A Dual–SPMA Framework for Convex MDPs

Fenchel Duality + Softmax Policy Mirror Ascent

Shervin Khamooshian Ahmed Magd Pegah Aryadoost Danielle Nguyen

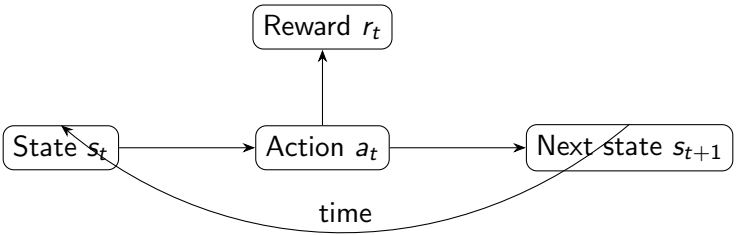
School of Computing Science, Simon Fraser University

Project Presentation

Outline

What is an MDP?

- Agent interacts with environment over time:
  - Observe state s_t
  - Choose action $a_t \sim \pi(\cdot|s_t)$
  - Environment transitions to s_{t+1}, emits reward r_t
- Policy π: mapping from states to action distributions.
- Goal: do well in the long run.



Standard RL objective

- Discounted return:

$$J(\pi) = (1-\gamma)\,\mathbb{E}\Big[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t)\Big]$$

- RL aims to find $\pi^\star = \arg\max_\pi J(\pi)$.
- We will re-express this in terms of *where* the policy spends time.

Occupancy measure: where the policy spends time

- **Discounted occupancy measure:**

$$d_\pi(s,a) = (1-\gamma)\sum_{t=0}^{\infty}\gamma^t \Pr_\pi(s_t = s, a_t = a)$$

- Think of d_π as a heatmap over (s,a):
  - High where the policy visits often.
  - Low where it rarely goes.
- d_π is a probability distribution over (s,a).

RL is linear in occupancy

- We can rewrite the return as:

$$J(\pi) = \sum_{s,a} r(s,a) d_\pi(s,a) = \langle r, d_\pi \rangle$$

- Standard RL $\Rightarrow$ maximize a **linear** function of d_π.
- But many interesting goals are not linear in d_π:
  - Safety constraints.
  - Imitation (matching an expert occupancy).
  - Exploration / entropy bonuses.

Convex MDPs

- Let D be the set of feasible occupancies (flow constraints).
- **Convex MDP:**

$$\min_{\pi} f(d_\pi) \quad \Leftrightarrow \quad \min_{d \in D} f(d)$$

- Examples of $f(d)$ (Appendix A):
  - **Constrained safety:** $f(d) = -\langle r, d \rangle + \lambda \max\{0, \langle c, d \rangle - \tau\}$.
  - **Entropy-regularized RL:** $f(d) = -\langle r, d \rangle - \alpha H(\pi_d)$.

Fenchel duality and saddle-point form

- Fenchel conjugate: $f^*(y) = \sup_{x \in D}\{\langle y, x\rangle - f(x)\}$.
- Fenchel–Moreau:

$$f(d) = \max_y \{\langle y, d\rangle - f^*(y)\}$$

- Plug into the convex MDP:

$$\min_{d \in D} f(d) = \min_{d \in D} \max_y \{\langle y, d\rangle - f^*(y)\}$$

$$= \min_\pi \max_y L(\pi, y), \quad L(\pi, y) = \langle y, d_\pi\rangle - f^*(y)$$

From convex objective to shaped reward

- For fixed y, minimizing $L(\pi, y)$ over π is equivalent to:

$$\min_{\pi} \langle y, d_\pi \rangle$$

- Using the discounted occupancy identity:

$$\langle y, d_\pi \rangle = (1-\gamma)\,\mathbb{E}_\pi\Big[\sum_{t\geq 0} \gamma^t y(s_t, a_t)\Big]$$

- So the policy step is standard RL with shaped reward:

$$r_y(s,a) = -y(s,a)$$

(or $r_y(s,a) = -\phi(s,a)^\top w$ in FA).

Related work: convex MDPs and general utilities

- **Convex MDPs** (Zahavy et al., 2021):
  - Formalize convex MDPs and the Fenchel dual reduction.
  - Provide a meta-algorithm with policy and cost players.
- **General utilities / occupancy methods**:
  - Recent work characterizes policy gradient methods on utility functions of occupancies.
  - Highlights importance of efficient occupancy estimation in large spaces.

Related work: SPMA and policy-gradient geometry

- Trust-region and mirror-descent methods:
  - TRPO, PPO, MDPO: geometry-aware PG with KL or mirror maps.
- **Softmax Policy Mirror Ascent (SPMA)** (Asad et al., 2024):
  - Mirror ascent in the *logit space* using log-sum-exp mirror map.
  - Tabular update:

$$\pi_{t+1}(a|s) = \pi_t(a|s)\big(1 + \eta A^{\pi_t}(s,a)\big)$$

  - Linear convergence in tabular MDPs, convex projection in FA.

Related work: CMDP primal–dual methods (NPG–PD)

- **NPG–PD** (Ding et al., 2020):
  - Lagrangian: $L(\pi, \lambda) = V_r^\pi(\rho) + \lambda(V_g^\pi(\rho) - b)$.
  - Updates:
    - Natural policy gradient ascent on π.
    - Projected subgradient ascent on λ.
  - Guarantees: $\mathcal{O}(1/\sqrt{T})$ gap and constraint violation, dimension-free.
- Serves as our main CMDP baseline.

Baseline: NPG–PD in our setup

- CMDP Lagrangian (same as Dual–SPMA):

$$L(\pi, \lambda) = J_r(\pi) + \lambda \left(J_c(\pi) - \tau\right).$$

- Primal step: one natural policy gradient step on the shaped reward

$$r_\lambda(s, a) = r(s, a) - \lambda\, c(s, a),$$

  using a diagonal-Fisher NPG update on a softmax actor.

- Dual step: projected ascent on λ:

$$\lambda_{k+1} = \left[\lambda_k + \beta \left(J_c(\pi_k) - \tau\right)\right]_+.$$

- Implementation mirrors Ding et al. (2020):
  - Actor–critic with GAE advantages (same networks as SPMA).
  - No shaped dual variables in the loss; only r_λ.
  - One NPG step per outer iteration.

Dual–SPMA loop: high-level view

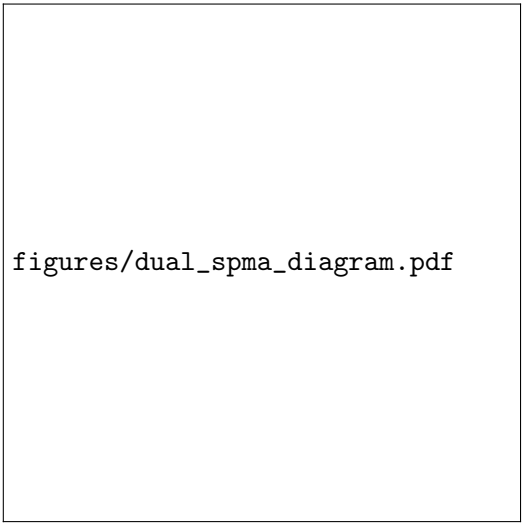
- We consider the saddle problem:

$$\min_{\pi} \max_{y} L(\pi, y) = \langle y, d_\pi \rangle - f^*(y)$$

- **Outer loop (dual):** mirror ascent on y:

$$y_{k+1} = y_k + \alpha(d_{\pi_k} - \nabla f^*(y_k))$$

- **Inner loop (policy):** run K_{in} SPMA steps under reward r_{y_k}.



Policy player: SPMA actor–critic oracle

- Inner loop implemented as SPMA actor–critic:
  - Softmax or Gaussian policy networks.
  - Value critic + GAE advantages.
  - SPMA loss on a batch:

$$\mathcal{L} = \mathbb{E}\big[- \Delta \log \pi \cdot A + \frac{1}{\eta}\big(\exp(\Delta \log \pi) - 1 - \Delta \log \pi \big)\big]$$

- We use Armijo backtracking (plus fallback) to choose the step size.
- Our `PolicyOracleSPMA` wraps:
  - A shaped-reward env ($r_y = -y$ or $-\phi^\top w$).
  - SPMA updates for K_{in} iterations.
  - Occupancy / feature estimators.

Estimating occupancies and feature expectations

- **Tabular estimator** (Appendix C):

$$\hat{d}_\pi(s,a) = \frac{1-\gamma}{N} \sum_{i=1}^{N} \sum_{t} \gamma^t \mathbf{1}\{s_t^{(i)} = s, a_t^{(i)} = a\}$$

- **Feature estimator** (FA):

$$\widehat{\mathbb{E}}[\phi] = \frac{1-\gamma}{N} \sum_{i,t} \gamma^t \phi(s_t^{(i)}, a_t^{(i)})$$

- Both are implemented as simple Monte Carlo estimators with episode counting.
- We verified in tests that $\sum_{s,a} \hat{d}_\pi(s,a) \approx 1$.

Occupancy estimation: MC vs MLE

- Current estimators (Appendix C):
  - Tabular: discounted visit counts $\hat{d}_\pi(s,a)$.
  - FA: Monte Carlo estimate of $\mathbb{E}_\pi[\phi(s,a)]$.

  Variance scales with feature dimension d.
- MLE-style estimator (Barakat et al., 2024):

$$\lambda_\omega(s,a) \propto \exp(\omega^\top \phi(s,a)).$$

  Fit ω by maximizing the (discounted) log-likelihood of (s_t, a_t) samples.
- Guarantees: $\|\hat{\lambda} - \lambda_\pi\|_1 = O\big(\sqrt{m/n}\big)$, depends on parameter dimension m instead of $|S||A|$.
- Implementation:
  - Log-linear model over (s,a) features.
  - Adam updates on the weighted log-likelihood.
  - Returns a normalized occupancy estimate $\hat{d}_\pi(s,a)$.

Outer loop: entropy-regularized RL example

- Entropy-regularized objective (simplified occupancy form):

$$f(d) = -\langle r, d \rangle + \alpha \sum_i d_i \log d_i$$

- Conjugate:

$$f^*(y) = \alpha \log \sum_i \exp\left((y_i + r_i)/\alpha\right)$$

$$\nabla f^*(y) = \mathrm{softmax}\big((y + r)/\alpha\big)$$

- Dual update:

$$y_{k+1} = y_k + \alpha_y \big(\hat{d}_{\pi_k} - \mathrm{softmax}((y_k + r)/\alpha)\big)$$

- Policy step: SPMA under $r_{y_k}(s, a) = -y_k(s, a)$.

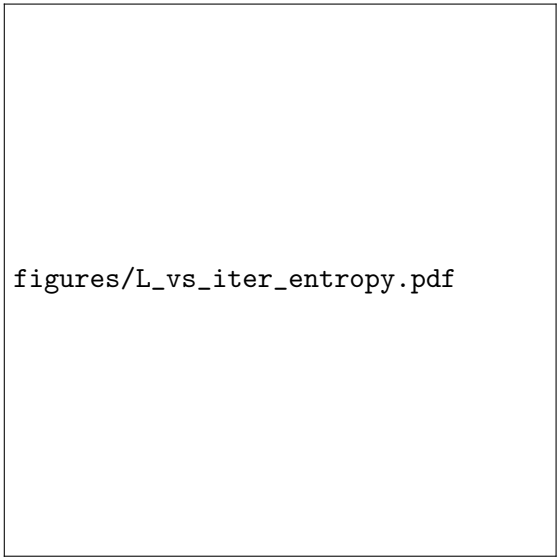
Outer loop: constrained safety via Lagrangian

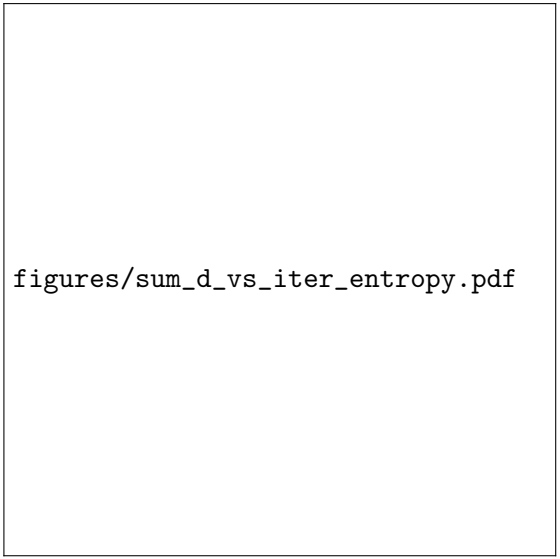
- CMDP: maximize $J_r(\pi)$ subject to $J_c(\pi) \leq \tau$.
- Lagrangian:
$$L(\pi, \lambda) = J_r(\pi) + \lambda(J_c(\pi) - \tau), \quad \lambda \geq 0$$
- Policy update:
$$r_\lambda(s, a) = r(s, a) - \lambda c(s, a)$$
implemented via $y_\lambda(s, a) = \lambda c(s, a) - r(s, a)$, $r_y = -y$.
- Dual update:
$$\lambda_{k+1} = \left[\lambda_k + \beta(J_c(\pi_k) - \tau)\right]_+$$

Experimental setup

- **Environments:**
  - FrozenLake (tabular) with deterministic transitions.
  - Simple cost structure for constrained safety (unsafe states).
  - Pendulum (FA) as a sanity check for feature expectations.
- **Metrics:**
  - Saddle objective $L(\pi, y)$ (when f^* known).
  - Constraint violation $J_c(\pi) - \tau$.
  - $\sum d_\pi$ and $\|E[\phi]\|$ (estimator sanity).
  - Average return under shaped reward.

Results: Dual–SPMA outer loop (entropy example)





Results: constrained safety Dual–SPMA

figures/Jr_Jc_vs_iter_cmdp.pdf

figures/constraint_violation_vs_iter_c

Results: Dual–SPMA vs NPG–PD (CMDP)

figures/Jr_vs_iter_spma_npg.pdf

$J_r(\pi_k)$ vs outer iterations

figures/constraint_violation_vs_iter_s

$J_c(\pi_k) - \tau$ and λ_k vs iterations

Results: occupancy heatmaps

figures/occupancy_heatmaps_gridworld.pdf

Takeaways & limitations

- **Recipe:** convex MDP $\Rightarrow$ Fenchel dual $\Rightarrow$ shaped-reward RL + SPMA.
- SPMA provides a practical policy player with strong convergence properties in tabular and FA settings.
- We implemented:
  - Dual–SPMA outer loops for entropy-regularized RL and constrained safety.
  - A sample-based NPG–PD baseline on the same CMDP tasks.
  - Three occupancy estimators: tabular MC, FA MC, and an MLE-style estimator.
- Limitations:
  - Experiments are still in low-dimensional environments.
  - MLE estimator and FA dual loop not yet stress-tested on large continuous tasks.

Future work

- Extend NPG–PD vs Dual–SPMA comparison to larger CMDPs and function approximation.
- Study empirically how MLE-based occupancy estimation scales in high-dimensional FA.
- Explore more general convex objectives (risk, imitation) with the same dual–SPMA template.
- Investigate variance-reduction techniques for dual gradients in the FA setting.

Questions?