - $V_{k+1} = V_k - \eta\Big((\gamma P - E)^\top \lambda_k + (1-\gamma)\mu\Big)$.
  - $\lambda_{k+1} \propto \lambda_k \odot e^{\eta(r+\gamma PV_k - EV_k)}$, where $\odot$ denotes entry wise multiplication.
- The second update is known as *mirror descent* over the simplex (see 22 for details). It is defined by

$$\lambda_{k+1} := \arg\max_{\lambda \in \Delta_{\mathcal{S}\times\mathcal{A}}} \left(\langle \lambda, r + \gamma PV_k - EV_k\rangle - \frac{1}{\eta}\mathsf{KL}\left(\lambda || \lambda_k\right)\right),$$

where $\mathsf{KL}\left(p||q\right) = \sum_i p_i \log\left(\frac{p_i}{q_i}\right)$ is the Kullback-Leibler divergence.

- The mirror descent update can be explicitely written as

$$\lambda_{k+1}(s,a) = \frac{\lambda_k(s,a)\exp(\eta[r+\gamma PV_k - EV_k](s,a))}{\sum_{s',a'}\lambda_k(s',a')\exp(\eta[r+\gamma PV_k - EV_k](s',a'))}.$$

Detour: Primal-dual π-learning given the model

Saddle point formulation

$$\min_{V} \max_{\lambda \in \Delta_{\mathcal{S}\times\mathcal{A}}} (1-\gamma)\langle \mu\,, V\rangle + \langle \lambda\,, r + \gamma PV - EV\rangle. \qquad \text{(Saddle-point problem)}$$

- For known dynamics, it can be solved via primal-dual gradient updates:
  - $V_{k+1} = V_k - \eta\Big((\gamma P - E)^\intercal \lambda_k + (1-\gamma)\mu\Big)$.
  - $\lambda_{k+1} \propto \lambda_k \odot e^{\eta(r+\gamma PV_k - EV_k)}$, where $\odot$ denotes entry wise multiplication.
- Gradients are expectations under the occupancy measure iterates λ_k and the transition law P

$\Rightarrow$ efficient stochastic implementation [Chen et al. 2018] [5], [Jin & Sidford. 2018] [12].

  - State-of-the-art sample complexity for solving small MDPs.
  - $\mathcal{O}\left(\frac{|\mathcal{S}||\mathcal{A}|\log(\frac{1}{\delta})}{(1-\gamma)^4\varepsilon^2}\right)$ samples for finding an ε-optimal policy with probability at least $1-\delta$.

REPS: A success story

- REPS is widely popular in the robotics community.
- It applies proximal point to the Dual LP.
- A robot trained with REPS manages to play table tennis.

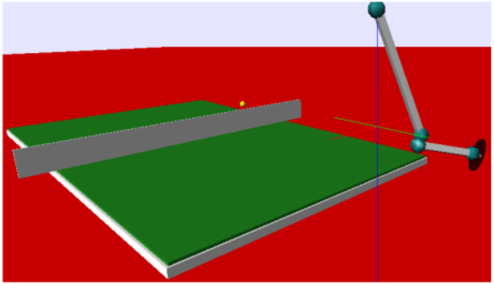


Figure: Source: Relative Entropy Policy Search [26]

Towards REPS: Proximal point on the dual LP

- Recall: Proximal point is generally an implicit method.
- However, for a linear objective PPM can be implemented.
- Hence, we can apply proximal point updates on the Lagrangian, which is just a bilinear form.

Recall: Dual LP

$$\begin{aligned}\lambda_k &= \operatorname{argmax}_{\lambda \in \Delta} \langle \lambda, r \rangle \\ &\qquad \text{s.t.} \quad E^T \lambda = \gamma P^T \lambda + (1-\gamma)\mu.\end{aligned}$$

Remarks:

- The problem in the current form suffers from $|\mathcal{S}|$ many constraints.

The Lagrangian: Towards an unconstrained problem.

◦ The corresponding Lagrangian is:

$$\max_{\lambda \in \Delta} \min_{V} \; \langle \lambda, r \rangle + \langle V, \gamma P^T \lambda - E^T \lambda \rangle + (1-\gamma)\langle V, \mu \rangle.$$

◦ Applying proximal point we obtain the following update:

$$\lambda_k = \operatorname{argmax}_{\lambda \in \Delta} \underbrace{\min_{V} \; \langle \lambda, r \rangle + \langle V, \gamma P^T \lambda - E^T \lambda \rangle + (1-\gamma)\langle V, \mu \rangle}_{:= f(\lambda)} - \frac{1}{\eta} D_{\mathsf{KL}}(\lambda, \lambda_{k-1}).$$

KKT conditions on the Lagrangian update.

Derivation:

- We notice by convexity of the Bregman divergence that the update is convex in λ.
- We introduce an auxiliary problem for any V as follows:

$$\lambda_k^V = \operatorname{argmax}_{\lambda \in \Delta} \langle \lambda, r \rangle + \langle V, \gamma P^T \lambda - E^T \lambda \rangle + (1-\gamma)\langle V, \mu \rangle - \frac{1}{\eta} D_{\mathsf{KL}}(\lambda, \lambda_{k-1}).$$

- By optimality conditions, it must hold

$$r + \gamma PV - EV - \frac{1}{\eta} \nabla_\lambda D_{\mathsf{KL}}(\lambda_k^V, \lambda_{k-1}) = 0.$$

- Thus, λ_k^V can be computed in closed form for any V

$$\lambda_k^V(s,a) = \frac{\lambda_{k-1}(s,a) e^{\eta(r(s,a)+\gamma(PV)(s,a)-(EV)(s,a))}}{\sum_{s',a'} \lambda_{k-1}(s',a') e^{\eta(r(s',a')+\gamma(PV)(s',a')-(EV)(s',a'))}}.$$

The unconstrained problem

◦ We can leverage the KKT conditions to write an unconstrained problem where the only decision variable is V:

$$\min_V \langle \lambda_k^V, r\rangle + \langle V, \gamma P^T \lambda_k^V - E^T \lambda_k^V \rangle + (1-\gamma)\langle V, \mu\rangle - \frac{1}{\eta} D_{\mathsf{KL}}(\lambda_k^V, \lambda_{k-1}).$$

◦ With some calculus, we have the following compact form.

Unconstrained problem (REPS)

$$V_k = \min_V (1-\gamma)\langle \mu, V\rangle + \frac{1}{\eta}\log \sum_{s,a} \lambda_{k-1}(s,a) e^{\eta(r(s,a)+\gamma(PV)(s,a)-(EV)(s,a))}.$$

Remarks:

◦ The decision variable V has dimension $|\mathcal{S}|$.

◦ The objective is convex and smooth with Lipschitz continuous gradient.

The REPS algorithm [26]

Algorithm: REPS

Initialize λ_0 (for example uniform)

for each iteration $k = 1, \ldots, K$ **do**

Solve the problem

$$V_k = \min_V (1-\gamma)\langle \mu, V \rangle + \frac{1}{\eta} \log \sum_{s,a} \lambda_{k-1}(s,a) e^{\eta(r(s,a)+\gamma(PV)(s,a)-(EV)(s,a))}$$

Update the occupancy measure:

$$\lambda_k(s,a) \propto \lambda_{k-1}(s,a) e^{\eta(r(s,a)+\gamma(PV_k)(s,a)-(EV_k)(s,a))}$$

end for

Sample complexity of REPS [25]

Algorithm	Oracle	Output
REPS	Exact gradient	$\mathcal{O}\left(\frac{\lvert\mathcal{S}\rvert^{3/2}}{(1-\gamma)^2\epsilon^2}\right)$
REPS	Stochastic Biased Gradients	$\mathcal{O}\left(\frac{\lvert\mathcal{S}\rvert^{3/2}}{(1-\gamma)^8\beta^2\epsilon^8}\right)$

Remarks:

- The exact gradient case achieves the best-known sample complexity
  - e.g., comparable to NPG (see Lecture 5)
- The sample complexity with stochastic gradients degrades.
- For the stochastic gradient case, one needs to assume that $\lambda_k(s,a) \geq \beta > 0$.
  - it solves the exploration problem by assumption.

Wrap Up

- The LP approach allows us to formulate RL as a convex optimization problem.
- The primal and dual LP are equivalent formulations of the RL objective.
- The saddle point formulation combines the primal and dual viewpoint.
- Applying the proximal point algorithm to the dual program yields the celebrated REPS algorithm.
- Offline policy evaluation and optimization are needed when we only learn from previously collected data.
  - see supplementary material at the end!

- Next lecture: Policy gradient methods (Part 1)!

References I

[1] Y. Abbasi-Yadkori, P. L. Bartlett, and A. Malek.
Linear programming for large-scale Markov decision problems.
In *International Conference on Machine Learning (ICML)*, 2014.
32

[2] Amir Beck and Marc Teboulle.
Mirror descent and nonlinear projected subgradient methods for convex optimization.
Operations Research Letters, 31(3):167–175, 2003.
71, 74, 75

[3] V. S. Borkar.
A convex analytic approach to Markov decision processes.
Probability Theory and Related Fields, 78(4):583–602, 1988.
32

[4] Volkan Cevher and Bang Cong Vu.
A reflected forward-backward splitting method for monotone inclusions involving lipschitzian operators.
Set-Valued and Variational Analysis, pages 1–12, 2020.
35, 68

[5] Y. Chen, L. Li, and M. Wang.
Scalable bilinear π learning using state and action features.
In *International Conference on Machine Learning (ICML)*, 2018.
41, 88

References II

[6] D. P. De Farias and B. Van Roy.
The linear programming approach to approximate dynamic programming.
Operations Research, 51(6):850–865, 2003.
32, 90, 91

[7] D. P. De Farias and B. Van Roy.
On constraint sampling in the linear programming approach to approximate dynamic programming.
Mathematics of Operations Research, 29(3):462–478, 2004.
32, 92, 93

[8] Vijay V. Desai, Vivek F. Farias, and Ciamac C. Moallemi.
Approximate dynamic programming via a smoothed linear program.
Operations Research, 60(3):655–674, 2012.
32

[9] Alireza Fallah, Aryan Mokhtari, and Asuman E. Ozdaglar.
On the convergence theory of gradient-based model-agnostic meta-learning algorithms.
CoRR, abs/1908.10400, 2019.
67, 69

[10] O. Hernández-Lerma and J. B. Lasserre.
Discrete-Time Markov Control Processes: Basic Optimality Criteria.
Springer-Verlag New York, 1996.
32

References III

[11] O. Hernández-Lerma and J. B. Lasserre.
Further Topics on Discrete-Time Markov Control Processes.
Springer-Verlag New York, 1999.
32

[12] Y. Jin and A. Sidford.
Efficiently solving MDPs with stochastic mirror descent.
In *International Conference on Machine Learning (ICML)*, 2020.
41

[13] Sham M. Kakade, Shai Shalev-Shwartz, and Ambuj Tewari.
Regularization techniques for learning with matrices.
Journal of Machine Learning Research, 13(59):1865–1890, 2012.
81, 82

[14] G. M. Korpelevic.
An extragradient method for finding saddle-points and for other problems.
Èkonom. i Mat. Metody., 12(4):747–756, 1976.
66

[15] Galina M Korpelevich.
The extragradient method for finding saddle points and other problems.
Matecon, 12:747–756, 1976.
35

References IV

[16] C. Lakshminarayanan, S. Bhatnagar, and C. Szepesvári.
A linearly relaxed approximate linear program for Markov decision processes.
IEEE Transactions on Automatic Control, 63(4):1185–1191, 2018.
32

[17] Yura Malitsky and Matthew K Tam.
A forward-backward splitting method for monotone inclusions without cocoercivity.
SIAM Journal on Optimization, 30(2):1451–1472, 2020.
68

[18] A. Manne.
Linear programming and sequential decisions.
Management Science, 6(3):259–267, 1960.
32

[19] Jiri Matousek and Bernd Gärtner.
Understanding and using linear programming, volume 1.
Springer, 2007.
12

[20] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan Kumaran, Daan Wierstra, Shane Legg, and Demis Hassabis.
Human-level control through deep reinforcement learning.
Nature, 518(7540):529–533, 2015.
4

References V

[21] Aryan Mokhtari, Asuman Ozdaglar, and Sarath Pattathil.
A unified analysis of extra-gradient and optimistic gradient methods for saddle point problems: Proximal point approach.
In Silvia Chiappa and Roberto Calandra, editors, *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pages 1497–1507. PMLR, 26–28 Aug 2020.
35, 67, 69

[22] O. Nachum and B. Dai.
Reinforcement learning via Fenchel-Rockafellar duality.
arXiv:2001.01866, 2020.
78, 99

[23] Ofir Nachum, Yinlam Chow, Bo Dai, and Lihong Li.
Dualdice: Behavior-agnostic estimation of discounted stationary distribution corrections.
In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.
78

[24] Ofir Nachum, Bo Dai, Ilya Kostrikov, Yinlam Chow, Lihong Li, and Dale Schuurmans.
Algaedice: Policy gradient from arbitrary experience.
arXiv:1912.02074, 2019.
78, 100

[25] Aldo Pacchiano, Jonathan Lee, Peter Bartlett, and Ofir Nachum.
Near optimal policy optimization via REPS.
In A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021.
48

References VI

[26] Jan Peters, Katharina Mulling, and Yasemin Altun.
Relative entropy policy search.
In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 24, 2010.
42, 47

[27] Marek Petrik, Gavin Taylor, Ron Parr, and Shlomo Zilberstein.
Feature selection using regularization in approximate linear programs for markov decision processes.
In *International Conference on International Conference on Machine Learning (ICML)*, 2010.
32

[28] Marek Petrik and Shlomo Zilberstein.
Constraint relaxation in approximate linear programs.
In *International Conference on Machine Learning (ICML)*, 2009.
32

[29] Leonid Denisovich Popov.
A modification of the arrow-hurwicz method for search of saddle points.
Mathematical notes of the Academy of Sciences of the USSR, 28(5):845–848, 1980.
68

[30] M. L. Puterman.
Markov Decision Processes: Discrete Stochastic Dynamic Programming.
John Wiley & Sons, Inc., USA, 1st edition, 1994.
17, 18

References VII

[31] Alexander Rakhlin and Karthik Sridharan.
Optimization, learning, and games with predictable sequences.
arXiv preprint arXiv:1311.1869, 2013.
68

[32] R Tyrrell Rockafellar.
Conjugate convex functions in optimal control and the calculus of variations.
Journal of Mathematical Analysis and Applications, 32(1):174–222, 1970.
71

[33] R. Tyrrell Rockafellar.
Monotone operators and the proximal point algorithm.
SIAM Journal on Control and Optimization, 14(5):877–898, 1976.
35, 36, 65

[34] Paul J Schweitzer and Abraham Seidmann.
Generalized polynomial approximations in markovian decision processes.
Journal of Mathematical Analysis and Applications, 110(2):568–582, 1985.
32, 89

[35] W. Zhan, B. Huang, A. Huang, N. Jiang, and J. D. Lee.
Offline reinforcement learning with realizability and single-policy concentrability, 2022.
101

Supplementary

Mathematical background

Supplementary Material: Linear Programming Basics

Definition (LP)

A linear program in inequality form is an optimization problem of the form

$$\begin{aligned} \max_{\mathbf{x}\in\mathbb{R}^n} \quad & \mathbf{c}^T\mathbf{x} \\ \textit{s.t.} \quad & \mathbf{A}\mathbf{x} \leq \mathbf{b}, \end{aligned} \tag{6}$$

where $\mathbf{c}\in\mathbb{R}^n$, $\mathbf{b}\in\mathbb{R}^m$, *and* $\mathbf{A}\in\mathbb{R}^{m\times n}$.

Definition (Dual LP)

The dual LP of the LP in (6) *is*

$$\begin{aligned} \min_{\mathbf{y}\in\mathbb{R}^m} \quad & \mathbf{b}^T\mathbf{y} \\ \textit{s.t.} \quad & \mathbf{A}^T\mathbf{y} = \mathbf{c}, \\ & \mathbf{y} \geq \mathbf{0}. \end{aligned} \tag{7}$$

Supplementary Material: Linear Programming Basics (cont'd)

∘ We say that an LP has a *feasible solution* if there is an assignment satisfying its constraints. Formally, for 6 this means that here exists $\mathbf{x} \in \mathbb{R}^n$ such that $\mathbf{A}\mathbf{x} \leq \mathbf{b}$.

∘ We say that an LP is *bounded* if its objective is uniformly bounded across all feasible solutions. Formally, for 6 this means that $\sup \left\{ \mathbf{c}^T \mathbf{x} \mid \mathbf{A}\mathbf{x} \leq \mathbf{b} \right\} < \infty$.

Theorem (Strong duality)

Suppose that the primal LP in (6) *has a feasible solution and is bounded. Then both 6 and 7 attain optimal solutions* $\mathbf{x}^\star$ *and* $\mathbf{y}^\star$*, and they satisfy*

$$\mathbf{c}^T \mathbf{x}^\star = \mathbf{b}^T \mathbf{y}^\star.$$

∘ Self-study: Prove that in the LP formulation of MDPs, (D) is indeed the dual program of (P).

Supplementary Material: Dominated convergence

∘ To understand why we can swap limit and expectation, recall the dominated convergence theorem from real analysis.

Theorem (Dominated convergence, DCT)

Let $(f_n)_{n\in\mathbb{N}}$ be a sequence of real-valued measurable functions on some measure space (Ω, Σ, ν). Suppose f_n converges pointwise to f ($\lim_{n\to\infty} f_n(\omega) = f(\omega)$ for all $\omega \in \Omega$). Suppose further that $(f_n)_n$ is dominated by some integrable function g ($|f_n(\omega)| \le g(\omega)$ and $\int_\Omega |g_n| d\nu < \infty$). Then

$$\int_\Omega f d\nu = \lim_{n\to\infty} \int_\Omega f_n d\nu.$$

Supplementary Material: Dominated convergence (cont'd)

◦ On Slide 19, we used the DCT with

- (Ω, Σ, ν) the probability space over the trajectories $\tau = (s_0, a_0, s_1, a_1, s_2, \dots)$ under policy π
- $f_n(\tau) = \sum_{t=0}^{n} \gamma^t \mathbb{1} s_t = s, a_t = a$, which converge to $f(\tau) = \sum_{t=0}^{\infty} \gamma^t \mathbb{1} s_t = s, a_t = a$ pointwise
- $g(\tau) = \sum_{t=0}^{\infty} \gamma^t 1 = \frac{1}{1-\gamma}$.

Applying the DCT, we confirm

$$
\begin{aligned}
\mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t \mathbb{1} s_t = s, a_t = a \mid s_0 \sim \mu,\ \pi\right] &= \int_{\Omega} f d\nu \\
&= \lim_{n\to\infty} \int_{\Omega} f_n d\nu \\
&= \lim_{n\to\infty} \mathbb{E}\left[\sum_{t=0}^{n} \gamma^t \mathbb{1} s_t = s, a_t = a \mid s_0 \sim \mu,\ \pi\right] \\
&= \sum_{t=0}^{\infty} \gamma^t\ \mathbb{P}[s_t = s, a_t = a \mid s_0 \sim \mu,\ \pi],
\end{aligned}
$$

where the last step holds by linearity of expectation.

Supplementary

LP and optimization

Supplementary Material: Bellman Equation for State-action Visitation Distribution

Recall the definition

$$\lambda^\pi(s,a) := \sum_{t=0}^{\infty} \gamma^t \mathbb{P}[s_t = s, a_t = a \,|\, \pi, s_0 \sim \mu].$$

Bellman Equation for λ^π

$$\lambda^\pi(s,a) = \mu(s)\pi(a|s) + \gamma \sum_{s',a'} \pi(a|s)\mathsf{P}(s|s',a')\lambda^\pi(s',a').$$

Supplementary Material: Bellman Equation for State-action Visitation Distribution

Proof.

$$
\begin{aligned}
&\lambda^\pi(s,a)\\
=&\mathbb{P}[s_0=s,a_0=a]+\sum_{t=1}^{\infty}\gamma^t\mathbb{P}[s_t=s,a_t=a|\pi,s_0\sim\mu]\\
=&\mu(s)\pi(a|s)+\sum_{t=1}^{\infty}\gamma^t\sum_{s',a'}\mathbb{P}\Big[s_t=s,a_t=a|s_{t-1}=s',a_{t-1}=a',\pi,s_0\sim\mu\Big]\mathbb{P}\Big[s_{t-1}=s',a_{t-1}=a'|\pi,s_0\sim\mu\Big]\\
=&\mu(s)\pi(a|s)+\gamma\sum_{t=1}^{\infty}\mathbb{P}\Big[s_t=s,a_t=a|s_{t-1}=s',a_{t-1}=a'\Big]\mathbb{P}\Big[s_{t-1}=s',a_{t-1}=a'|\pi,s_0\sim\mu\Big]\\
=&\mu(s)\pi(a|s)+\gamma\sum_{t=1}^{\infty}\pi(a|s)\mathsf{P}(s|s',a')\sum_{t=1}^{\infty}\gamma^{t-1}\mathbb{P}\Big[s_{t-1}=s',a_{t-1}=a'|\pi,s_0\sim\mu\Big]\\
=&\mu(s)\pi(a|s)+\gamma\sum_{s',a'}\pi(a|s)\mathsf{P}(s|s',a')\lambda^\pi(s',a')
\end{aligned}
$$

where the third equality is due to Markov property. □

PPM guarantees for minimax optimization

Theorem (Convergence of PPM [33])

Suppose $(\mathbf{x}^k, \mathbf{y}^k)$ *be the iterates generated by PPM (i.e., (5)), then for the averaged iterates, it holds that*

$$\left| \Phi\left(\frac{1}{K}\sum_{k=1}^{K}\mathbf{x}^k, \frac{1}{K}\sum_{k=1}^{K}\mathbf{y}^k \right) - \Phi(\mathbf{x}^\star, \mathbf{y}^\star) \right| \leq \frac{\|\mathbf{x}^0 - \mathbf{x}^\star\|^2 + \|\mathbf{y}^0 - \mathbf{y}^\star\|^2}{\tau K}.$$

Theorem (Linear convergence [33])

Suppose $(\mathbf{x}^k, \mathbf{y}^k)$ *be the iterates generated by* (5), $\Phi(\cdot,\cdot)$ *is* μ_x*−strongly convex in* $\mathbf{x}$ *and* μ_y*−strongly concave in* $\mathbf{y}$. *Let* $\mu = \max\{\mu_x, \mu_y\}$. *Then, for any* $\tau > 0$, $(\mathbf{x}^k, \mathbf{y}^k)$ *satisfies the following*

$$r^{k+1} \leq \frac{1}{1+\mu\tau} r^k,$$

where $r^k = \|\mathbf{x}^k - \mathbf{x}^\star\|^2 + \|\mathbf{y}^k - \mathbf{y}^\star\|^2$.

Remark:
- Still need an implementable and convergent algorithm beyond the stylized bilinear case.
- Note what happens when $\tau \to \infty$.

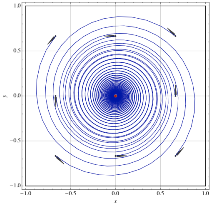
Extra-gradient algorithm (EG) [14]

EG method for saddle point problems

1. Choose $\mathbf{x}^0, \mathbf{y}^0$ and τ.

2. For $k = 0, 1, \cdots,$ perform:

$$\begin{aligned}
\tilde{\mathbf{x}}^k &:= \mathbf{x}^k - \tau \nabla_{\mathbf{x}} \Phi(\mathbf{x}^k, \mathbf{y}^k), \\
\tilde{\mathbf{y}}^k &:= \mathbf{y}^k + \tau \nabla_{\mathbf{y}} \Phi(\mathbf{x}^k, \mathbf{y}^k). \\
\mathbf{x}^{k+1} &:= \mathbf{x}^k - \tau \nabla_{\mathbf{x}} \Phi(\tilde{\mathbf{x}}^k, \tilde{\mathbf{y}}^k). \\
\mathbf{y}^{k+1} &:= \mathbf{y}^k + \tau \nabla_{\mathbf{y}} \Phi(\tilde{\mathbf{x}}^k, \tilde{\mathbf{y}}^k).
\end{aligned}$$



◦ Idea: Predict the gradient at the next point

$$\boxed{\mathbf{z}^{k+1} = \mathbf{z}^k - \tau \mathbf{V}\big(\underbrace{\mathbf{z}^k - \tau \mathbf{V}(\mathbf{z}^k)}_{\text{prediction of } \mathbf{z}^{k+1}} \big)} \tag{EG}$$

Remark: ◦ 1-extra-gradient computation per iteration

Extra-gradient algorithm: Convergence

Theorem (General case [9])

Let $0 < \tau \leq \frac{1}{L}$. It holds that

- *Iterates $(\mathbf{x}^k, \mathbf{y}^k)$ remains bounded in a convex compact set.*
- *Primal-dual gap reduces:* $\mathrm{Gap}\left(\frac{1}{K}\sum_{k=1}^{K}\mathbf{x}^k, \frac{1}{K}\sum_{k=1}^{K}\mathbf{y}^k\right) \leq \mathcal{O}\left(\frac{1}{K}\right)$.

Theorem (Linear convergence [21])

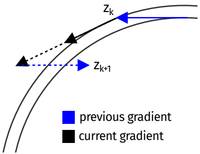
Suppose $(\mathbf{x}^k, \mathbf{y}^k)$ be the iterates generated by Extra-gradient algorithm, $\Phi(\cdot,\cdot)$ is μ_x-strongly convex in $\mathbf{x}$ and μ_y-strongly concave in $\mathbf{y}$. Let $\mu = \max\{\mu_x, \mu_y\}$. Then, for $\tau = \frac{1}{4L}$, $(\mathbf{x}^k, \mathbf{y}^k)$ satisfies,

$$r^{k+1} \leq \left(1 - \frac{1}{c\kappa}\right)^k r^0,$$

where $r^k = \|\mathbf{x}^k - \mathbf{x}^\star\|^2 + \|\mathbf{y}^k - \mathbf{y}^\star\|^2$, $\kappa = \frac{L}{\mu}$ is the condition number of the problem, and c is a constant which is independent of the problem parameters.

Optimistic gradient descent ascent algorithm (OGDA) [31]

OGDA for saddle point problems
1. Choose $\mathbf{x}^0, \mathbf{y}^0, \mathbf{x}^1, \mathbf{y}^1$ and τ.
2. For $k = 1, \cdots,$ perform:
$\mathbf{x}^{k+1} := \mathbf{x}^k - 2\tau\nabla_\mathbf{x}\Phi(\mathbf{x}^k, \mathbf{y}^k) + \tau\nabla_\mathbf{x}\Phi(\mathbf{x}^{k-1}, \mathbf{y}^{k-1}).$
$\mathbf{y}^{k+1} := \mathbf{y}^k + 2\tau\nabla_\mathbf{y}\Phi(\mathbf{x}^k, \mathbf{y}^k) - \tau\nabla_\mathbf{y}\Phi(\mathbf{x}^{k-1}, \mathbf{y}^{k-1}).$



- Main difference from the GDA: Add a "momentum" or "reflection" term to the updates

$$\boxed{\mathbf{z}^{k+1} = \mathbf{z}^k - \tau\left[\mathbf{V}(\mathbf{z}^k) + \underbrace{(\mathbf{V}(\mathbf{z}^k) - \mathbf{V}(\mathbf{z}^{k-1}))}_{\text{momentum}}\right].} \tag{OGDA}$$

- Known as Popov's method [29], it is also a special case of the Forward-Reflected-Backward method [17].
- It has ties to the Reflected-Forward-Backward Splitting (RFBS) method [4]:

$$\boxed{\mathbf{z}^{k+1} = \mathbf{z}^k - \tau\mathbf{V}(2\mathbf{z}^k - \mathbf{z}^{k-1}).} \tag{RFBS}$$

Remark: ◦ Advanced material at the end: OGDA is an approximation of PPM for bilinear problems.

OGDA: Convergence

Theorem (General case [9])

Let $0 < \tau \leq \frac{1}{2L}$, $\mathbf{x}^1 = \mathbf{x}^0, \mathbf{y}^1 = y^0$. It holds that

- *Iterates $(\mathbf{x}^k, \mathbf{y}^k)$ remains bounded in a convex compact set.*
- *Primal-dual gap reduces:* $\mathrm{Gap}\left(\frac{1}{K}\sum_{k=1}^K \mathbf{x}^k, \frac{1}{K}\sum_{k=1}^K \mathbf{y}^k\right) \leq \mathcal{O}\left(\frac{1}{K}\right)$.

Theorem (Linear convergence [21])

Suppose $(\mathbf{x}^k, \mathbf{y}^k)$ be the iterates generated by OGDA, $\Phi(\cdot,\cdot)$ is μ_x-strongly convex in $\mathbf{x}$ and μ_y-strongly concave in $\mathbf{y}$. Let $\mu = \max\{\mu_x, \mu_y\}$. Then, for $\tau = \frac{1}{4L}$, $(\mathbf{x}^k, \mathbf{y}^k)$ satisfies,

$$r^{k+1} \leq \left(1 - \frac{1}{c\kappa}\right)^k r^0,$$

where $r^k = \|\mathbf{x}^k - \mathbf{x}^\star\|^2 + \|\mathbf{y}^k - \mathbf{y}^\star\|^2$, $\kappa = \frac{L}{\mu}$ is the condition number of the problem, and c is a constant which is independent of the problem parameters.

*Bregman divergences

Table: **Bregman functions** $\psi(\mathbf{x})$ & corresponding Bregman divergences/distances $d_\psi(\mathbf{x},\mathbf{y})^a$.

Name (or Loss)	**Domain**[b]	$\psi(\mathbf{x})$	$d_\psi(\mathbf{x},\mathbf{y})$
Squared loss	$\mathbb{R}$	x^2	$(x-y)^2$
Itakura-Saito divergence	$\mathbb{R}_{++}$	$-\log x$	$\frac{x}{y} - \log\left(\frac{x}{y}\right) - 1$
Squared Euclidean distance	$\mathbb{R}^p$	$\\|\mathbf{x}\\|_2^2$	$\\|\mathbf{x}-\mathbf{y}\\|_2^2$
Squared Mahalanobis distance	$\mathbb{R}^p$	$\langle \mathbf{x}, \mathbf{A}\mathbf{x}\rangle$	$\langle (\mathbf{x}-\mathbf{y}), \mathbf{A}(\mathbf{x}-\mathbf{y})\rangle^c$
Entropy distance	p-simplex[d]	$\sum_i x_i \log x_i$	$\sum_i x_i \log\left(\frac{x_i}{y_i}\right)$
Generalized I-divergence	$\mathbb{R}^p_+$	$\sum_i x_i \log x_i$	$\sum_i \left(\log\left(\frac{x_i}{y_i}\right) - (x_i - y_i)\right)$
von Neumann divergence	$\mathbb{S}^{p\times p}_+$	$\mathbf{X}\log\mathbf{X} - \mathbf{X}$	$\mathrm{tr}\left(\mathbf{X}(\log\mathbf{X} - \log\mathbf{Y}) - \mathbf{X} + \mathbf{Y}\right)^e$
logdet divergence	$\mathbb{S}^{p\times p}_+$	$-\log\det\mathbf{X}$	$\mathrm{tr}\left(\mathbf{X}\mathbf{Y}^{-1}\right) - \log\det\left(\mathbf{X}\mathbf{Y}^{-1}\right) - p$

[a] $x, y \in \mathbb{R}$, $\mathbf{x},\mathbf{y} \in \mathbb{R}^p$ and $\mathbf{X},\mathbf{Y} \in \mathbb{R}^{p\times p}$.
[b] $\mathbb{R}_+$ and $\mathbb{R}_{++}$ denote non-negative and positive real numbers respectively.
[c] $\mathbf{A} \in \mathbb{S}^{p\times p}_+$, the set of symmetric positive semidefinite matrix.
[d] p-simplex:= $\{\mathbf{x}\in\mathbb{R}^p : \sum_{i=1}^p x_i = 1, x_i \geq 0, i = 1,\dots,p\}$
[e] $\mathrm{tr}(\mathbf{A})$ is the trace of $\mathbf{A}$.

*Mirror descent [2]

What happens if we use a Bregman distance d_ψ in gradient descent?

Let $\psi : \mathbb{R}^p \to \mathbb{R}$ be a μ-strongly convex and continuously differentiable function and let the associated Bregman distance be $d_\psi(\mathbf{x}, \mathbf{y}) = \psi(\mathbf{x}) - \psi(\mathbf{y}) - \langle \mathbf{x} - \mathbf{y}, \nabla\psi(\mathbf{y})\rangle$.
Assume that the inverse mapping $\psi^\star$ of ψ is easily computable (i.e., its convex conjugate).

- **Majorize**: Find α_k such that

$$f(\mathbf{x}) \leq f(\mathbf{x}^k) + \langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k\rangle + \frac{1}{\alpha_k} d_\psi(\mathbf{x}, \mathbf{x}^k) := Q_\psi^k(\mathbf{x}, \mathbf{x}^k)$$

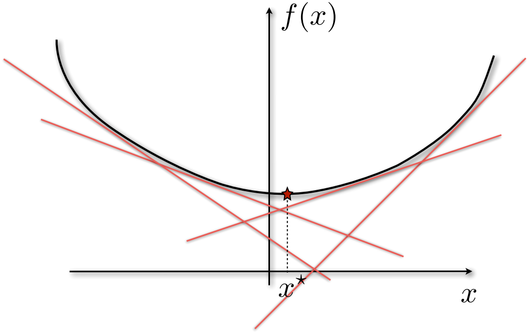
- **Minimize**

$$\begin{aligned}
\mathbf{x}^{k+1} &= \arg\min_{\mathbf{x}} Q_\psi^k(\mathbf{x}, \mathbf{x}^k) \Rightarrow \nabla f(\mathbf{x}^k) + \frac{1}{\alpha_k}\left(\nabla\psi(\mathbf{x}^{k+1}) - \nabla\psi(\mathbf{x}^k)\right) = 0 \\
\nabla\psi(\mathbf{x}^{k+1}) &= \nabla\psi(\mathbf{x}^k) - \alpha_k \nabla f(\mathbf{x}^k) \\
\mathbf{x}^{k+1} &= \nabla\psi^\star(\nabla\psi(\mathbf{x}^k) - \alpha_k\nabla f(\mathbf{x}^k)) \qquad (\nabla\psi(\cdot))^{-1} = \nabla\psi^\star(\cdot)[32].
\end{aligned}$$

- Mirror descent is a **generalization** of gradient descent for functions that are Lipschitz-gradient in norms other than the Euclidean.
- MD allows to deal with some **constraints** via a proper choice of ψ.

*What to keep in mind about mirror descent?

- Approximates the optimum by lower bounding the function via hyperplanes at $\mathbf{x}_t$



- The smaller the gradients, the better the approximation!

*Mirror descent example

How can we minimize a convex function over the unit simplex?

$$\min_{\mathbf{x}\in\Delta} f(\mathbf{x}),$$

where

- $\Delta := \{\mathbf{x}\in\mathbb{R}^p \ : \ \sum_{j=1}^p x_j = 1, \mathbf{x}\geq 0\}$ is the **unit simplex**;
- f is convex L_f-Lipschitz continuous with respect to some norm $\|\cdot\|$. (not necessarily *L-Lipschitz gradient*)

Entropy function

- Define the entropy function

$$\psi_e(\mathbf{x}) = \sum_{j=1}^p x_j \ln x_j \quad \text{if } \mathbf{x}\in\Delta, \quad +\infty \text{ otherwise.}$$

- ψ_e is 1-strongly convex over $\mathrm{int}\Delta$ with respect to $\|\cdot\|_1$.
- $\psi_e^\star(\mathbf{z}) = \ln\sum_{j=1}^p e^{z_j}$ and $\|\nabla\psi_e(\mathbf{x})\|\to\infty$ as $\mathbf{x}\to\tilde{\mathbf{x}}\in\Delta$.
- Let $\mathbf{x}^0 = p^{-1}\mathbf{1}$, then $d_\psi(\mathbf{x},\mathbf{x}^0)\leq \ln p$ for all $\mathbf{x}\in\Delta$.

⋆Entropic descent algorithm [2]

Entropic descent algorithm (EDA)

Let $\mathbf{x}^0 = p^{-1}\mathbf{1}$ and generate the following sequence

$$x_j^{k+1} = \frac{x_j^k e^{-t_k f_j'(\mathbf{x}^k)}}{\sum_{j=1}^p x_j^k e^{-t_k f_j'(\mathbf{x}^k)}}, \quad t_k = \frac{\sqrt{2 \ln p}}{L_f} \frac{1}{\sqrt{k}},$$

where $f'(\mathbf{x}) = (f_1(\mathbf{x})', \ldots, f_p(\mathbf{x})')^T \in \partial f(\mathbf{x})$, which is the **subdifferential** of f at $\mathbf{x}$.

- This is an example of **non-smooth** and **constrained** optimization;
- The updates are multiplicative.

*Convergence of mirror descent

Problem

$$\min_{\mathbf{x}\in\mathcal{X}} f(\mathbf{x}) \tag{8}$$

where

- $\mathcal{X}$ *is a closed convex subset of* $\mathbb{R}^p$;
- f *is convex* L_f*-Lipschitz continuous with respect to some norm* $\|\cdot\|$.

Theorem ([2])

Let $\{\mathbf{x}^k\}$ *be the sequence generated by mirror descent with* $\mathbf{x}^0 \in \mathrm{int}\mathcal{X}$.
If the step-sizes are chosen as

$$\alpha_k = \frac{\sqrt{2\mu d_\psi(\mathbf{x}^\star, \mathbf{x}^0)}}{L_f}\frac{1}{\sqrt{k}}$$

the following convergence rate holds

$$\min_{0\leq s\leq k} f(\mathbf{x}^s) - f^\star \leq L_f\sqrt{\frac{2d_\psi(\mathbf{x}^\star, \mathbf{x}^0)}{\mu}}\frac{1}{\sqrt{k}}$$

- *This convergence rate is* **optimal** *for solving* (8) *with a first-order method.*

Supplementary material

Offline policy evaluation

A primal LP for policy evaluation.

◦ Recall that $Q^\pi(s,a)$ is a fixed point for the expectation Bellman operator $\mathcal{T}^\pi$.

$$Q^\pi(s,a) = (\mathcal{T}^\pi Q^\pi)(s,a) = r(s,a) + \gamma \sum_{s',a'} \mathsf{P}(s'|s,a)\pi(a'|s')Q^\pi(s',a')$$

Derivation:

◦ It follows that Q^π belongs to the set given by

$$\left\{ Q \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|} : Q^\pi(s,a) \geq r(s,a) + \gamma \sum_{s',a'} \mathsf{P}(s'|s,a)\pi(a'|s')Q^\pi(s',a') \right\}$$

◦ Therefore, we can write the following program for Q^π:

$$\begin{aligned} Q^\pi &= \operatorname{argmin}_Q \langle c, Q \rangle \\ &\quad \text{s.t.} Q(s,a) \geq r(s,a) + \gamma \sum_{s',a'} \mathsf{P}(s'|s,a)\pi(a'|s')Q(s',a') \quad \forall s,a \in \mathcal{S} \times \mathcal{A} \end{aligned}$$

◦ The variable c is a vector of dimension $|\mathcal{S}||\mathcal{A}|$ defined as $c(s,a) = (1-\gamma)\pi(a|s)\mu(s)$.

The corresponding dual LP.

◦ With standard techniques we can derive the following dual formulation over the occupancy measure.

$$\lambda^\pi = \operatorname{argmax}_{\lambda \geq 0} \langle r, \lambda \rangle$$

$$\text{s.t.} \lambda(s,a) = \gamma \sum_{s',a'} \mathsf{P}(s|s',a')\pi(a|s)\lambda(s',a') + c(s,a) \quad \forall s,a \in \mathcal{S} \times \mathcal{A}$$

Remark:

◦ The only feasible point is λ^π [22].

◦ We can change the objective without affecting the maximizer.

◦ However, we change the objective value.

◦ Several recent works proposed to add an f-divergence to the objective. [22, 24, 23]

A modified Dual LP

Dual LP with f-divergences

$$\lambda^\pi = \text{argmax}_{\lambda \geq 0} \langle r, \lambda \rangle - \frac{1}{\eta} D_f(\lambda, \widetilde{\lambda^\pi})$$

$$\text{s.t.}\lambda(s,a) = \gamma \sum_{s',a'} \mathsf{P}(s|s',a')\pi(a|s)\lambda(s',a') + c(s,a) \quad \forall s,a \in \mathcal{S} \times \mathcal{A}$$

Remarks:

- ∘ Notice that the constraints are different from the one used in the LP formulation for REPS.
- ∘ We use more general f-divergences D_f instead than KL divergence.
- ∘ The center point is $\widetilde{\lambda^\pi}$ as opposed to λ_{k-1}.

Conjugation of functions

∘ Idea: Represent a convex function in $\max$-form:

Definition

Let $\mathcal{Q}$ be a Euclidean space and $\mathcal{Q}^*$ be its dual space. Given a proper, closed and convex function $f : \mathcal{Q} \to \mathbb{R} \cup \{+\infty\}$, the function $f^* : \mathcal{Q}^* \to \mathbb{R} \cup \{+\infty\}$ such that

$$f^*(\mathbf{y}) = \sup_{\mathbf{x} \in \mathrm{dom}(f)} \left\{ \mathbf{y}^T \mathbf{x} - f(\mathbf{x}) \right\}$$

is called the Fenchel conjugate (or conjugate) of f.

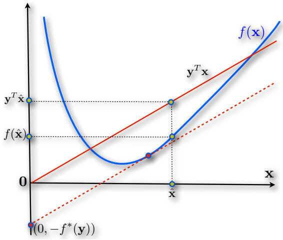


Figure: The conjugate function $f^*(\mathbf{y})$ is the maximum gap between the linear function $\mathbf{x}^T\mathbf{y}$ (red line) and $f(\mathbf{x})$.

Observations:
∘ $\mathbf{y}$: slope of the hyperplane
∘ $-f^*(\mathbf{y})$: intercept of the hyperplane

Conjugation of functions

Definition

Given a proper, closed and convex function $f : \mathcal{Q} \to \mathbb{R} \cup \{+\infty\}$, the function $f^* : Q^* \to \mathbb{R} \cup \{+\infty\}$ such that

$$f^*(\mathbf{y}) = \sup_{\mathbf{x} \in \mathrm{dom}(f)} \left\{ \mathbf{y}^T \mathbf{x} - f(\mathbf{x}) \right\}$$

is called the Fenchel conjugate (or conjugate) of f.

Conjugation of functions

Definition

Given a proper, closed and convex function $f : \mathcal{Q} \to \mathbb{R} \cup \{+\infty\}$, the function $f^* : Q^* \to \mathbb{R} \cup \{+\infty\}$ such that

$$f^*(\mathbf{y}) = \sup_{\mathbf{x} \in \mathrm{dom}(f)} \left\{ \mathbf{y}^T \mathbf{x} - f(\mathbf{x}) \right\}$$

is called the Fenchel conjugate (or conjugate) of f.

Properties

- f^* is a convex and lower semicontinuous function by construction as the supremum of affine functions of $\mathbf{y}$.
- The conjugate of the conjugate of a convex function f is the same function f; i.e., $f^{**} = f$ for $f \in \mathcal{F}(\mathcal{Q})$.
- The conjugate of the conjugate of a non-convex function f is its lower convex envelope when $\mathcal{Q}$ is compact:
  - $f^{**}(\mathbf{x}) = \sup\{g(\mathbf{x}) : g$ is convex and $g \leq f, \forall \mathbf{x} \in \mathcal{Q}\}$.
- For closed convex f, μ-strong convexity w.r.t. $\|\cdot\|$ is equivalent to $\frac{1}{\mu}$ smoothness of f^* w.r.t. $\|\cdot\|_*$.
  - Recall dual norm: $\|\mathbf{y}\|_* = \sup_{\mathbf{x}}\{\langle \mathbf{x}, \mathbf{y} \rangle : \|\mathbf{x}\| \leq 1\}$.
  - See for example Theorem 3 in [13].

Fenchel duality of f-divergence

∘ Using Fenchel conjugation, we can rewrite an f-divergence as follows:

$$D_f(\lambda, \widetilde{\lambda^\pi}) = \sum_{s,a} \widetilde{\lambda^\pi}(s,a) f\left(\frac{\lambda(s,a)}{\widetilde{\lambda^\pi}(s,a)}\right) = \max_u \sum_{s,a} \lambda(s,a)u(s,a) - \widetilde{\lambda^\pi}(s,a) f^\star\left(u(s,a)\right)$$

where we used the dual function $u : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$.

Remark:

∘ When seeing $D_f(\lambda, \widetilde{\lambda^\pi})$ as a function of λ, we have that its Fenchel conjugate is given by the following expression $(D_f(\cdot, \widetilde{\lambda^\pi}))^* = \langle \widetilde{\lambda^\pi}, f^*(\cdot)\rangle$

Some additional operators towards the Lagrangian

- For compacteness we will consider the Bellman evaluation operator $\mathcal{L}_\pi : \mathbb{R}^{\mathcal{S}\times\mathcal{A}} \rightarrow \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$
- The action on $Q(s,a)$ is

$$(\mathcal{L}^\pi Q)(s,a) = Q(s,a) - \gamma \sum_{s',a'} \mathsf{P}(s'|s,a)\pi(a'|s')Q(s',a')$$

- The adjoint operator $\mathcal{L}_\pi^* : \mathbb{R}^{\mathcal{S}\times\mathcal{A}} \rightarrow \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$
- The action on $\lambda(s,a)$ is

$$(\mathcal{L}_\pi^* \lambda)(s,a) = \lambda(s,a) - \gamma \sum_{s',a'} \mathsf{P}(s|s',a')\pi(a|s)\lambda(s',a')$$

The Lagrangian

Derivation:

- Thanks to the Bellman evaluation operator we have that

$$\lambda^{\pi} \;=\; \mathrm{argmax}_{\lambda\geq 0} \min_{Q} \langle r, \lambda\rangle - \frac{1}{\eta} D_f(\lambda, \widetilde{\lambda^{\pi}}) - \langle Q, \mathcal{L}_{\pi}^{*}\lambda\rangle + \langle Q, c\rangle$$

- Rearranging the terms:

$$\lambda^{\pi} \;=\; \mathrm{argmax}_{\lambda\geq 0} \min_{Q} \langle r - \mathcal{L}_{\pi} Q, \lambda\rangle - \frac{1}{\eta} D_f(\lambda, \widetilde{\lambda^{\pi}}) + \langle Q, c\rangle$$

- Exchanging $\max$ and $\min$ by strong duality:

$$Q^{\pi} \;=\; \mathrm{argmin}_{Q} \max_{\lambda\geq 0} \langle r - \mathcal{L}_{\pi} Q, \lambda\rangle - \frac{1}{\eta} D_f(\lambda, \widetilde{\lambda^{\pi}}) + \langle Q, c\rangle$$

- Recognizing the Fenchel dual:

$$Q^{\pi} \;=\; \mathrm{argmin}_{Q} \langle \widetilde{\lambda^{\pi}}, f^{*}(\eta(r - \mathcal{L}_{\pi} Q))\rangle + \langle Q, c\rangle$$

- We derived the formulation used in AlgaeDICE for policy evaluation.

LP with function approximation

a.k.a. Approximate Linear Programming (ALP)

Scaling up primal-dual π-learning

Large-scale MDPs $\Rightarrow$ Large-scale optimization

◦ Parameterize λ and V via linear functions

- $\lambda_\nu = \Psi\nu$, for some feature matrix $\Psi \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|\times n}$
- $V_\theta = \Phi\theta$, for some feature matrix $\Phi \in \mathbb{R}^{|\mathcal{S}|\times m}$

Assumption: The columns of Ψ are probability distributions.

Relaxed saddle point formulation

$$\min_{\theta} \max_{\nu \in \Delta_{[n]}} (1-\gamma)\langle \mu , \Phi\theta\rangle + \langle \nu , \Psi^\top (r + \gamma P\Phi\theta - E\Phi\theta)\rangle$$

Scaling up primal-dual π-learning(cont'd)

Relaxed saddle point formulation

$$\min_{\theta} \max_{\nu \in \Delta_{[n]}} (1-\gamma)\langle \mu \,,\, \Phi\theta \rangle + \langle \nu \,,\, \Psi^\top (r + \gamma P \Phi\theta - E\Phi\theta) \rangle$$

- Primal-dual updates:
  - $\theta_{k+1} = \theta_k - \eta\Big((\gamma P\Phi - E\Phi)^\top \Psi \nu_k + \Phi^\top \mu\Big)$,
  - $\nu_{k+1} \propto \nu_k \odot e^{\eta \Psi^\top (r + \gamma P \Phi \theta_k - E\Phi\theta_k)}$.
- Implementable with only sample access to the columns of Ψ and the transition law P [Chen et al. 2018] [5].
  - $\mathcal{O}\left(\frac{n\, m \log(\frac{1}{\delta})}{(1-\gamma)^4 \varepsilon^2}\right)$ samples for finding an $\varepsilon + \varepsilon_{\text{approx}}$-optimal policy with probability at least $1-\delta$.
  - $\varepsilon_{\text{approx}}$ captures the expressivity of the approximation architecture.

Prior works in ALP - Linear function approximation

Large-scale MDPs $\Rightarrow$ Large-scale optimization

- Reduce the number of decision variables by projecting onto a lower-dimensional subspace.
  - Let $\phi_1, \ldots, \phi_k : \mathcal{S} \to \mathbb{R}$ be k basis functions (or features).
  - $\Phi := \begin{bmatrix} \phi_1 & \ldots & \phi_k \end{bmatrix} \in \mathbb{R}^{|\mathcal{S}| \times k}$ is the corresponding feature matrix.
  - The (ALP) is obtained by adding the linear constraint $V = \Phi\theta = \sum_{i=1}^{k} \theta_i \phi_i$ to the original primal LP (P).

Approximate linear program [Schweitzer & Seidman 1982] [34]

$$
\begin{aligned}
\min_{\theta \in \mathbb{R}^k} \quad & (1-\gamma) \sum_{s \in \mathcal{S}} \mu(s)(\Phi\theta)(s) \\
\text{s.t.} \quad & (\Phi\theta)(s) \geq \; r(s,a) + \gamma \sum_{s' \in \mathcal{S}} \mathrm{P}(s'|s,a)(\Phi\theta)(s'), \quad \forall\; s \in \mathcal{S},\; a \in \mathcal{A}.
\end{aligned}
\tag{ALP}
$$

Prior works in ALP - Linear function approximation (cont'd)

Assumptions:

- The set $\{\phi_1, \ldots, \phi_k\}$ is linearly independent.
- $\mathbf{1} \in \operatorname{span}\big(\{\phi_1, \ldots, \phi_k\}\big) := \{\Phi\theta \mid \theta \in \mathbb{R}^k\}$. This ensures that (ALP) is feasible [6] .
- The values $\sum_{s' \in \mathcal{S}} \mathsf{P}(s'|s,a)\phi_i(s')$ and $\mu^\top \phi_i$, $i = 1, \ldots, k$, can be accessed in $\mathcal{O}(1)$ time.

Quality of the approximate solution (Th.2 in [De Farias & Van Roy 2003] [6])

$$\|V^\star - V^\star_{\text{ALP}}\|_{1,\mu} \le \frac{2}{1-\gamma} \underbrace{\min_\theta \|V^\star - \Phi\theta\|_\infty}_{\varepsilon_{\text{approx}}:\ \text{approximation error}}.$$

Notation:

- $\theta^\star_{\text{ALP}}$ is optimal to (ALP) and $V^\star_{\text{ALP}} = \Phi\theta^\star_{\text{ALP}}$ is the approximate value function.
- $\|V\|_{1,\mu} := \sum_{s \in \mathcal{S}} \mu(s)|V(s)|$ is the μ-weighted ℓ_1-norm, where $\mu > 0$.
- $\Phi\theta^\star$ is the $\|\cdot\|_\infty$-norm projection of $V^\star$ to the subspace $V = \Phi\theta$.
- $\varepsilon_{\text{approx}} := \min_\theta \|V^\star - \Phi\theta\|_\infty = \|V^\star - \Phi\theta^\star\|_\infty$ is called the approximation error.

Prior works in ALP - Linear function approximation (cont'd)

Quality of the approximate solution

$$\|V^\star - V^\star_{\text{ALP}}\|_{1,\mu} \leq \frac{2}{1-\gamma}\varepsilon_{\text{approx}}.$$

Remarks:

- $\varepsilon_{\text{approx}} = \min_\theta \|V^\star - \Phi\theta\|_\infty$ captures the approximation power of the feature map.
- If $V^\star \in \text{span}\big(\phi_1, \ldots, \phi_k\big)$, then $V^\star = \Phi\theta^\star_{\text{ALP}}$.
- In general, $\|V^\star - V^\star_{\text{ALP}}\|_{1,\mu} = \mathcal{O}(\varepsilon_{\text{approx}})$.
- Focus on finding a good basis, leaving the search of the "right" weights to an LP solver.

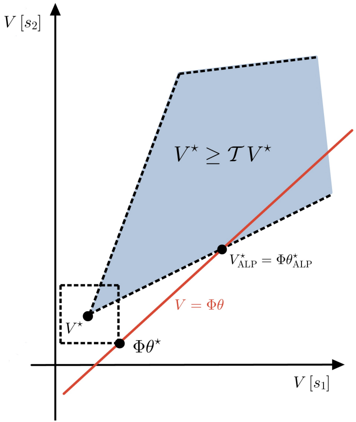


Figure: Graphical interpretation of ALP [6]

Prior works in ALP - Constraint sampling

∘ Reduce the number of constraints by constraint sampling.

- (x, a) is treated as an uncertainty parameter.
- $\mathcal{S} \times \mathcal{A}$ is the uncertainty space.
- $\mathbb{P}$ is a probability distribution on $\mathcal{S} \times \mathcal{A}$.
- $\{(s_i, a_i)\}_{i=1}^N$ i.i.d. samples on $(\mathcal{S} \times \mathcal{A}, \mathbb{P})$.
- $\mathcal{N} \subset \mathbb{R}^k$ is a bounding set.
- The relaxed LP (RLP) is obtained from (ALP) by restricting $\theta \in \mathcal{N}$ with N sampled constraints.

Relaxed linear program [De Farias & Van Roy 2001] [7]

$$\begin{aligned} \min_{\theta \in \mathcal{N}} \quad & (1-\gamma) \sum_{s \in \mathcal{S}} \mu(s)(\Phi\theta)(s) \\ \text{s.t.} \quad & (\Phi\theta)(s_i) \geq r(s_i, a_i) + \gamma \sum_{s' \in \mathcal{S}} \mathsf{P}(s'|s_i, a_i)(\Phi\theta)(s'), \quad \forall\, i = 1, \ldots, N. \end{aligned} \tag{RLP}$$

Prior works in ALP - Constraint sampling (cont'd)

Assumptions:

- The set $\mathcal{N} \subset \mathbb{R}^k$ is compact, i.e., bounded and closed.
- The optimal solution $\theta^\star_{\text{ALP}}$ to (ALP) is in $\mathcal{N}$.
- The sampling probability distribution is $\mathbb{P} \propto \lambda^{\pi^\star}$, i.e., the state-action visitation distribution induced by an optimal policy $\pi^\star$.

How many samples give a good solution (Th.3.1 in [De Farias & Van Roy 2004] [7])

Let $\varepsilon, \delta \in (0,1)$. If $N \geq \tilde{\mathcal{O}}\Big(\frac{4k\log(\frac{1}{\delta})}{(1-\gamma)\varepsilon}\frac{\sup_{\theta\in\mathcal{N}}\|V^\star-\Phi\theta\|_\infty}{\mu^\top V^\star}\Big)$, then with probability at least $1-\delta$, we have

$$\|V^\star - V^\star_{\text{RLP}}\|_{1,\mu} \leq \|V^\star - V^\star_{\text{ALP}}\|_{1,\mu} + \varepsilon\|V^\star\|_{1,\mu},$$

where the probability is taken over the random sampling of constraints.

Notation:

- $\theta^\star_{\text{RLP}}$ is optimal to (RLP) and $V^\star_{\text{RLP}} = \Phi\theta^\star_{\text{RLP}}$ is the approximate value function.
- $\varepsilon \in (0,1)$ is the desired approximation accuracy.
- $\delta \in (0,1)$ is the desired confidence level.

Prior works in ALP - Constraint sampling (cont'd)

Remarks:

- (RLP) is a relaxation of (ALP).
- The constraint $\theta \in \mathcal{N}$ ensures that the optimal value of (RLP) is bounded.
- The relaxed linear program (RLP) is random.
- $\theta^\star_{\text{RLP}}$ and $V^\star_{\text{RLP}} = \Phi\theta^\star_{\text{RLP}}$ are random variables.
- A lower bound on the number of samples needed to achieve an ε-accurate solution with probability at least $1-\delta$, is called the sample complexity of the problem.
- The sample complexity bound depends on the choice of the bounding set $\mathcal{N}$.
- The sample complexity bound requires access to samples from the optimal state-action visitation distribution (which is not known a priori).

Common theme of all prior ALP works

- Reduce the number of decision variables by projecting on a low-dimensional subspace.
- Reduce the number of constraints (e.g., by constraint sampling).
- Solve the resulted LP with generic solver.
- Analyze the quality of the approximate solution.
- Either scale badly with the size of the state-action spaces or
- Require access to samples from a distribution that depends on the optimal policy.
- Require knowledge of dynamics or access to a simulator.
- Focus mainly on the approximation of the optimal value function but not so much on extracting a nearly optimal policy.

Off-policy reinforcement learning (aka batch reinforcement learning)

◦ Learn to control from a previously collected dataset.

◦ Important for safety-critical applications, where deploying a suboptimal policy during learning is impossible.

- Think about drug testing.

Remarks:

◦ This setting is distinct from IRL, where the data is given by an "expert" policy.

◦ In this setting, we do have access to a reward signal from previous experience.

◦ We assume that the data covers the state-action space sufficiently well.

Off-policy reinforcement learning: The formalism

◦ In off-policy RL, we focus on the usual objective, which is:

$$J(\pi) = \mathbb{E}_{s\sim\mu}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t) \,|\, s_0 = s,\, \pi\right].$$

◦ However, we assume access only to samples from a fixed policy $\widetilde{\pi}$.

Remarks: ◦ The policy $\widetilde{\pi}$ represents the policy previously used to collect the experience dataset.

◦ In drug testing, $\widetilde{\pi}$ may represent the policy used by the human doctors (not necessarily optimal).

A useful subproblem: Offline policy evaluation

◦ We saw that often we find an optimal policy via learning the state-action value function:

$$Q^\pi(s,a) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t,a_t) \,|\, s_0 = s,\, a_0 = a,\, \pi\right].$$

◦ However, we assume access only to samples from a fixed policy $\widetilde{\pi}$.

◦ Estimating $Q^\pi(s,a)$ using samples from $\widetilde{\pi}$ is known as **offline policy evaluation**.

◦ Next, we derive a convex programming approach to compute $Q^\pi(s,a)$.

Self-study: ◦ Compare to the derivation of the Primal LP to compute $V^\star$.

An offline policy evaluation (OPE) approach

OPE via f-divergences

Let g be the convex conjugate of an f-divergence. [22] proposes to use the following formulation via Q^π:

$$Q^\pi \;=\; \text{argmin}_Q \mathbb{E}_{\lambda^{\tilde{\pi}}} g(r - \mathcal{L}_\pi Q) + (1-\gamma)\langle Q, c\rangle, \tag{OPE}$$

where $c(s,a) = \pi(a|s)\mu(s)$ is the joint state-action distribution.

Remarks:

- Recall the operator $\mathcal{L}^\pi$:

$$(\mathcal{L}^\pi Q)(s,a) = Q(s,a) - \gamma \sum_{s',a'} \mathsf{P}(s'|s,a)\pi(a'|s')Q(s',a').$$

- The problem (OPE) is convex and smooth in Q because g is convex.
- The problem (OPE) is unconstrained and g acts like a loss function.
- A biased objective estimate can be obtained by sampling from c and $\widetilde{\lambda^\pi}$.
- The name *offline* comes from not needing samples from λ^π.

From policy evaluation to policy optimization

AlgaeDICE [24]

Maximizing (OPE) objective over π gives us a policy optimization objective, dubbed as AlgaeDICE:

$$\pi^\star \in \operatorname{argmax}_\pi \min_Q \; (1-\gamma)\langle c, Q\rangle + \mathbb{E}_{\lambda^{\tilde{\pi}}} g\left(r - \mathcal{L}_\pi Q\right).$$

Remarks:

- We only need to sample from the initial distribution μ, the policy π, and the offline policy $\tilde{\pi}$.
- We only interact with the environment via $\tilde{\pi}$.

An alternative offline policy evaluation from the Lagrangian perspective [35]

- The approach in [35] *PRO-RL* exploits the Lagrangian of (LP) formulation.
- It has the same underpinnings of REPS adapted for the offline RL.

PRO-RL [35]

Let h be a strongly convex function. The PRO-RL approach uses the following formulation:

$$\max_{\lambda \in \Delta} \min_{V} \; \langle \lambda, r + \gamma PV - V \rangle + (1-\gamma)\langle \mu, V \rangle - \frac{1}{\eta} \mathbb{E}_{(s,a)\sim\lambda^{\tilde{\pi}}} \left(h\Big(\frac{\lambda(s,a)}{\lambda^{\tilde{\pi}}(s,a)} \Big) \right).$$

Remarks:

- The inner product with λ are equivalent to expectations with samples drawn from λ:

$$\langle \lambda, r + \gamma PV - V \rangle = \mathbb{E}_{(s,a)\sim\lambda} \left[r(s,a) + \gamma PV(s,a) - V(s) \right].$$

- [35] proposes to optimize an empirical objective obtained from samples.
- AlgaeDICE is a Q-based offline RL approach, whereas PRO-RL is value-based.

Guarantees for PRO-RL

Algorithm	Main assumptions	Samples for ϵ-optimal policy
PRO-RL	$\frac{\lambda^\star(s,a)}{\lambda^{\tilde{\pi}}(s,a)} \leq B < \infty$, $h(\cdot)$ is M_h-strongly convex	$\mathcal{O}\left(\frac{B\|\mathcal{S}\|}{(1-\gamma)^4 \epsilon^6 M_f}\right)$

Remarks:

- The assumption $\frac{\lambda^\star(s,a)}{\lambda^{\tilde{\pi}}(s,a)} < \infty$ has the interpretation that the occupancy measure $\lambda^{\tilde{\pi}}$ has support larger than the support of the optimal occupancy measure $\lambda^\star$.
- The sample complexity gurantees worsen as B increases.
- That means that the more "different" $\lambda^{\tilde{\pi}}$ and $\lambda^\star$ are, the more samples are required.