Safe Policy Search for Lifelong Reinforcement Learning with Sublinear Regret

Haitham Bou Ammar
Univ. of Pennsylvania

Rasul Tutunov
Univ. of Pennsylvania

Eric Eaton
Univ. of Pennsylvania
**Problem 1:** Without prior knowledge, RL in a new task is slow

**Idea:** Reuse knowledge from previously learned tasks

We focus on the **lifelong learning** case:

- Agent learns multiple tasks consecutively
- Want a **fully online** method with **sublinear regret**
Problem 2: Robot control policies must obey safety constraints
• Prevent damage to the robot or environment
• Limit joint velocities
• Avoid catastrophic failure

Idea: Incorporate constraints directly into policy optimization
Safe lifelong policy gradient reinforcement learner

- Learns multiple, consecutive RL tasks online
- Operates in an adversarial setting
- Ensures that policies respect given safety constraints
- Exhibits sublinear regret for lifelong policy search
Background: Policy Gradient Methods for Control

- Agent interacts with environment, taking consecutive actions
- PG methods support continuous state and action spaces
  - Have shown recent success in applications to robotic control
    [Kober & Peters 2011; Peters & Schaal 2008; Sutton et al. 2000]

Agent makes sequential decisions
Background: Policy Gradient Methods for Control

- Agent interacts with environment, taking consecutive actions
- PG methods support continuous state and action spaces
  - Have shown recent success in applications to robotic control
    [Kober & Peters 2011; Peters & Schaal 2008; Sutton et al. 2000]

Goal: find policy $\pi_\alpha$ that minimizes

$$ l(\alpha) = \sum_{k=1}^{n} p_\alpha(\tau^{(k)}) C(\tau^{(k)}) $$

- probability of trajectory

$$ p_\alpha(\tau^{(k)}) = p_0(x^{(k)}_0) \prod_{m=0}^{M-1} p(x^{(k)}_{m+1}|x^{(k)}_m, u^{(k)}_m) \pi_\alpha(u^{(k)}_m|x^{(k)}_m) $$

- cost of trajectory

$$ C(\tau^{(k)}) = \frac{1}{M} \sum_{m=0}^{M-1} c^{(k)}_{m+1} $$
Background: Online Learning & Regret Analysis

**Regret Minimization Game:** Each round $j = 1 \ldots R$,

a.) agent chooses a prediction $\alpha_j$, and

b.) environment (i.e., the adversary) chooses a loss function $l_j$

**Goal:** minimize cumulative regret (modified for multi-task case)

$$\mathcal{R}_R = \sum_{j=1}^{R} l_{t_j}(\alpha_j) - \inf_{\theta \in \mathcal{K}} \left[ \sum_{j=1}^{R} l_{t_j}(\theta) \right]$$

- **agent’s total loss**
- **best fixed loss in hindsight**
- **loss of task $t$ at round $j$**

H. Bou Ammar, R. Tutunov, E. Eaton
Lifelong Machine Learning

1.) Tasks are received consecutively
2.) Knowledge is transferred from previously learned tasks
3.) New knowledge is stored for future use
4.) Existing knowledge is refined

πα

previously learned knowledge \( L_{j-1} \)

learned policy

current round \( j \)

future learning rounds

Lifelong Learning System
Task Model

Policy gradient objective: \[ l(\alpha) = \sum_{k=1}^{n} p_{\alpha}(\tau^{(k)}) C(\tau^{(k)}) \]

- For a specific task \( t_j \), find the optimal policy
  \[ \pi_{\alpha^*_t_j}(u | x) \quad \text{s.t.} \quad \alpha^*_t_j = \min_{\alpha} l_{t_j}(\alpha) \]

- The parameters \( \alpha_{t_j} \) are linear combinations of a shared basis \( L \)
  \[ \alpha_{t_j} = Ls_{t_j} \quad L \in \mathbb{R}^{d \times k}, \ s_{t_j} \in \mathbb{R}^k \]
Each task $t_j$ has associated safety constraints $(A_{t_j}, b_{t_j})$ such that $A_{t_j} \alpha_{t_j} \leq b_{t_j}$
Each round, we observe $n_{t_j}$ trajectories of task $t_j$

**Goal:** minimize total cumulative loss-so-far

**Online Multi-task Objective:** After observing $r$ rounds,

$$\min_{L,S} \sum_{j=1}^r \left[ \eta_{t_j} l_{t_j} (L s_{t_j}) \right] + \mu_1 \|S\|_F^2 + \mu_2 \|L\|_F^2$$

- loss for task $t_j$
- regularize projections and shared repository

s.t. $A_{t_j} \alpha_{t_j} \leq b_{t_j}$ \forall t_j \in \mathcal{I}_r$

$$\lambda_{\min}(LL^T) \geq p \text{ and } \lambda_{\max}(LL^T) \leq q$$

- ensure “informative” policies by bounding $\|L\|_F$
Online Formulation

Online MTL Objective

\[
\min_{L, S} \sum_{j=1}^{r} \left[ \eta_{t_j} l_{t_j} \left( \mathbf{L} s_{t_j} \right) \right] + \mu_1 \| S \|_F^2 + \mu_2 \| L \|_F^2 \\
\text{s.t. } \mathbf{A}_{t_j} \alpha_{t_j} \leq \mathbf{b}_{t_j} \quad \forall t_j \in \mathcal{I}_r \\
\lambda_{\text{min}}(LL^T) \geq p \text{ and } \lambda_{\text{max}}(LL^T) \leq q
\]

Let \( \theta = [\text{vec}(\mathbf{L}), \text{vec}(\mathbf{S})]^T \)

We can re-write the objective as:

\[
\theta_{r+1} = \arg \min_{\theta \in \mathcal{K}} \Omega_r(\theta) \quad \Omega_0(\theta) = \mu_2 \sum_{i=1}^{dk} \theta_i^2 + \mu_1 \sum_{i=1}^{dk+1} \theta_i^2 \\
\Omega_j(\theta) = \Omega_{j-1}(\theta) + \eta_{t_j} l_{t_j}(\theta)
\]

set of safe policies

H. Bou Ammar, R. Tutunov, E. Eaton
Solution Strategy

**Step 1: Unconstrained Solution**

a.) Update $L$, holding $S$ fixed

$$L_{β+1} = L_β - \eta_β \nabla_L e_r(L, S)$$

b.) Update $S$, holding $L$ fixed

$$s_{λ+1}^{(t_j)} = s_λ^{(t_j)} - \eta_λ \nabla_S e_r(L, S)$$

$$\tilde{θ}_{r+1}$$ unconstrained solution

**Step 2: Constrained Solution**

**Idea:** Alternate to learn projection of $\tilde{θ}_{r+1}$ onto the constraint set

**Problem:** Computationally Expensive
Constrained Projection Learning

Learning the constrained solution is equivalent to:

$$\hat{\theta}_{r+1} = \arg \min_{\theta \in \mathcal{K}} \mathcal{B}_{\Omega_r, \mathcal{K}} \left( \theta, \tilde{\theta}_{r+1} \right)$$

Bregman divergence

Reduce computational complexity by linearizing losses

$$l_{t_r}(\hat{u}) = \hat{f}_{t_r} \bigg|_{\hat{\theta}_r}^T \hat{u}$$

linearized loss around constrained solution to previous round

$$\hat{f}_{t_r} \bigg|_{\hat{\theta}_r} = \left[ \nabla_{\theta} l_{t_r}(\theta) \bigg|_{\hat{\theta}_r}, l_{t_r}(\theta) \bigg|_{\hat{\theta}_r} - \nabla_{\theta} l_{t_r}(\theta) \bigg|_{\hat{\theta}_r, \hat{\theta}_r} \right]^T$$

first-order term

H. Bou Ammar, R. Tutunov, E. Eaton
Constrained Projection Learning

Using linearized losses, the constrained solution simplifies to:

\[
\hat{\theta}_{r+1} = \arg \min_{\theta \in \mathcal{K}} B_{\Omega_0, \mathcal{K}} (\theta, \tilde{\theta}_{r+1})
\]

Constrained Problem for Determining Safe Policies

\[
\begin{align*}
\min_{L, S} & \mu_1 \|S\|_F^2 + \mu_2 \|L\|_F^2 + 2\mu_1 \text{tr} \left( S \tilde{\theta}_{r+1}^T S \right) + 2\mu_2 \text{tr} \left( L \tilde{\theta}_{r+1}^T L \right) \\
\text{s.t.} & \quad A_{t_j} L \alpha_{t_j} \leq b_{t_j} \quad \forall t_j \in \mathcal{I}_r \\
& \quad \lambda_{\min}(LL^T) \geq p \quad \text{and} \quad \lambda_{\max}(LL^T) \leq q
\end{align*}
\]

Solved via (1) a 2nd order cone program for \( S \) and (2) a semi-definite program for \( L \)
**Theorem** (Sublinear Regret):
After $R$ rounds, our algorithm attains sublinear regret:
\[
\sum_{j=1}^{R} l_{t_j}(\hat{\theta}_j) - l_{t_j}(u) = \mathcal{O}(\sqrt{R}) \quad \text{for any } u \in \mathcal{K}
\]

**Proof Sketch:**
Bound \[\left\| \hat{f}_{t_r} \right\|_{2}^{*} \]
\[
\left\| \hat{f}_{t_r} \right\|_{2} \leq \left\| l_{t_r}(\theta) \right\|_{\hat{\theta}_r} + \left\| \nabla_{\theta} l_{t_r}(\theta) \right\|_{\hat{\theta}_r} + \left\| \nabla_{\theta} l_{t_r}(\theta) \right\|_{\hat{\theta}_r} \]

1. **constant**
2. **bounded in terms of local losses**
3. **constraints**
**Experiments**

**Goal:** Learn policies for consecutive control tasks on three types of dynamical systems

- **Simple Mass**
- **Cart Pole**
- **Quadro rotor**

Generated 10 tasks per system by varying specifications

Compared to (1) standard PG and

(2) PG-ELLA lifelong learner [Bou Ammar et al, ICML’14]
6.3. Application to Quadrotor Control

We generated 10 different quadrotor systems by varying the inertia around the x, y and z-axes. We used a linear model to control the four rotational velocities to initialize the policies in both the learning and testing state, and the lengths of the rods supporting the rotors. Although the overall state of the system can be described by the rotor's speed affects the overall variation of the system's constraint as discussed above to update the models.

Figure 2 shows the performance of the unconstrained solution (e) and demonstrates that our approach abides by the constraints (the dashed black region).

Figure 3 shows the performance of Safe Online 10 Iterations, Safe Online 50 Iterations, Safe Online 100 Iterations, Standard PG, PG-ELLA, and Safe PG 100 Iterations for each task (i.e., Quadrotor, Simple Mass, Cart Pole) for each of the ro- tors to stabilize the system. To ensure realistic dynamics, we used the simulated model described by (Bouabdallah et al., 2007), which has been verified to efficiently updated over time.

We also applied our approach to the more challenging domain of quadrotor control. The dynamics of the quadrotor system (Figure 6.3) and demonstrating that our approach abides by the constraints (the dashed black region).

We described the first lifelong PG learner that provides sublinear regret vs 1000, 2000, 4000, 6000, 8000, 10000 rounds, and Safe Online 10 Iterations, Safe Online 50 Iterations, Safe Online 100 Iterations, Standard PG, PG-ELLA, Safe Policy Search for Lifelong Reinforcement Learning with Sublinear Regret, illustrating that our approach outperforms standard PG and PG-ELLA.

The results also evaluated constrained tasks in a similar manner, again comparing against standard PG and PG-ELLA. Figures (c) and (d) examine over consecutive unconstrained tasks, showing that our approach outperforms standard PG and PG-ELLA. Figures (c) and (d) examine.

Safe lifelong learner shows superior performance
Results: Safety Constraint Enforcement

Enforces safety constraints, unlike alternative methods
Our approach immediately projects policies to safe regions, even during the policy search process.
Teaser: Autonomous Cross-Domain Transfer

**Key Idea:** Use projections to specialize a shared KB to individual task domains for lifelong RL

[Bou Ammar, Eaton, et al., IJCAI’15]
Conclusion

The safe lifelong policy gradient learner:

• Fully online learning of multiple, consecutive RL tasks
• Ensures “safe” policies by respecting safety constraints
• Exhibits *sublinear regret* for lifelong policy search
• Validated on benchmark dynamical systems and quadrotor control
Thank you!

Questions?

Haitham Bou Ammar
haithamb@seas.upenn.edu

Rasul Tutunov
tutunov@seas.upenn.edu

Eric Eaton
eeaton@cis.upenn.edu

This research was supported by ONR grant #N00014-11-1-0139 and AFRL grant #FA8750-14-1-0069.

We thank Ali Jadbabaie for assistance with the optimization solution, and the anonymous reviewers for their helpful feedback.
Backup Slides
Constrained Solution

Alternate to determine safety-constrained $L$ and $S$:

**Semi-Definite Program for $L$:**

$$\min_{X \in S_{++}} \mu_2 \text{trace}(X) + 2\mu_2 \left| \left| L \right|_{\tilde{\theta}_{r+1}} \right|_F \sqrt{\text{trace}(X)}$$

s.t. $s_{t_j}^T X s_{t_j} = a_{t_j}^T a_{t_j} \quad \forall t_j \in \mathcal{I}_r$

$X \preceq pI$ and $X \succeq qI$, with $X = L^T L$

**Second-Order Cone Program for $S$:**

$$\min_{s_{t_1}, \ldots, s_{t_j}, c_{t_1}, \ldots, c_{t_j}} \mu_1 \sum_{j=1}^{r} \left| \left| s_{t_j} \right|_2 \right|_2 + 2\mu_1 \sum_{j=1}^{r} s_{t_j}^T \left| \left| \hat{\theta}_r \right| s_{t_j} \right|_F$$

s.t. $A_{t_j} L s_{t_j} = b_{t_j} - c_{t_j}$

$c_{t_j} > 0 \quad \left| \left| c_{t_j} \right|_2 \right|_2 \leq c_{\max}^2 \quad \forall t_j \in \mathcal{I}_r$.