Logarithmic Regret for Online Control

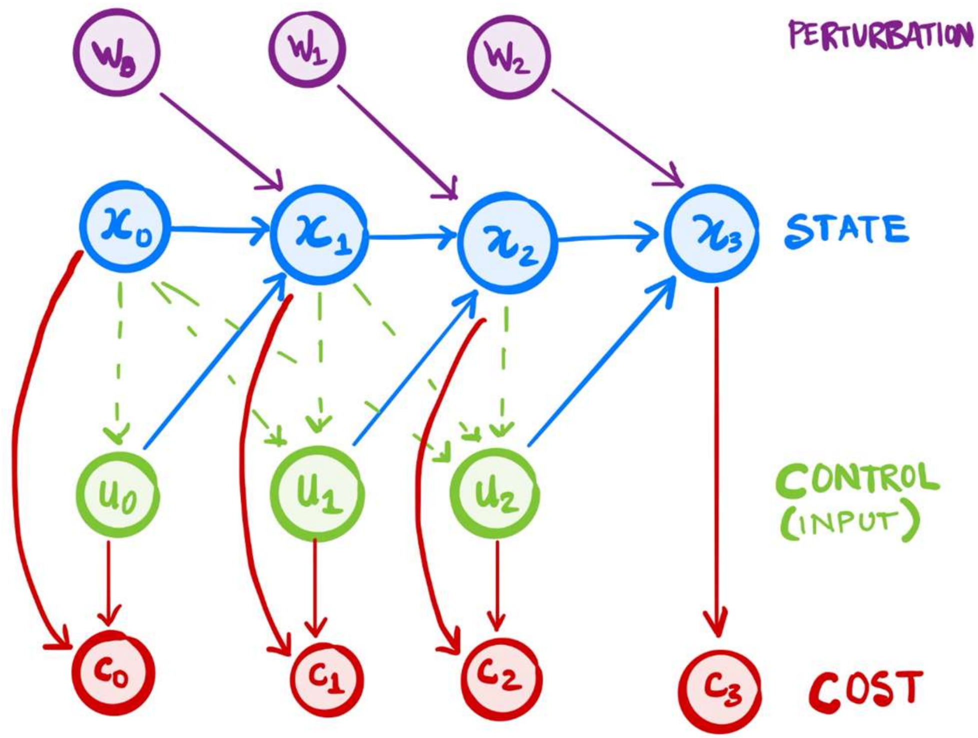
NeurIPS 2019

Naman Agarwal, Elad Hazan, Karan Singh

Controlling Dynamical Systems

$$\min_{u(x)} \; \mathbb{E}\left[\sum_{t=1}^{T} c(x_t, u_t)\right]$$

$$\text{s.t.} \quad x_{t+1} = f(x_t, u_t, w_t)$$



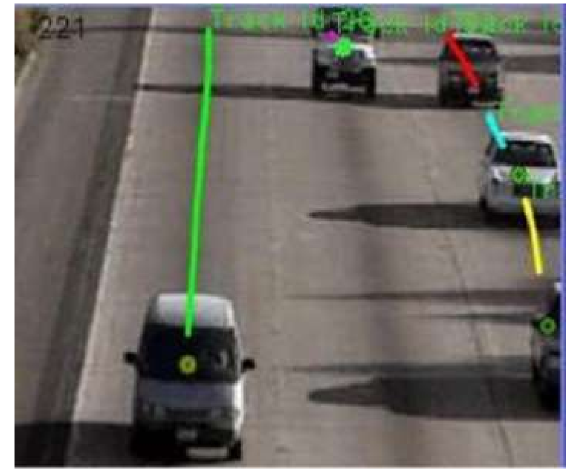
x_t is the state.

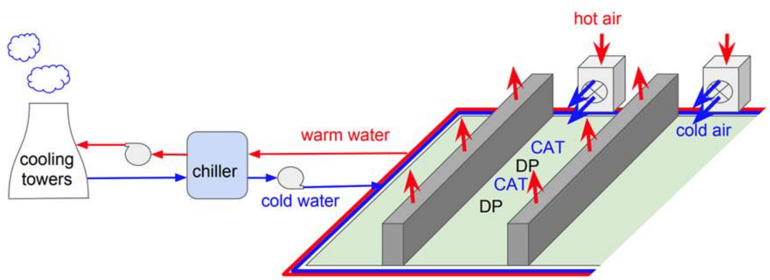
u_t is the <u>control</u> input.

w_t is the perturbation sequence.

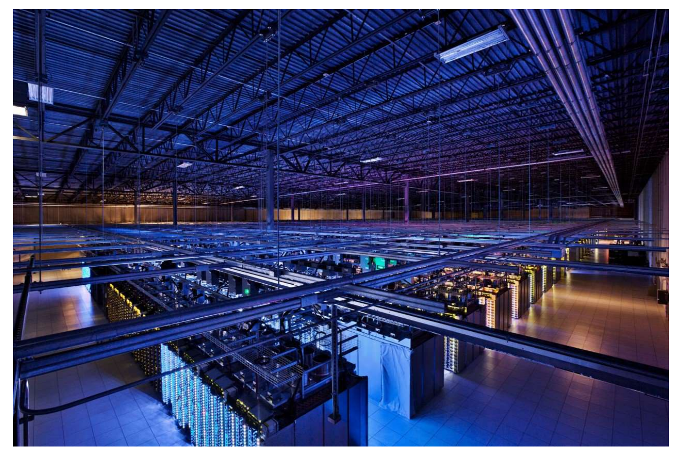
Success Stories

- Robotics
- Autonomous Vehicles
- Physical systems





[Cohen et al ‘18]

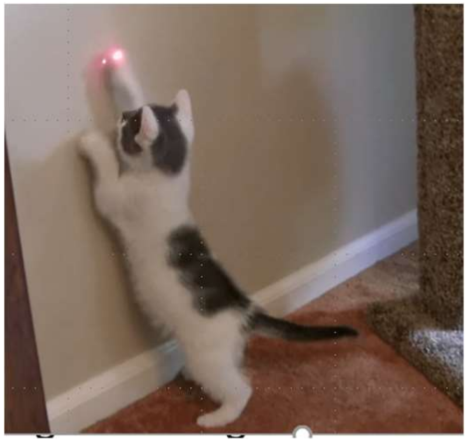


Setting: Changing Costs

- Control a noisy Linear Dynamical System with **Changing Convex Costs**

$$x_{t+1} = Ax_t + Bu_t + w_t; \qquad w_t \sim N(0, \Sigma)$$

Strongly Convex: $c_t(x_t, u_t)$



- Generalizes the classical **Tracking** problem
  - $c_t(x_t, u_t) = |x_t - x_t^*|^2 + |u_t|^2$
  - x_t^* = state sequence of exogenous target

Goal: Low-regret Control

- Goal: **POLICY REGRET** (compete with “what would have happened”)

$$\max_{w_{1:T}} \left(\sum_{t=1}^{T} c_t(x_t, u_t) - \min_{K} \sum_{t=1}^{T} c_t(\hat{x}_t, K\hat{x}_t) \right)$$

- The comparator K has the complete foreknowledge of $c_{1:T}$.
- $\hat{x}_t$ = **counterfactual state sequence** under

$$\hat{u}_t = K\hat{x}_t, \quad \hat{x}_{t+1} = A\hat{x}_t + B\hat{u}_t + w_t$$

Main Result

Efficient online algorithm s.t.

$$\sum_{t=1}^{T} c_t(x_t, u_t) \; - \; \min_{K \in stable} \sum_{t=1}^{T} c_t(\hat{x}_t, K\hat{x}_t) \le O(\mathrm{polylog}\, T)$$

- **First logarithmic** regret (fast rate) for control with changing costs.
  - Efficient → Polynomial in system parameters, logarithmic in T
  - K is stable if $\rho(A + BK) < 1$.

- DP-based approach [Abbasi-Yadkori et al 2014, Cohen et al 2018] yields $\sqrt{T}$.

Ingredient 1: Error Feedback Policy

- Even if $c_{1:T}$ is known, computing optimal K is a non-convex problem
- Linear Policy (K):

$$u_{t+1}(K) = Kx_{t+1} = K \cdot \left(\sum_{i=0}^{t} (A + BK)^i w_{t-i} \right)$$

- Relaxation ($\vec{M} = \{M_1 \ \dots M_t\}$):

$$\min_{M} \left(\sum_{t=1}^{T} c_t \left(x_t \left(\vec{M} \right), u_t \left(\vec{M} \right) \right) \right)$$

is convex!

$$u_{t+1}\left(\vec{M}\right) = \overrightarrow{M_t} \cdot \overrightarrow{w_t} = \left(\sum_{i=0}^{t} M_i w_{t-i} \right)$$

Ingredient 2: Enforcing stability

- Let K be any fixed stabilizing linear policy (determined completely from A, B).
- Choose optimal controls as:

$$u_t = Kx_t + \sum_{i=1}^{H} M_i w_{t-i}$$

- **Representational Power:** With $H \approx \log T$, can emulate any strongly stable policy.
- **Stability:** K is stable $\Rightarrow$ any (even non-stationary) error feedback policy is stable.

Ingredient 3: OCO with memory

- Adversarial sequence with time dependency:

$$c(x_t, u_t) = f_t\left(\vec{M}^t, \vec{M}^{t-1}, \dots, \vec{M}^{t-H}\right)$$

- Regret vs. best fixed decision

$$\sum_{t=1}^{T} f_t\left(\vec{M}^t, \vec{M}^{t-1}, \dots, \vec{M}^{t-H}\right) - \min_{\vec{M}} \sum_t f_t\left(\vec{M}, \vec{M}, \dots, \vec{M}\right) = O(H\sqrt{T})$$

- Efficient algorithms that guarantee low regret [Anava et al 2013, Even-Dar et al 2009]

The Online Algorithm

Initialize matrices $\vec{M} = M_1, \ldots, M_H$
For $t = 1,2, \ldots, T$ do

1. Use $u_t = Kx_t + \sum_{i \le H} M_i \, w_{t-i}$.

2. Observe state x_{t+1}, estimate $w_{t+1} = x_{t+1} - Ax_t - Bu_t$.

3. Construct the "counterfactual" cost function:

$$\ell_t\left(\vec{M}\right) = c\left(x_t\left(\vec{M}\right), u_t\left(\vec{M}\right)\right)$$

4. Update $\vec{M}$.

$$\vec{M} \leftarrow \vec{M} - \eta \, \nabla_{\vec{M}} \, \ell_t\left(\vec{M}\right)$$

Key Challenge

- **Previous Reduction:**

$$c_t\left(x_t\left(\vec{M}\right), u_t\left(\vec{M}\right)\right), \text{ cost}_t \text{ of error feedback policy.}$$

- Strongly convex $c_t(x, u)$ $\neq$ Strongly convex cost_t(policy).

- **Challenge:** <u>overparameterization</u>!

$$K \text{ is a } |X| \times |U| \text{ matrix.}$$

$$\vec{M} \text{ has } |X| \times |U| \times \log T \text{ parameters.}$$

Key Observation

- **<u>Theorem:</u>** $c_t\left(x_t\left(\vec{M}\right), u_t\left(\vec{M}\right)\right)$ is strongly convex in $\vec{M}$.

- Say f is 1-strongly convex.
  $f(Ax)$ is α-strongly convex (in x) iff $AA^T \succcurlyeq \alpha I$.

- <u>1-d case:</u>

$$\mathbb{E}\left[\left(\frac{dx_t}{d\vec{M}}\right)\left(\frac{dx_t}{d\vec{M}}\right)^T\right] \approx \begin{bmatrix} S_{\gamma^2}(H) & \gamma S_{\gamma^2}(H-1) & \gamma^2 S_{\gamma^2}(H-1) & \dots & \gamma^{H-1} S_{\gamma^2}(1) \\ \gamma S_{\gamma^2}(H-1) & S_{\gamma^2}(H) & \gamma S_{\gamma^2}(H-1) & \dots & \gamma^{H-2} S_{\gamma^2}(2) \\ \vdots & \vdots & \vdots & \dots & \vdots \\ \gamma^{H-1} S_{\gamma^2}(1) & \gamma^{H-2} S_{\gamma^2}(2) & \dots & \gamma^{H-1} S_{\gamma^2}(1) & S_{\gamma^2}(H) \end{bmatrix}$$

$$\succcurlyeq \Omega(\gamma, H^{-1}) I$$

Summary: Efficient *gradient-based* **control** algorithms achieving $\log T$ regret for LDS with changing costs, a generalization of the tracking problem.

Thank you !

Naman Agarwal

Elad Hazan