Differentially Private LQG

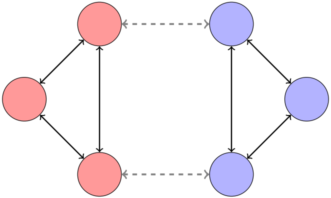
Matthew Hale

Department of Mechanical and Aerospace Engineering
University of Florida

AFOSR Center of Excellence on Assured Autonomy in Contested Environments
October 15, 2019

Goal: Share Information Without Oversharing

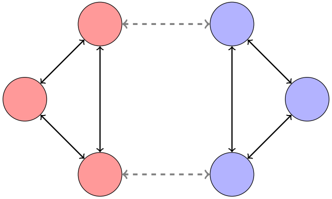
- In coalitions, we want to collaborate while keeping secrets



- To work together, red and blue must exchange information

Goal: Share Information Without Oversharing

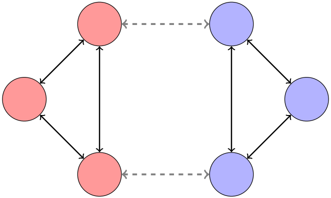
- In coalitions, we want to collaborate while keeping secrets



- To work together, red and blue must exchange information
- Agents must protect states from eavesdroppers *and* the other team

Goal: Share Information Without Oversharing

- In coalitions, we want to collaborate while keeping secrets



- To work together, red and blue must exchange information
- Agents must protect states from eavesdroppers *and* the other team

Fundamental Problem

How can agents safeguard state trajectories and still collaborate?

How should we provide privacy?

How should we provide privacy?

Differential Privacy (DP)

DP is a privacy framework with a several key features:

- It offers a formal definition of “privacy”

Differential Privacy (DP)

DP is a privacy framework with a several key features:

- It offers a formal definition of "privacy"
- It is immune to post-processing
  - x private $\Rightarrow f(x)$ private for all f

How should we provide privacy?

Differential Privacy (DP)

DP is a privacy framework with a several key features:

- It offers a formal definition of "privacy"
- It is immune to post-processing
  - x private $\Rightarrow f(x)$ private for all f
- It is robust to side information

How should we provide privacy?

Differential Privacy (DP)

DP is a privacy framework with a several key features:

- It offers a formal definition of "privacy"
- It is immune to post-processing
  - x private $\Rightarrow f(x)$ private for all f
- It is robust to side information

- Used by: Apple Google Uber

UBER

How should we provide privacy?

Differential Privacy (DP)

DP is a privacy framework with a several key features:

- It offers a formal definition of "privacy"
- It is immune to post-processing
  - x private $\Rightarrow f(x)$ private for all f
- It is robust to side information

- Used by: Apple Google Uber

UBER

DP Idea

Make "adjacent" state trajectories produce "similar" outputs

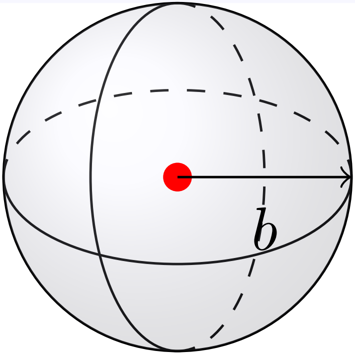
UF UNIVERSITY of FLORIDA Duke UNIVERSITY TEXAS The University of Texas at Austin UC SANTA CRUZ

Adjacency Specifies What to Protect

Adjacent trajectories in ℓ_p-spaces

We fix a constant $b > 0$ and define $\text{Adj}_b : \ell_p^n \times \ell_p^n \to \{0, 1\}$ as

$$\text{Adj}_b(x_1, x_2) = 1 \iff \|x_1 - x_2\|_{\ell_p} \leq b.$$

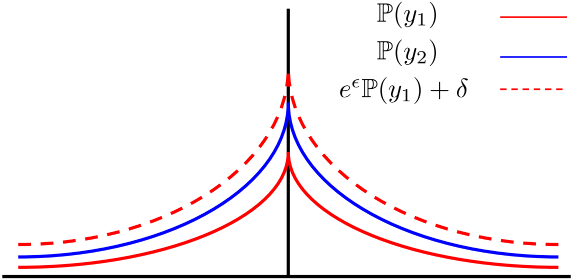


Differential Privacy is a Statistical Guarantee

Fundamental Inequality of Differential Privacy

For adjacent state trajectories x_1 and x_2, we want the outputs y_1, y_2 to satisfy

$$\mathbb{P}(y_2) \leq e^{\epsilon}\mathbb{P}(y_1) + \delta,$$



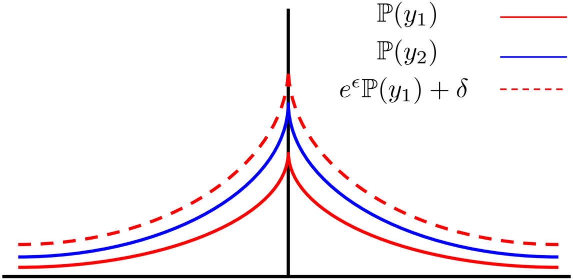
Differential Privacy is a Statistical Guarantee

Fundamental Inequality of Differential Privacy

For adjacent state trajectories x_1 and x_2, we want the outputs y_1, y_2 to satisfy

$$\mathbb{P}(y_2) \leq e^{\epsilon}\mathbb{P}(y_1) + \delta,$$

This is the definition of (ϵ, δ)-differential privacy.



Mechanisms for Differential Privacy

- Fix a probability space $(\Omega, \Sigma, \mathbb{P})$. Differential privacy is enforced by a *mechanism* of the form

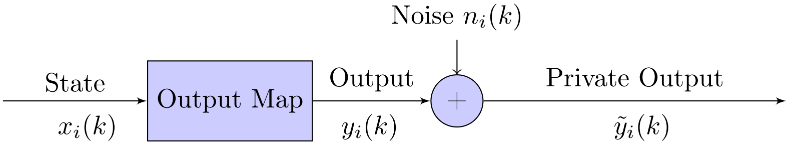
$$M : \ell_p^n \times \Omega \to \ell_q^r.$$

Mechanisms for Differential Privacy

- Fix a probability space $(\Omega, \Sigma, \mathbb{P})$. Differential privacy is enforced by a *mechanism* of the form

$$M : \ell_p^n \times \Omega \to \ell_q^r.$$

- For us this will take the form



Formalizing Multi-Agent LQ

- Consider problems with N agents
- Agent i has the update and output maps

$$
\begin{aligned}
x_i(k+1) &= A_i x_i(k) + B_i u_i(k) \\
y_i(k) &= C_i x_i(k)
\end{aligned}
$$

Formalizing Multi-Agent LQ

- Consider problems with N agents
- Agent i has the update and output maps

$$
\begin{aligned}
x_i(k+1) &= A_i x_i(k) + B_i u_i(k) + w_i(k) \\
y_i(k) &= C_i x_i(k) + v_i(k),
\end{aligned}
$$

where $w_i(k) \sim \mathcal{N}(0, W_i)$, $v_i(k) \sim \mathcal{N}(0, V_i)$

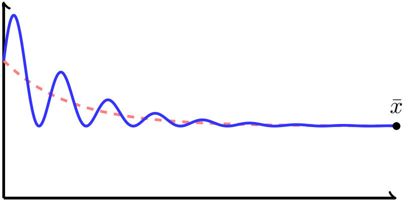
Formalizing Multi-Agent LQ

- Consider problems with N agents
- Agent i has the update and output maps

$$x_i(k+1) = A_i x_i(k) + B_i u_i(k) + w_i(k)$$
$$y_i(k) = C_i x_i(k) + v_i(k),$$

where $w_i(k) \sim \mathcal{N}(0, W_i)$, $v_i(k) \sim \mathcal{N}(0, V_i)$

- Agent i wants to track $\{\bar{x}_i(k)\}_{k \in \mathbb{N}}$



Generating Optimal LQG Control Values

- We want to minimize the quadratic cost

$$J = \lim_{T_f \to \infty} \frac{1}{T_f} \mathbb{E} \left[\sum_{k=1}^{T_f} \underbrace{\big(x(k) - \bar{x}(k)\big)^T Q \big(x(k) - \bar{x}(k)\big)}_{\text{Tracking error}} + \underbrace{u(k)^T R u(k)}_{\text{Control energy}} \right]$$

Generating Optimal LQG Control Values

- We want to minimize the quadratic cost

$$J = \lim_{T_f \to \infty} \frac{1}{T_f} \mathbb{E} \left[\sum_{k=1}^{T_f} \underbrace{\big(x(k) - \bar{x}(k)\big)^T Q \big(x(k) - \bar{x}(k)\big)}_{\text{Tracking error}} + \underbrace{u(k)^T R u(k)}_{\text{Control energy}} \right]$$

- Subject to the linear dynamics

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + w(k) \\ y(k) &= Cx(k) + v(k) \end{aligned}$$

Generating Optimal LQG Control Values

- We want to minimize the quadratic cost

$$J = \lim_{T_f \to \infty} \frac{1}{T_f} \mathbb{E}\left[\sum_{k=1}^{T_f} \underbrace{\big(x(k) - \bar{x}(k)\big)^T Q \big(x(k) - \bar{x}(k)\big)}_{\text{Tracking error}} + \underbrace{u(k)^T R u(k)}_{\text{Control energy}} \right]$$

- Subject to the linear dynamics

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + w(k) \\ y(k) &= Cx(k) + v(k) \end{aligned}$$

- Solution is

$$u^*(k) = L\mathbb{E}[x(k)] + Mg$$

for known M, L, and g

Agents Must Share State Information

- Agent i computes

$$u_i^*(k) = \Big(L\mathbb{E}[x(k)]\Big)_i + \Big(Mg\Big)_i$$

- Computing $\mathbb{E}[x(k)]$ can be done with a Kalman filter, but requires agents to share states

Agents Must Share State Information

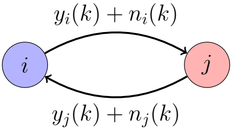
- Agent i computes

$$u_i^*(k) = \Big(L\mathbb{E}[x(k)]\Big)_i + \Big(Mg\Big)_i$$

- Computing $\mathbb{E}[x(k)]$ can be done with a Kalman filter, but requires agents to share states

- Agent i privatizes its own transmissions by sending

$$\tilde{y}_i(k) = C_i x_i(k) + v_i(k) + n_i(k)$$



Computing g also Requires Privacy

- Agents also need to compute

$$g = NQ\bar{x},$$

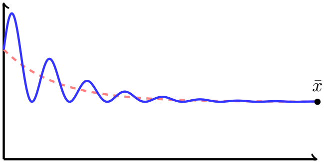
but $\bar{x}$ is very sensitive!

Computing g also Requires Privacy

- Agents also need to compute

$$g = NQ\bar{x},$$

but $\bar{x}$ is very sensitive!

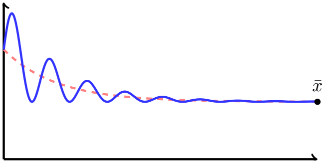


Computing g also Requires Privacy

- Agents also need to compute

$$g = NQ\bar{x},$$

but $\bar{x}$ is very sensitive!

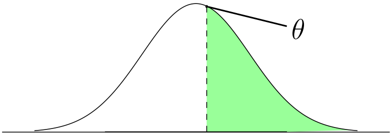


- Agent i instead shares $\tilde{x}_i := \bar{x}_i + \bar{w}_i$
- Then agent i computes

$$u_i^*(k) = \Big(L\mathbb{E}[x(k) \mid \tilde{y}(k)]\Big)_i + \Big(MNQ\tilde{x}\Big)_i$$

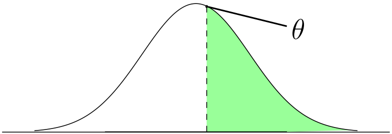
When is this private?

- Need the $\mathcal{Q}$-function: $\mathcal{Q}(\theta) = \frac{1}{\sqrt{2\pi}} \int_{\theta}^{\infty} e^{-\frac{z^2}{2}} dz$



When is this private?

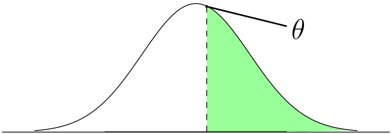
- Need the $\mathcal{Q}$-function: $\mathcal{Q}(\theta) = \frac{1}{\sqrt{2\pi}} \int_{\theta}^{\infty} e^{-\frac{z^2}{2}} dz$



- Define $K_\delta = \mathcal{Q}^{-1}(\delta)$ and $\kappa(\delta, \epsilon) = \frac{1}{2\epsilon}\left(K_\delta + \sqrt{K_\delta^2 + 2\epsilon}\right)$

When is this private?

- Need the $\mathcal{Q}$-function: $\mathcal{Q}(\theta) = \frac{1}{\sqrt{2\pi}} \int_\theta^\infty e^{-\frac{z^2}{2}} dz$



- Define $K_\delta = \mathcal{Q}^{-1}(\delta)$ and $\kappa(\delta, \epsilon) = \frac{1}{2\epsilon}\left(K_\delta + \sqrt{K_\delta^2 + 2\epsilon}\right)$

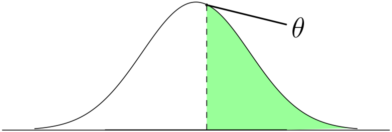
Theorem: Multi-Agent LQ Privacy

Agent i uses $\epsilon_i > 0$, $\delta_i \in (0, 1/2)$. Agent i attains (ϵ_i, δ_i)-privacy if:

i. $\tilde{x}_i := \bar{x}_i + \bar{w}_i$ has $\bar{w}_i \sim \mathcal{N}\big(0, \kappa(\delta_i, \epsilon_i) b_i\big)$

When is this private?

- Need the $\mathcal{Q}$-function: $\mathcal{Q}(\theta) = \frac{1}{\sqrt{2\pi}} \int_{\theta}^{\infty} e^{-\frac{z^2}{2}} dz$



- Define $K_{\delta} = \mathcal{Q}^{-1}(\delta)$ and $\kappa(\delta, \epsilon) = \frac{1}{2\epsilon}\left(K_{\delta} + \sqrt{K_{\delta}^2 + 2\epsilon}\right)$

Theorem: Multi-Agent LQ Privacy

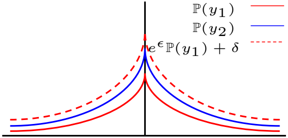
Agent i uses $\epsilon_i > 0$, $\delta_i \in (0, 1/2)$. Agent i attains (ϵ_i, δ_i)-privacy if:

i. $\tilde{x}_i := \bar{x}_i + \bar{w}_i$ has $\bar{w}_i \sim \mathcal{N}\big(0, \kappa(\delta_i, \epsilon_i) b_i\big)$

ii. $\tilde{y}_i(k) := y(k) + n_i(k)$ has $n_i(k) \sim \mathcal{N}\big(0, \kappa(\delta_i, \epsilon_i) b_i \sqrt{\lambda_{max}(C_i)}\big)$

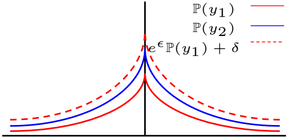
What does privacy reveal?

- Privacy's guarantees are only about information



What does privacy reveal?

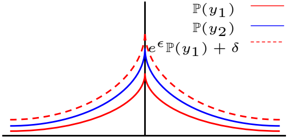
- Privacy's guarantees are only about information



- Its impact is often stated in terms of only ϵ and δ

What does privacy reveal?

- Privacy's guarantees are only about information



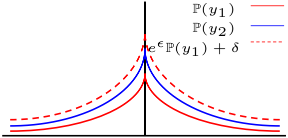
- Its impact is often stated in terms of only ϵ and δ

Questions in Private Control

1. How does privacy affect control performance?

What does privacy reveal?

- Privacy's guarantees are only about information



- Its impact is often stated in terms of only ϵ and δ

Questions in Private Control

1. How does privacy affect control performance?
2. What are the tradeoffs between them?

How do we calibrate privacy?

How do we calibrate privacy?

- Quantifying the increase in J gives a natural control-theoretic cost of privacy to use for privacy calibration

How do we calibrate privacy?

- Quantifying the increase in J gives a natural control-theoretic cost of privacy to use for privacy calibration

Theorem: Cost of Privacy

The cost of privatizing LQG is

$$\Delta J(\epsilon, \delta) = \mathrm{tr}\big(M_1 \Sigma + M_2 \bar{\Sigma}\big) - \mathrm{tr}(M_3) + \mathrm{tr}(M_4 \bar{W})$$

How do we calibrate privacy?

- Quantifying the increase in J gives a natural control-theoretic cost of privacy to use for privacy calibration

Theorem: Cost of Privacy

The cost of privatizing LQG is

$$\Delta J(\epsilon, \delta) = \text{tr}\big(M_1 \Sigma + M_2 \bar{\Sigma}\big) - \text{tr}(M_3) + \text{tr}(M_4 \bar{W})$$

- $\bar{W}$ is the covariance of privacy noise for $\bar{x}$

- Quantifying the increase in J gives a natural control-theoretic cost of privacy to use for privacy calibration

Theorem: Cost of Privacy

The cost of privatizing LQG is

$$\Delta J(\epsilon, \delta) = \text{tr}\left(M_1\Sigma + M_2\bar{\Sigma}\right) - \text{tr}(M_3) + \text{tr}(M_4\bar{W})$$

- $\bar{W}$ is the covariance of privacy noise for $\bar{x}$
- Σ solves the ARE

$$\Sigma = A\Sigma A^T - A\Sigma C^T\left(C\Sigma C^T + V(\epsilon, \delta)\right)^{-1}C\Sigma A^T + W$$

How do we calibrate privacy?

- Quantifying the increase in J gives a natural control-theoretic cost of privacy to use for privacy calibration

Theorem: Cost of Privacy

The cost of privatizing LQG is

$$\Delta J(\epsilon, \delta) = \text{tr}\big(M_1 \Sigma + M_2 \bar{\Sigma}\big) - \text{tr}(M_3) + \text{tr}(M_4 \bar{W})$$

- $\bar{W}$ is the covariance of privacy noise for $\bar{x}$
- Σ solves the ARE

$$\Sigma = A\Sigma A^T - A\Sigma C^T \Big(C\Sigma C^T + V(\epsilon, \delta)\Big)^{-1} C\Sigma A^T + W$$

- $\bar{\Sigma}$ is computed via

$$\bar{\Sigma} = \Sigma - \Sigma C^T (C\Sigma C^T + V)^{-1} C\Sigma$$

Can I relax privacy for better performance?

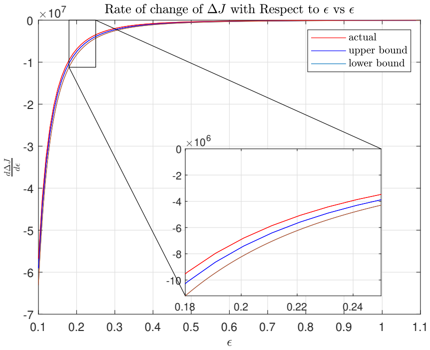
- A privacy rule of thumb is that "all small epsilons are alike"

Can I relax privacy for better performance?

- A privacy rule of thumb is that "all small epsilons are alike"
- Slightly reducing privacy doesn't reveal much more, can save on cost

Can I relax privacy for better performance?

- A privacy rule of thumb is that “all small epsilons are alike”
- Slightly reducing privacy doesn’t reveal much more, can save on cost



- Across many problems, increasing any $\epsilon < 0.5$ leads to substantial reductions in cost