Linear Quadratic Control from an Optimization Viewpoint

Yujie Tang

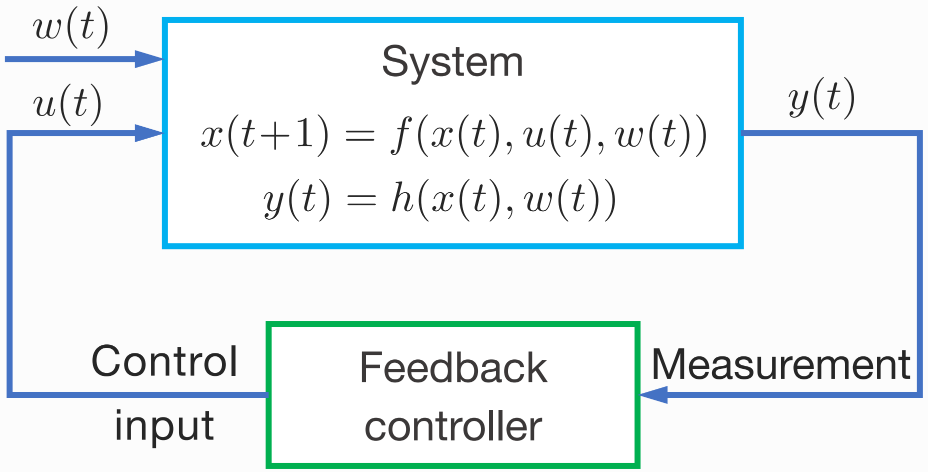
Yang Zheng · Yingying Li · Runyu Zhang · Na Li

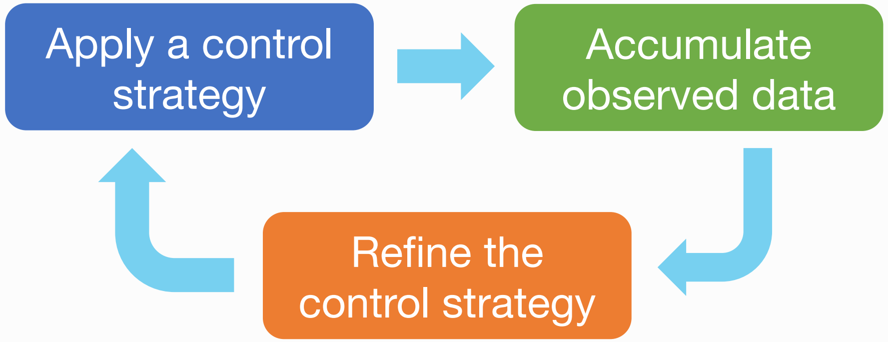
Harvard John A. Paulson School of Engineering and Applied Sciences

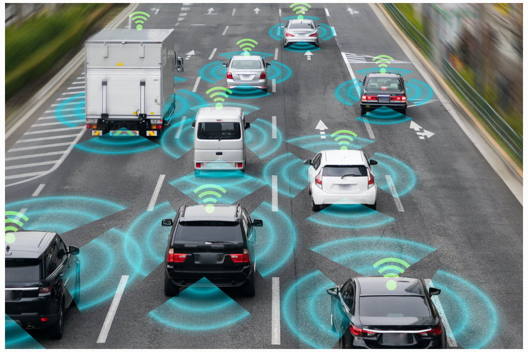
UC San Diego
JACOBS SCHOOL OF ENGINEERING
Electrical and Computer Engineering

Reinforcement Learning of Feedback Control Systems

- Learn the feedback controller with unknown/incomplete/complex system model






Autonomous driving

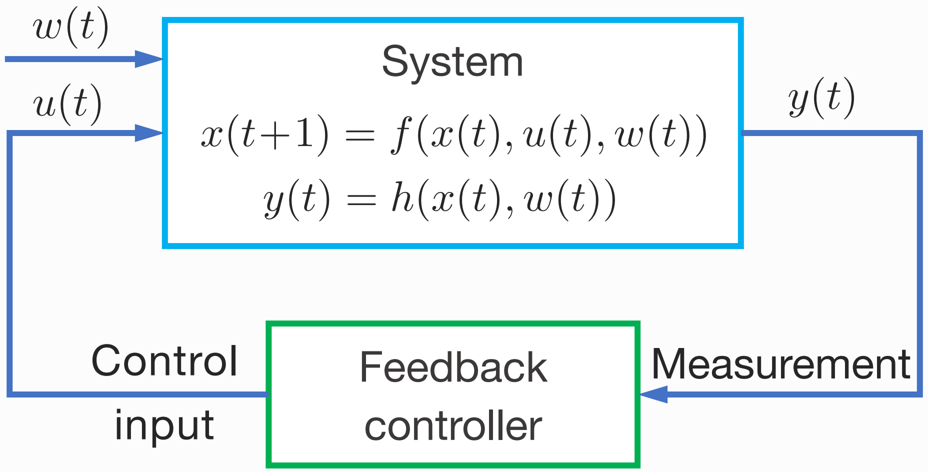

Swarm robotics

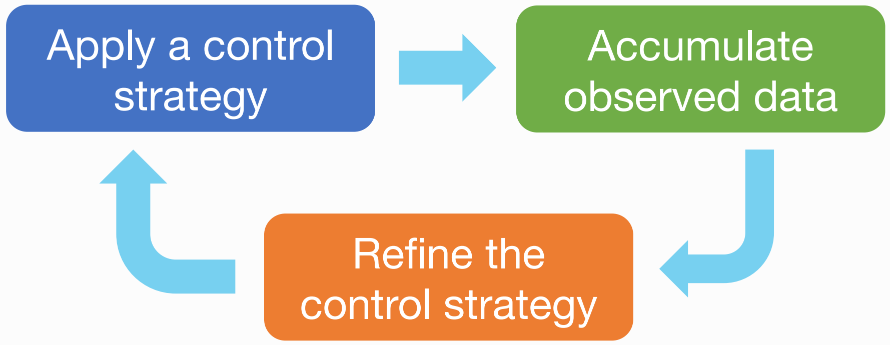

Manufacturing


Sensor networks

Reinforcement Learning of Feedback Control Systems

- Learn the feedback controller with unknown/incomplete/complex system model





Opportunities:

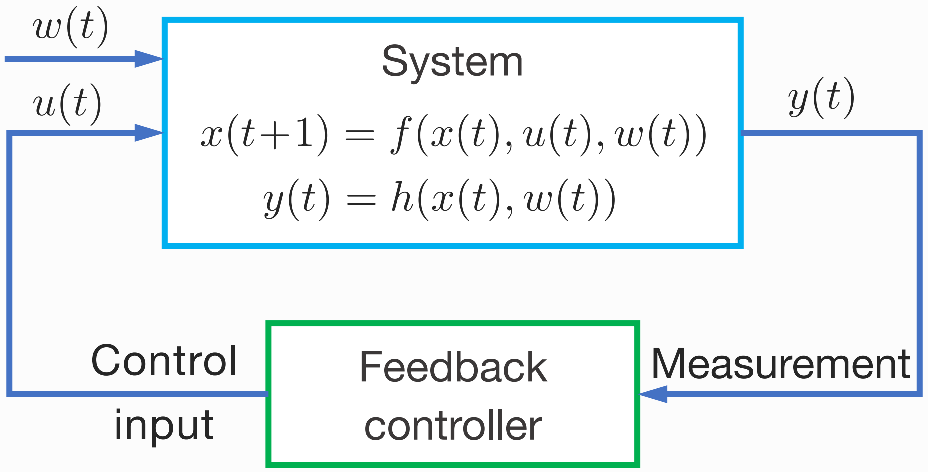
- Abundant, real-time data
- Computational power

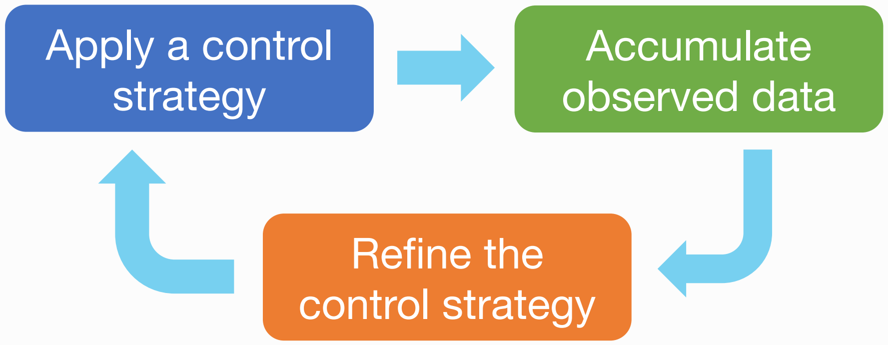
Challenges:

- Information restriction/incomplete measurement
- Rigorous performance guarantees
- Scalability
- …

Reinforcement Learning of Feedback Control Systems

- Learn the feedback controller with unknown/incomplete/complex system model





Observed data:

- Measurement $y(t)$
- Stage cost $c(x(t),u(t))$

Feedback controller/control policy:

- A mapping from historical measurements $(y(t), y(t-1), \dots)$ to the control input $u(t)$

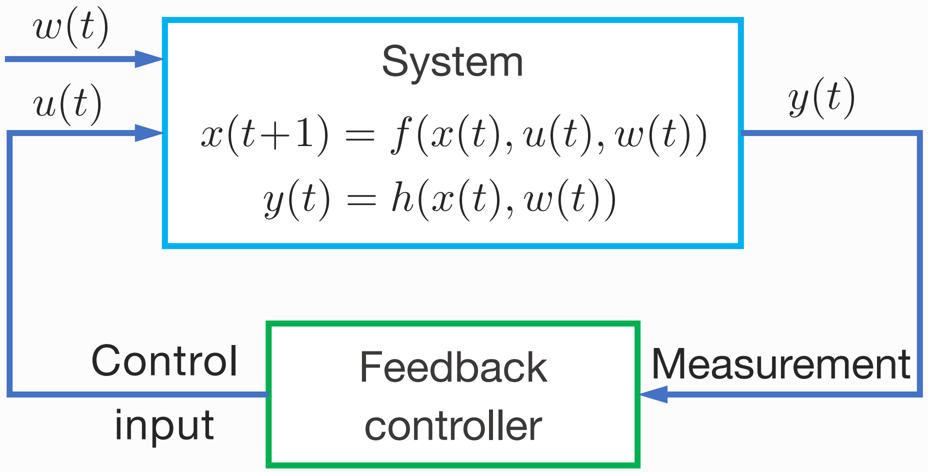
Goal: Find the best control policy that minimizes the accumulated cost

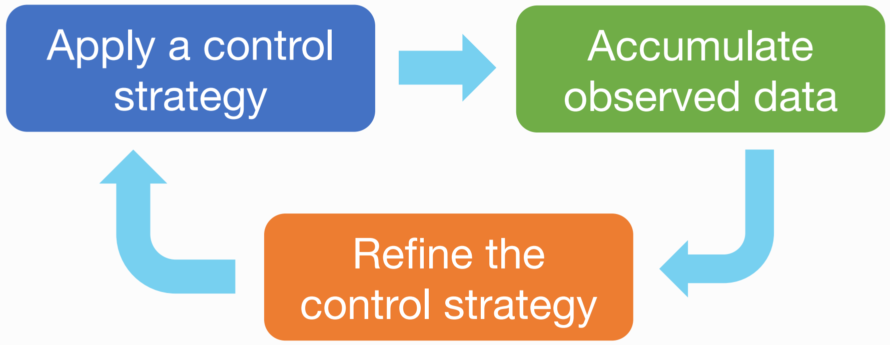
- discounted cost $\mathbb{E}[\sum_{t=0}^{\infty} \gamma^t c(x(t), u(t))]$
- infinite-horizon average cost

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}[c(x(t), u(t))]$$

Reinforcement Learning of Feedback Control Systems

- Learn the feedback controller with unknown/incomplete/complex system model





How to refine the control strategy based on observed data?

- **Model-free policy search**

  No model inference, use observed data more directly,

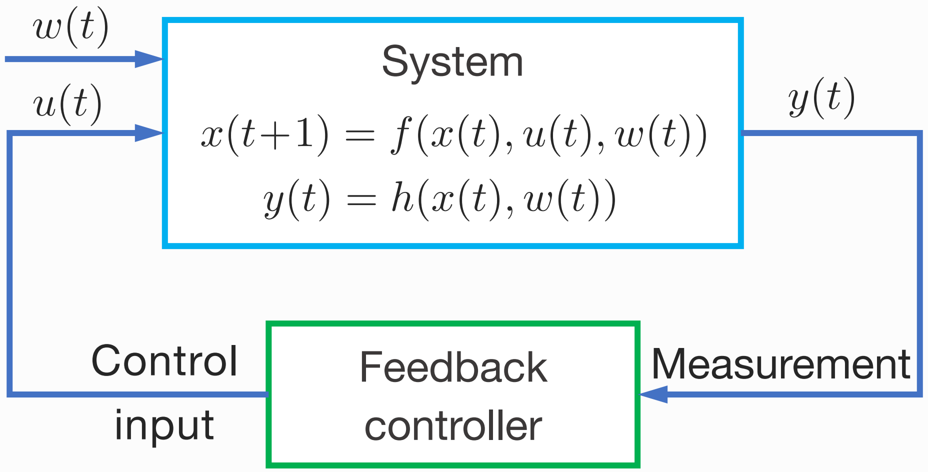
  - Policy gradient theorem/Q-learning
  - Zeroth-order optimization

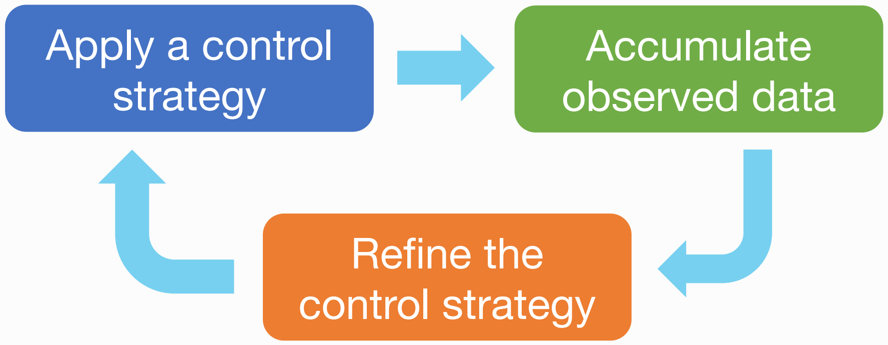
- **Model-based methods**

  Observed data → model inference → controller synthesis

Reinforcement Learning of Feedback Control Systems

- Learn the feedback controller with unknown/incomplete/complex system model



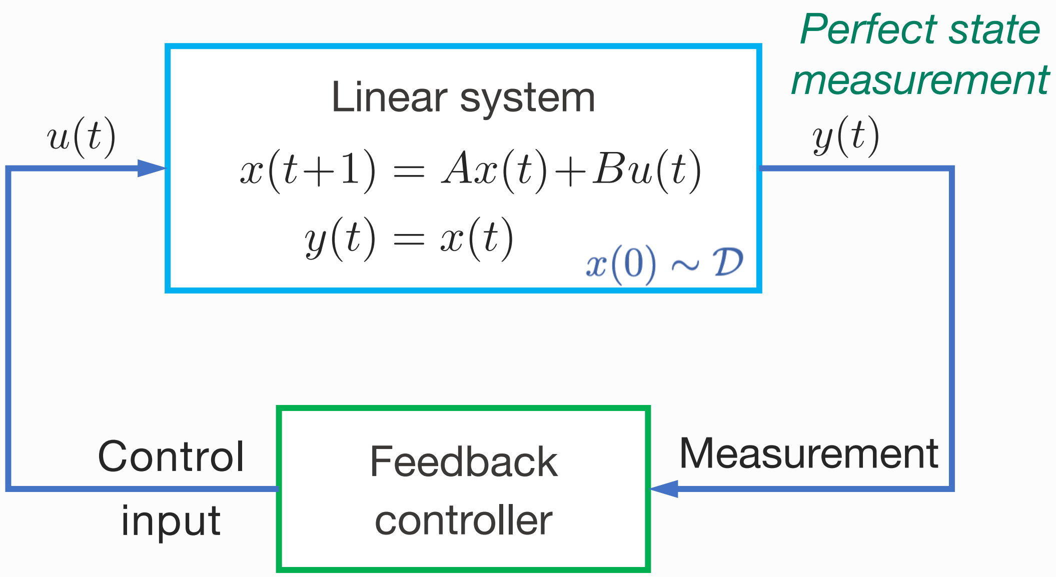


Theoretical & Practically Relevant Concerns

- **Sample complexity**
  # of measurement samples $\{y(t)\}$ needed to find an (approximately) optimal policy
- **Convergence rate**
  How fast the optimality gap decreases as we iteratively refine the control strategy
- **Stability**
  Whether the closed-loop system remains stable during the learning process

Reinforcement Learning of Linear Quadratic Regulators

Linear Quadratic Regulator (LQR)



- Control strategy: $u(t) = K\,x(t)$
- Accumulated cost:

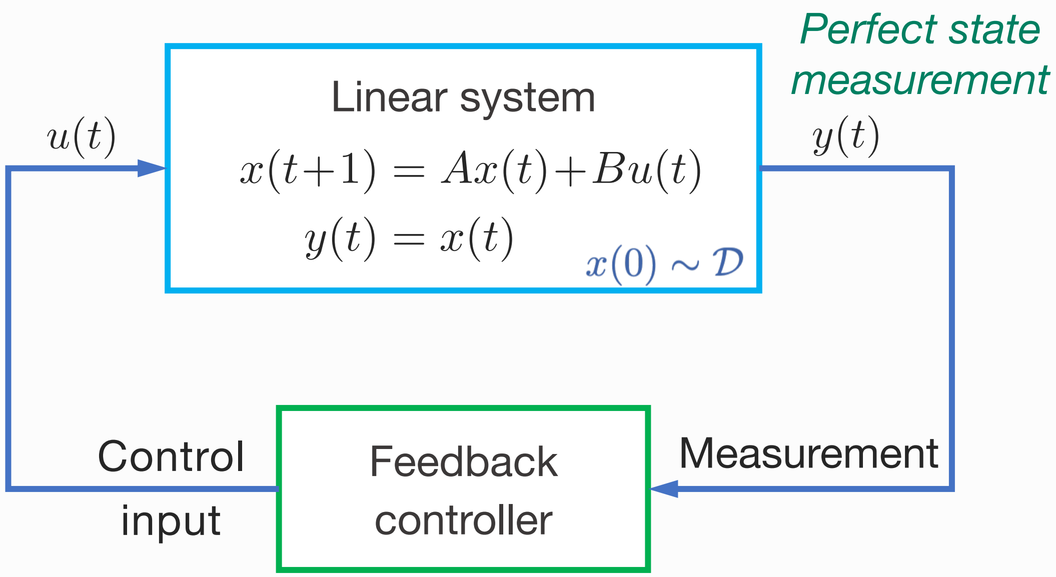
$$J(K) = \sum_{t=0}^{\infty} \mathbb{E}\big[\underbrace{x(t)^\top Q\,x(t) + u(t)^\top R\,u(t)}_{\text{Stage cost}}\big]$$

An optimization viewpoint:

$$\begin{aligned} \min_{K} \quad & J(K) \\ \text{s.t.} \quad & K \text{ stabilizes the system} \end{aligned}$$

Reinforcement Learning of Linear Quadratic Regulators

Linear Quadratic Regulator (LQR)



- Control strategy: $u(t) = K\,x(t)$
- Accumulated cost:

$$J(K) = \sum_{t=0}^{\infty} \mathbb{E}\big[\underbrace{x(t)^\top Q\,x(t) + u(t)^\top R\,u(t)}_{\text{Stage cost}}\big]$$

An optimization viewpoint:

$$K(s+1) = K(s) - \alpha \cdot \widehat{\nabla J(K(s))}$$

Zeroth-order gradient estimation

✓ **Fast global convergence** (exponential)

✓ **Low sample complexity**

✓ **Guaranteed stability** w.h.p.

[Fazel et al. 2018] [Malik et al. 2019] [Mohammadi et al. 2019]

Extension to Other Linear Quadratic Control Problems

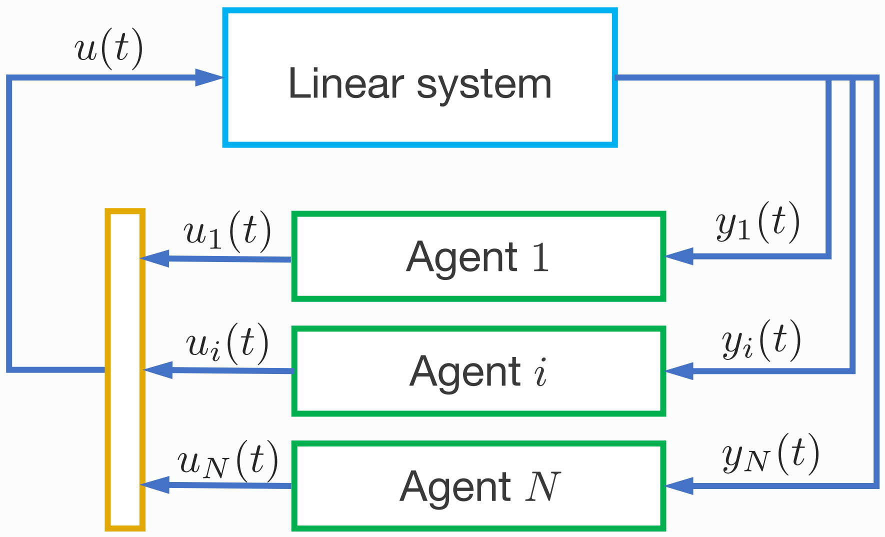
- Mixed $\mathcal{H}_2/\mathcal{H}_\infty$ design [Zhang et al. 2019], risk-constrained LQR [Zhao & You, 2021]

- This talk: Linear quadratic control with **partial/incomplete measurement**

Extension to Other Linear Quadratic Control Problems

Part I

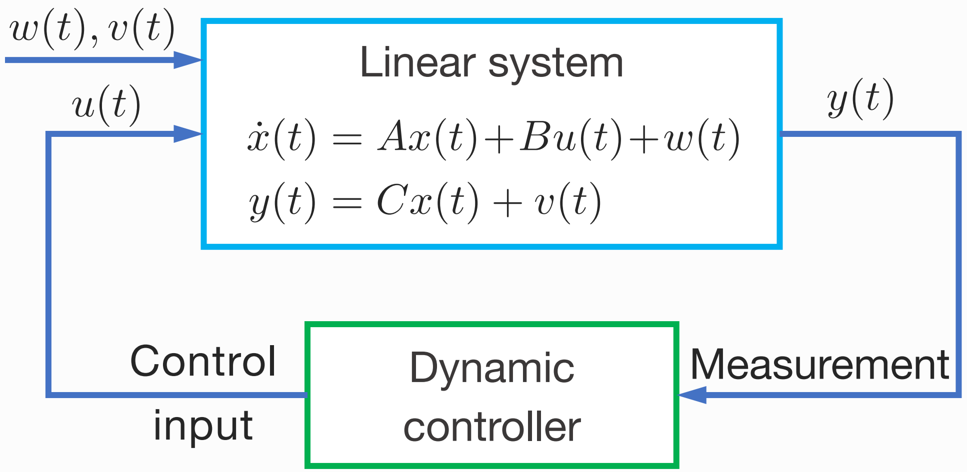
Distributed Reinforcement Learning for **Decentralized LQ** Control



➢ Swarm robotics, autonomous vehicles, mobile sensor networks

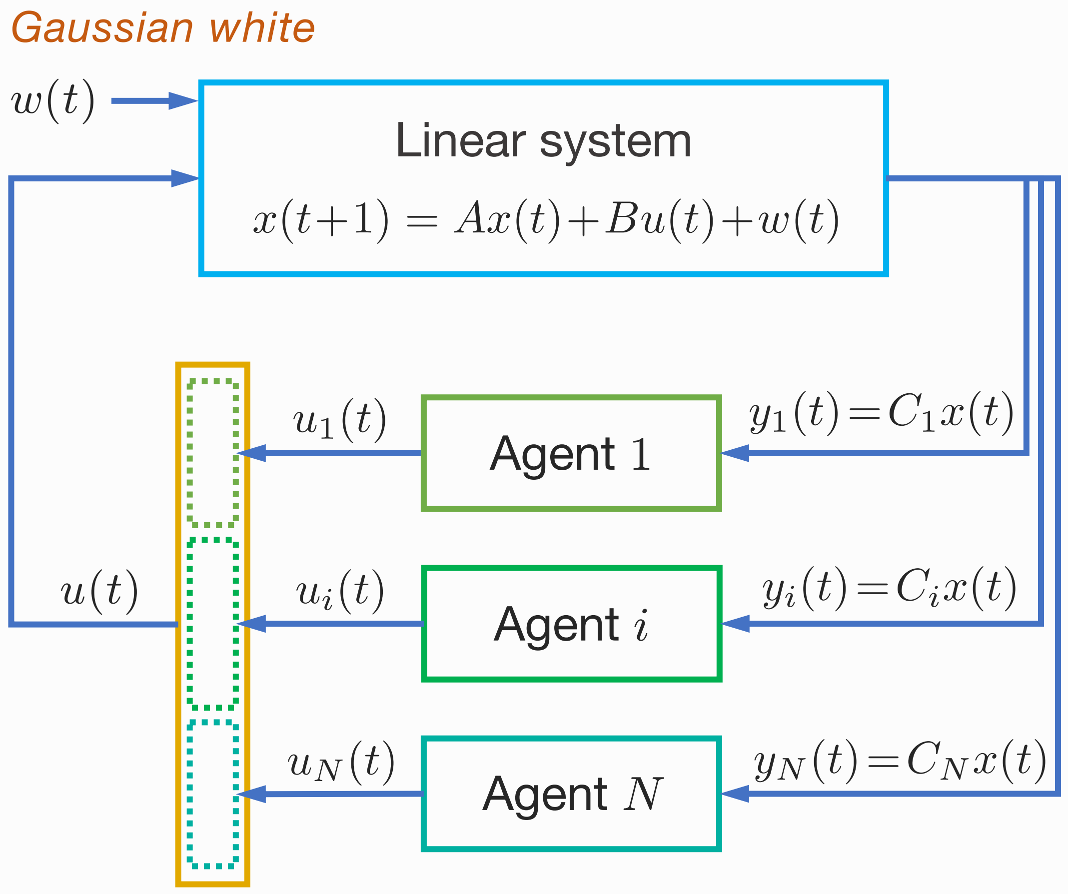
Part II

Optimization Landscape Analysis of **Linear Quadratic Gaussian** (LQG)



How does partial/imperfect measurement affect the problem structure?

Decentralized Linear Quadratic Control



- Control strategy: $u_i(t) = K_i\, y_i(t)$
- Stage cost: $c_i(t) = x(t)^\top Q_i\, x(t) + u(t)^\top Q_i\, u(t)$

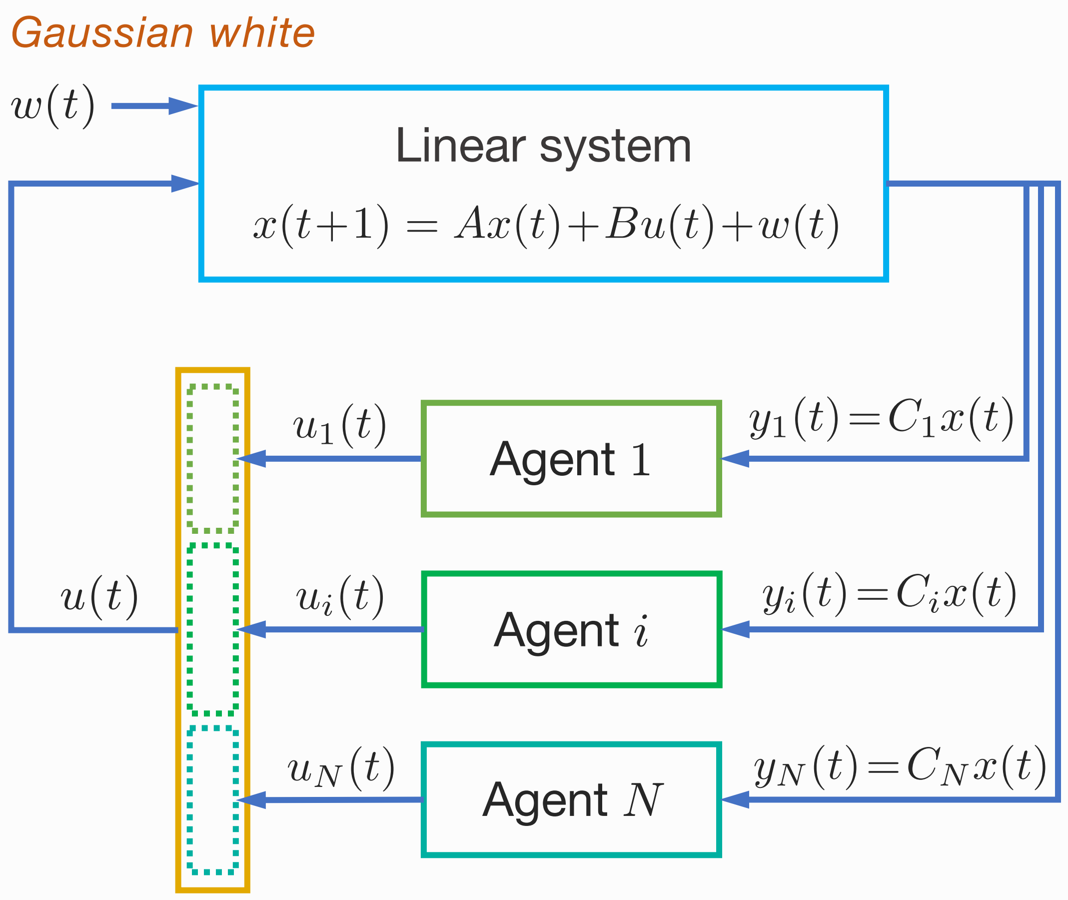
- **(Global) accumulated cost**

$$\text{minimize} \quad \frac{1}{N}\sum_{i=1}^{N} \lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[c_i(t)]$$

- **Local communication**

  Agents are connected by a bidirectional communication network $\mathcal{G} = (\{1,\ldots,N\},\mathcal{E})$

Decentralized Linear Quadratic Control

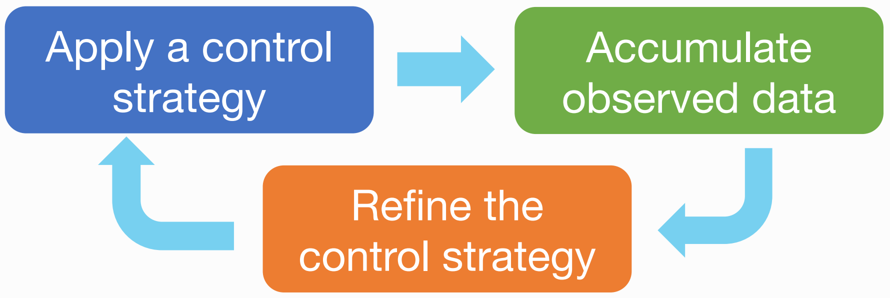


- Control strategy: $u_i(t) = K_i\, y_i(t)$
- Stage cost: $c_i(t) = x(t)^\top Q_i\, x(t) + u(t)^\top Q_i\, u(t)$

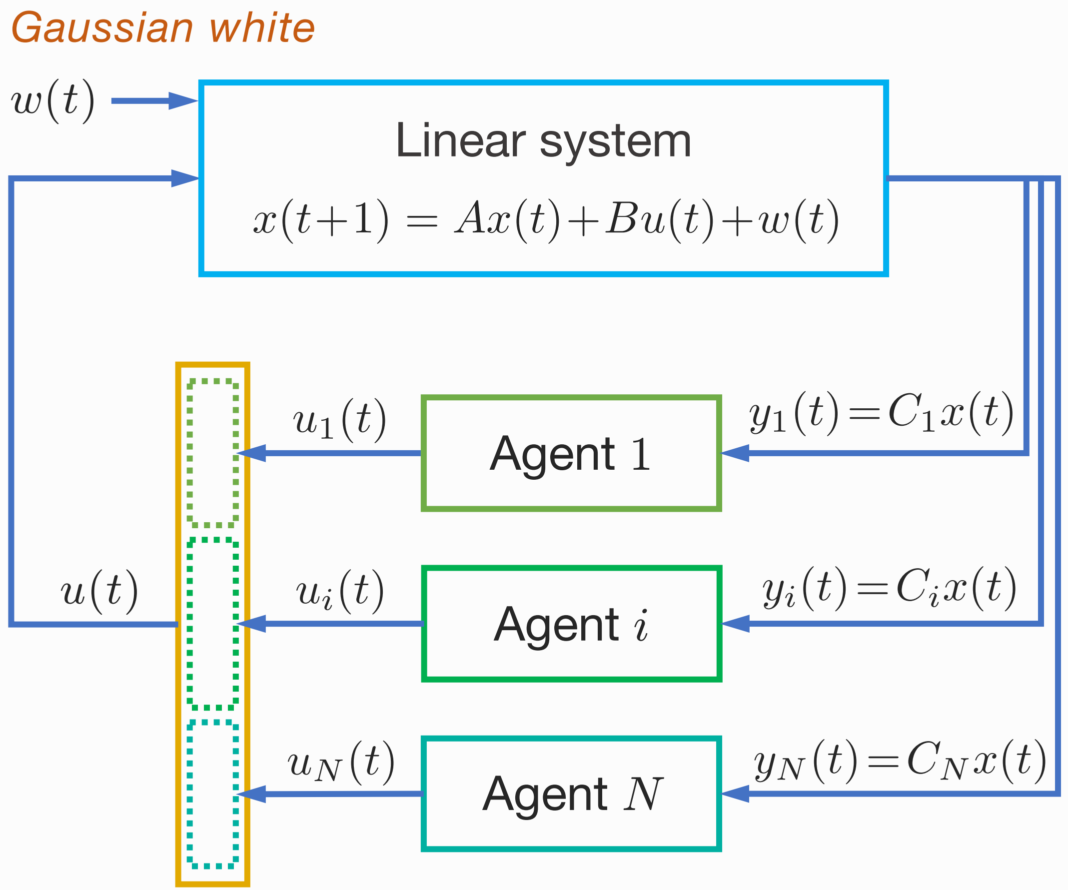
- **(Global) accumulated cost**

$$\text{minimize} \quad \frac{1}{N}\sum_{i=1}^{N} \lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[c_i(t)]$$

- **Local communication** $\mathcal{G} = (\{1,\ldots,N\},\mathcal{E})$
- **Distributed reinforcement learning**
  - Unknown system matrices A, B, C_i
  - Coordination via local communication rather than a central server



Decentralized Linear Quadratic Control

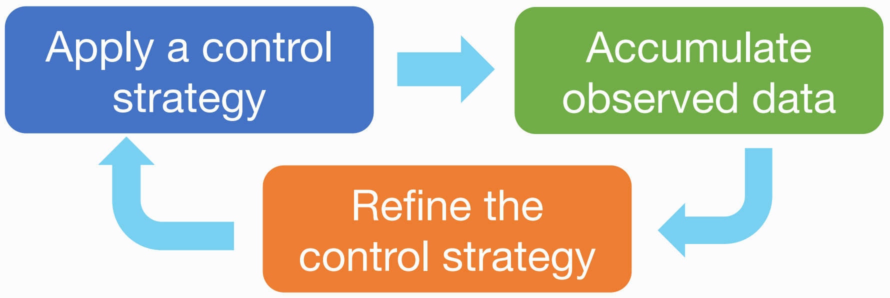


- Control strategy: $u_i(t) = K_i\, y_i(t)$
- Stage cost: $c_i(t) = x(t)^\top Q_i\, x(t) + u(t)^\top Q_i\, u(t)$

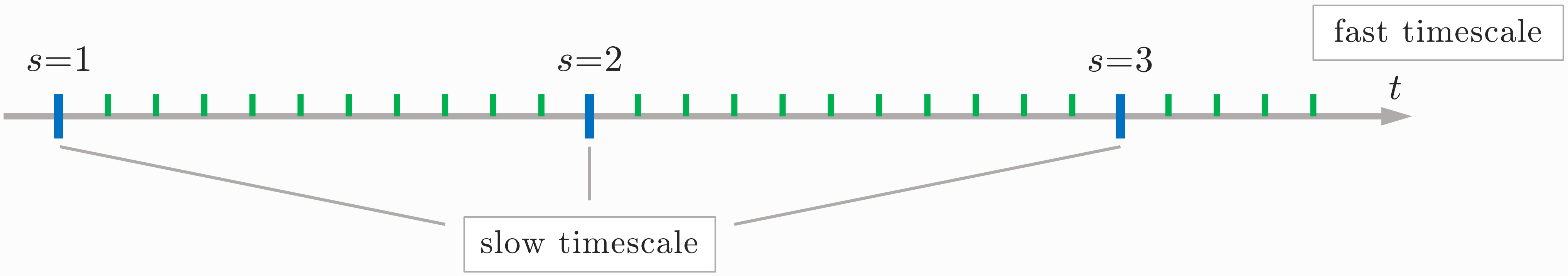
An optimization viewpoint:

$$\begin{aligned} \min_{K=(K_1,\dots,K_N)} \quad & J(K) \\ \text{s.t.} \quad & K \text{ stabilizes the system} \end{aligned}$$

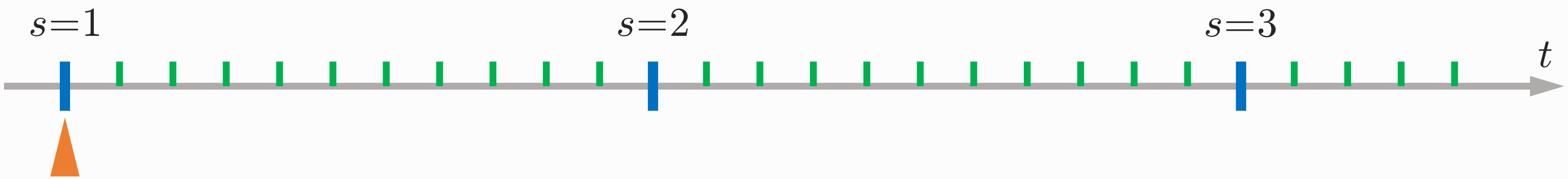
$$J(K) = \frac{1}{N}\sum_{i=1}^{N} \lim_{T\to\infty} \frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[c_i(t)]$$



Algorithm Design



Algorithm

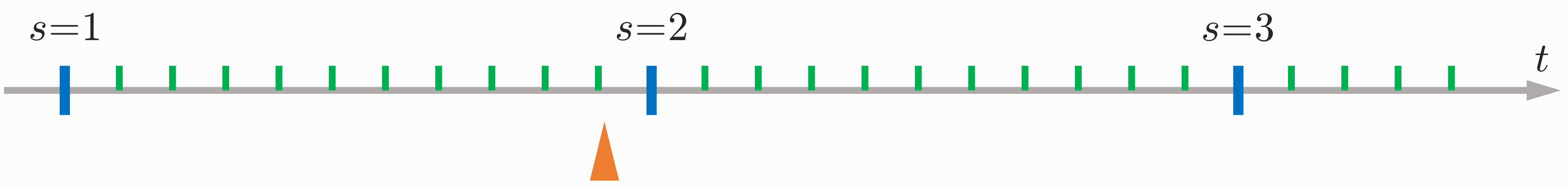


1. Generate random perturbation $z_i(s)$
2. Apply control policy $K_i(s) + rz_i(s)$ to the system

Algorithm

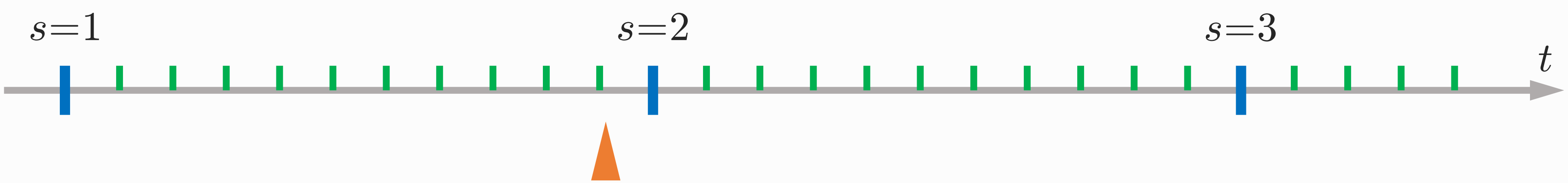
1. Generate random perturbation $z_i(s)$
2. Apply control policy $K_i(s) + rz_i(s)$ to the system
3. **Accumulate** costs $c_i(t)$ & **exchange info** with neighbors

Algorithm



1. Generate random perturbation $z_i(s)$
2. Apply control policy $K_i(s) + rz_i(s)$ to the system
3. Accumulate costs $c_i(t)$ & exchange info with neighbors
4. Obtain an estimated **global** obj. $\hat{J}_i(s) \approx J(K(s)+rz(s))$

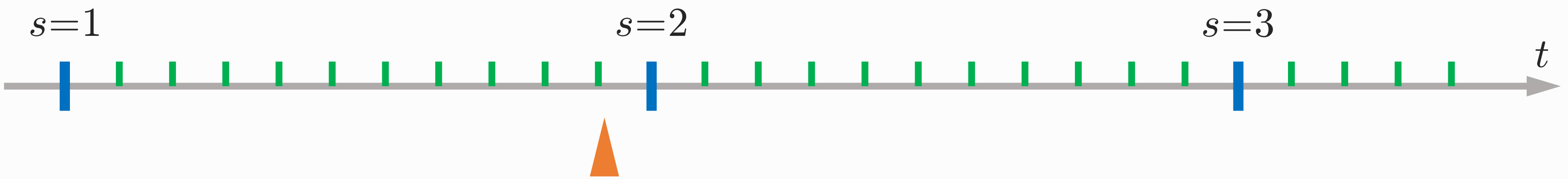
Algorithm



1. Generate random perturbation $z_i(s)$
2. Apply control policy $K_i(s) + rz_i(s)$ to the system
3. Accumulate costs $c_i(t)$ & exchange info with neighbors
4. Obtain an estimated global obj. $\hat{J}_i(s) \approx J(K(s)+rz(s))$
5. Construct **zeroth-order partial gradient estimator**

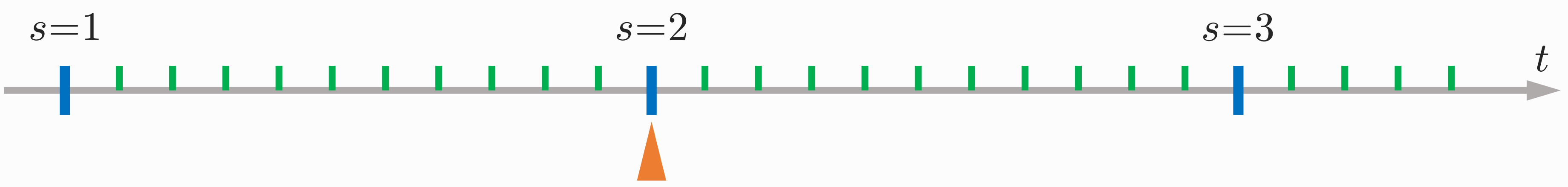
$$\hat{G}_i(s) = \frac{d}{r}\,\hat{J}_i(s)\, z_i(s)$$

Algorithm



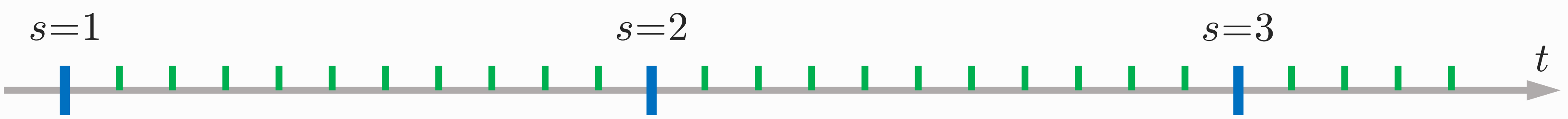
1. Generate random perturbation $z_i(s)$
2. Apply control policy $K_i(s) + r z_i(s)$ to the system
3. Accumulate costs $c_i(t)$ & exchange info with neighbors
4. Obtain an estimated global obj. $\hat{J}_i(s) \approx J(K(s)+rz(s))$
5. Construct zeroth-order partial gradient estimator
$$\hat{G}_i(s) = \frac{d}{r}\,\hat{J}_i(s)\, z_i(s)$$
6. Update by **stochastic gradient descent** $K_i(s+1) = K_i(s) - \eta \hat{G}_i(s)$

Algorithm



1. Generate random perturbation $z_i(s)$
2. Apply control policy $K_i(s) + rz_i(s)$ to the system
3. Accumulate costs $c_i(t)$ & exchange info with neighbors
4. Obtain an estimated global obj. $\hat{J}_i(s) \approx J(K(s){+}rz(s))$
5. Construct zeroth-order partial gradient estimator
$$\hat{G}_i(s) = \frac{d}{r}\,\hat{J}_i(s)\,z_i(s)$$
6. Update by stochastic gradient descent $K_i(s{+}1) = K_i(s) - \eta\hat{G}_i(s)$

Algorithm



1. Generate random perturbation $z_i(s)$
2. Apply control policy $K_i(s) + rz_i(s)$ to the system
3. Accumulate costs $c_i(t)$ & exchange info with neighbors
4. Obtain an estimated global obj. $\hat{J}_i(s) \approx J(K(s)+rz(s))$
5. Construct **zeroth-order partial gradient estimator**

$$\hat{G}_i(s) = \frac{d}{r}\,\hat{J}_i(s)\,z_i(s)$$

6. Update by stochastic gradient descent $K_i(s+1) = K_i(s) - \eta\hat{G}_i(s)$

Zeroth-order gradient estimation

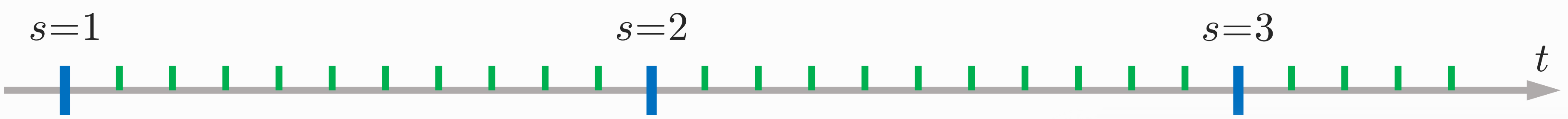
$$\mathsf{G}(K; r, z) = \frac{d}{r} J(K + rz) z$$

- d: dimension of K
- r: smoothing radius
- z: random perturbation

$$\mathbb{E}_z[\mathsf{G}(K; r, z)] = \nabla J(K) + O(r)$$

[Flaxman et al. 2005] [Nesterov & Spokoiny 2017]

Algorithm



1. Generate random perturbation $z_i(s)$
2. Apply control policy $K_i(s) + rz_i(s)$ to the system
3. Accumulate costs $c_i(t)$ & exchange info with neighbors
4. Obtain an estimated global obj. $\hat{J}_i(s) \approx J(K(s)\!+\!rz(s))$
5. Construct zeroth-order partial gradient estimator

$$\hat{G}_i(s) = \frac{d}{r}\,\hat{J}_i(s)\,z_i(s)$$

6. Update by stochastic gradient descent $K_i(s\!+\!1) = K_i(s)$

Consensus method

$$\mu_i(t) = \frac{t-1}{t}\sum_j W_{ij}\,\mu_i(t-1) + \frac{1}{t}c_i(t)$$

- W: communication weight matrix
  - $N\times N$ doubly stochastic
  - $W_{ij} = 0$ if (i,j) not connected

$$\mathbb{E}\left|\mu_i(T_J) - \frac{1}{NT_J}\sum_{i=1}^{N}\sum_{\tau=t}^{T_J} c_i(t)\right| = O\left(\frac{1}{T_J}\right)$$

Finite-horizon approximation of J

Theoretical Analysis

- Inspired by existing works on centralized LQR [Malik et al. 2019] [Bu et al. 2020]
- Major technical contributions in our extension to the decentralized setting:
  - Handling unbounded Gaussian process noise
  - Treating infinite-horizon average cost, rather than discounted cost
  - Bounding error caused by finite-horizon approximation in generating $z_i(s)$ and producing the estimate $\hat{J}_i(s) \approx J(K(s) + rz(s))$
  - Explicit bound for the sampling complexity

Performance Guarantees

Theorem (informal)

Let $\epsilon > 0$ be arbitrary. By choosing the parameters of the algorithm to satisfy

$$r \sim O(\sqrt{\epsilon}) \qquad \eta \sim O\left(\epsilon r^2\right) \qquad T_J \sim \Omega\left(\frac{1}{r\sqrt{\epsilon}}\right) \qquad T_G \sim \Theta\left(\frac{1}{\eta\epsilon}\right)$$

we can achieve the following with high probability:

- The closed-loop system remain **stable** during the learning procedure
- Optimality guarantee given by

$$\frac{1}{T_G}\sum_{s=1}^{T_G} \|\nabla J(K(s))\|^2 \leq \epsilon$$

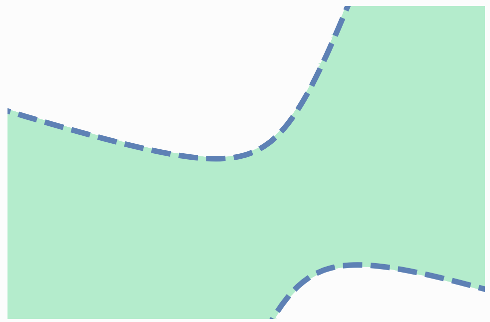
A relatively weak optimality guarantee

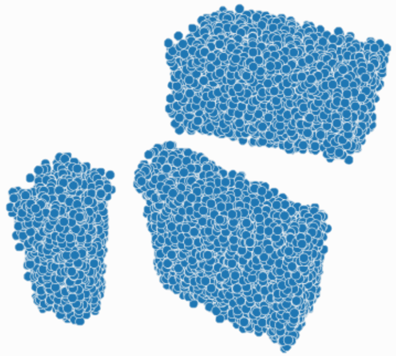
Why?

Corollary: Sample complexity bound given by $T_G T_J \sim \Theta\left(\frac{1}{\epsilon^4}\right)$

Comparison with Centralized LQR

	Centralized LQR	Decentralized LQ control
Stability	Y	Y
Optimality	$J(K(T_G)) - J(K^*) \le \epsilon$	$\frac{1}{T_G}\sum_{s=1}^{T_G} \|\nabla J(K(s))\|^2 \le \epsilon$
Domain	Nonconvex, connected	Multiple connected components

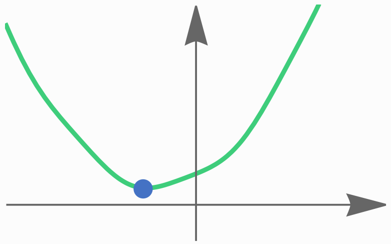




[Feng & Lavaei 2019]

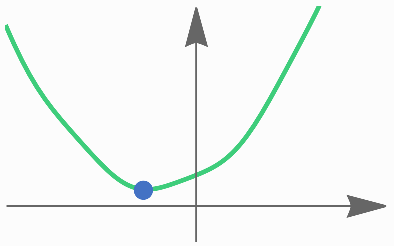
Comparison with Centralized LQR

	Centralized LQR	Decentralized LQ control
Stability	Y	Y
Optimality	$J(K(T_G)) - J(K^*) \leq \epsilon$	$\frac{1}{T_G}\sum_{s=1}^{T_G} \|\nabla J(K(s))\|^2 \leq \epsilon$
Domain	Nonconvex, connected	Multiple connected components
J(K)	• Coercive • Gradient dominance • Unique stationary point	• Coercive • Not gradient dominance • Multiple stationary points • Lacks good properties



[Fazel et al. 2018] [Malik et al. 2019] [Mohammadi et al. 2019] [Bu et al. 2019] [Feng & Lavaei 2019]

Comparison with Centralized LQR

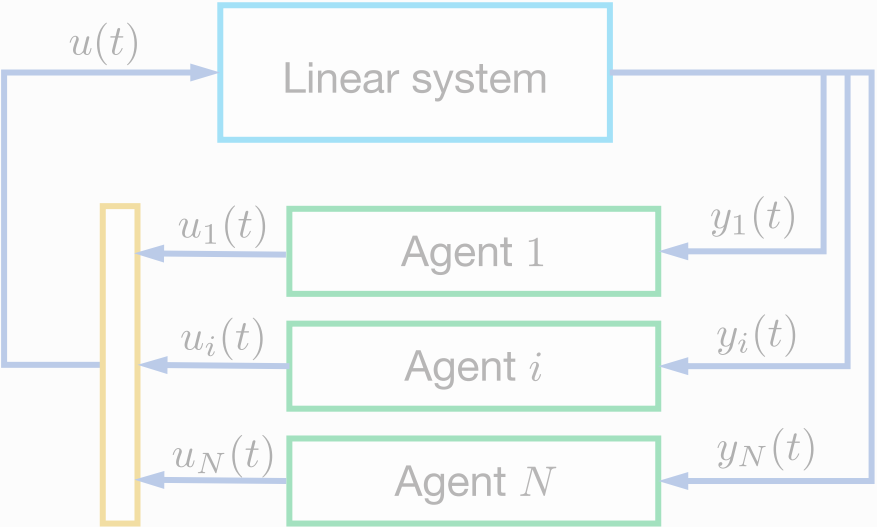
	Centralized LQR	Single-agent, partial measurement, $u(t) = K\,y(t)$
Stability	Y	Y
Optimality	$J(K(T_G)) - J(K^*) \le \epsilon$	$\frac{1}{T_G}\sum_{s=1}^{T_G} \|\nabla J(K(s))\|^2 \le \epsilon$
Domain	Nonconvex, connected	Multiple connected components
J(K)	• Coercive • Gradient dominance • Unique stationary point 	• Coercive • Not gradient dominance • Multiple stationary points • Lacks good properties

[Fazel et al. 2018] [Malik et al. 2019] [Mohammadi et al. 2019] [Bu et al. 2019] [Feng & Lavaei 2019]

Extension to Other Linear Quadratic Control Problems

Part I

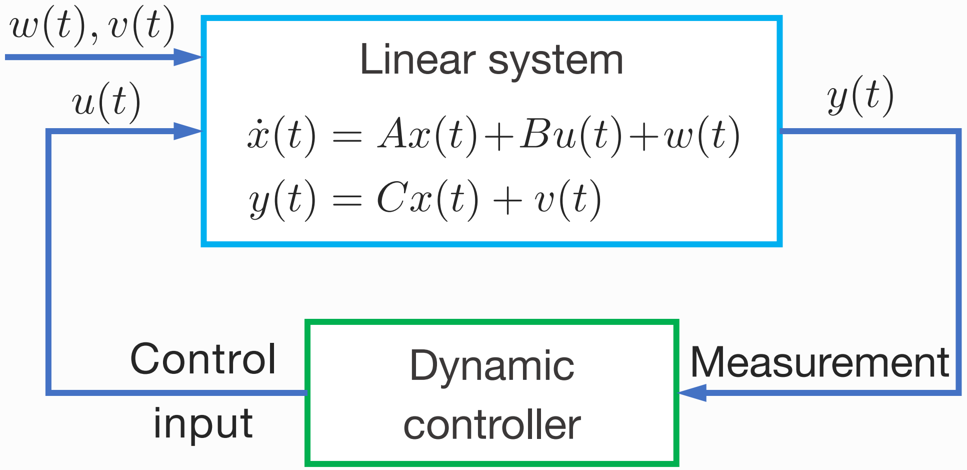
Distributed Reinforcement Learning
for **Decentralized LQ** Control



- Swarm robotics, autonomous vehicles, mobile sensor networks

Part II

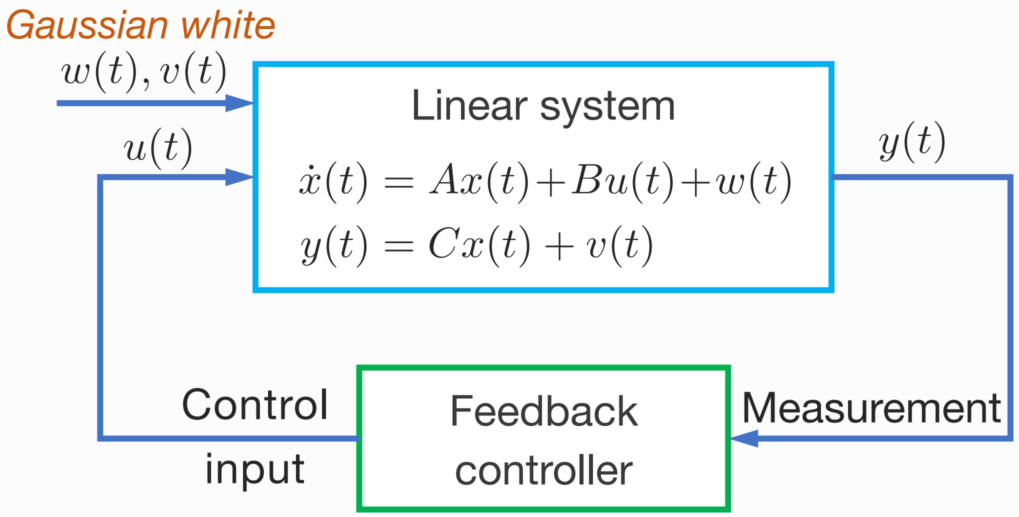
Optimization Landscape Analysis of
Linear Quadratic Gaussian (LQG)



How does partial/imperfect measurement affect the problem structure?

Optimization Landscape of LQG

Linear Quadratic Gaussian (LQG)



- Control strategy: $\mathsf{K} \in \mathcal{K}$
- Accumulated cost:

$$J(\mathsf{K}) = \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}\big[\underbrace{x(t)^\top Q\, x(t) + u(t)^\top R\, u(t)}_{\text{Stage cost}}\big]$$

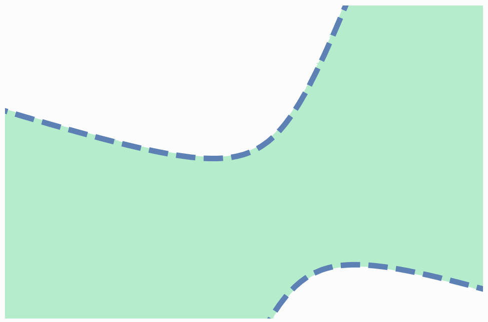
An optimization viewpoint:

$$\begin{aligned} \min_{\mathsf{K}\in\mathcal{K}} \quad & J(\mathsf{K}) \\ \text{s.t.} \quad & \mathsf{K} \text{ stabilizes the system} \end{aligned}$$

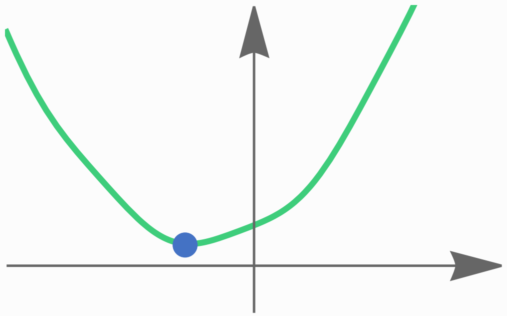
Optimization Landscape Analysis

- Properties of the domain (set of stabilizing controllers)
  - convexity, connectivity, open/closed
- Properties of the accumulated cost J
  - convexity, differentiability, coercivity
  - set of stationary points/local minima/global minima

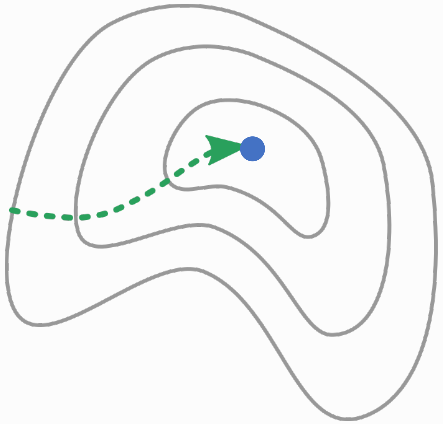
Existing Work: Optimization Landscape of LQR



Possibly nonconvex,
connected,



Coercive, gradient dominance,
unique stationary point



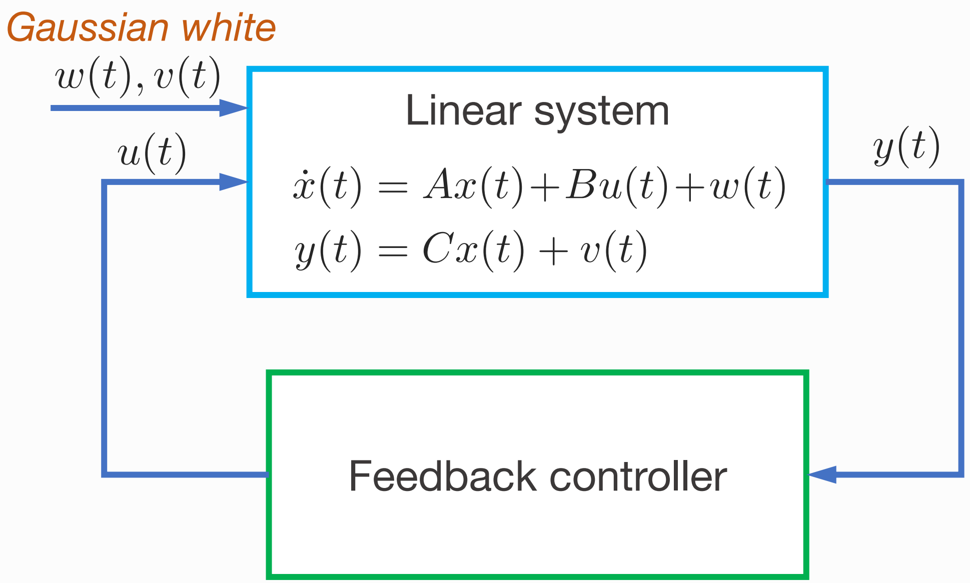
✓ Fast convergence to global optimum for gradient-based methods

[Fazel et al. 2018] [Malik et al. 2019] [Mohammadi et al. 2019] [Bu et al. 2019]

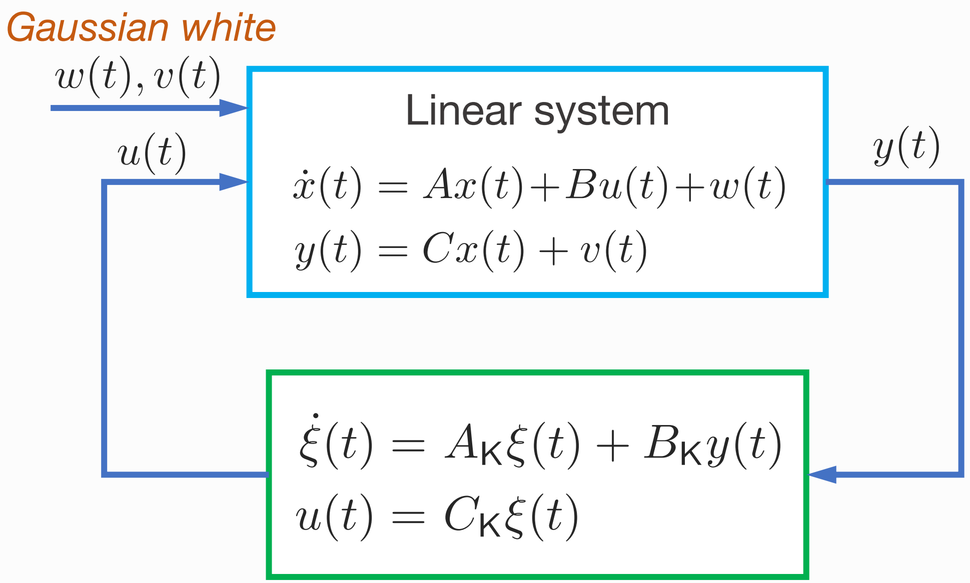
Our Focus: Optimization Landscape of LQG

- Extension from LQR to LQG is highly nontrivial
  - LQG control theory is more sophisticated
  - Some results of LQR may not hold for LQG anymore
  - The domain consists of **dynamic controllers**, leading to more complex landscape structure

Dynamic Controllers



Dynamic Controllers



dynamic controller

$\mathsf{K} = (A_{\mathsf{K}}, B_{\mathsf{K}}, C_{\mathsf{K}})$

$\xi(t)$ internal state of the controller

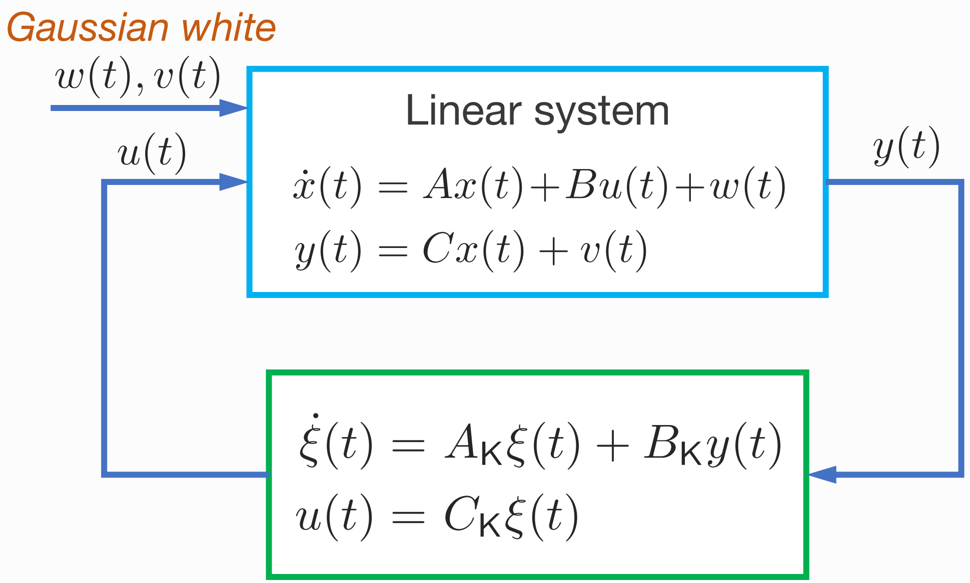
$\dim \xi(t)$ order of the controller

$\dim \xi(t) = \dim x(t)$ full-order

$\dim \xi(t) < \dim x(t)$ reduced-order

Theorem. The optimal control policy for LQG is a full-order dynamic controller.

Dynamic Controllers



dynamic controller

$K = (A_K, B_K, C_K)$

$\xi(t)$ internal state of the controller

$\dim \xi(t)$ order of the controller

$\dim \xi(t) = \dim x(t)$ full-order

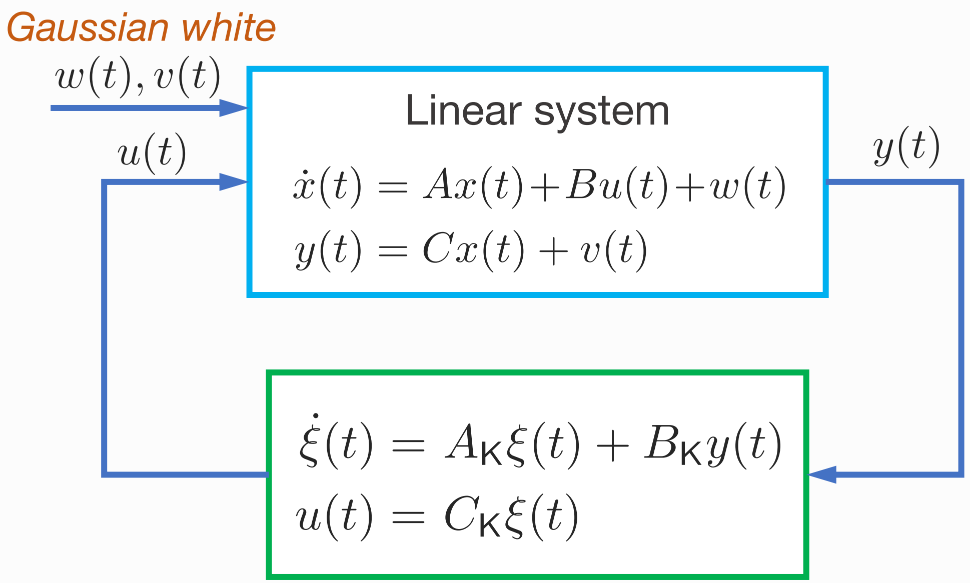
$\dim \xi(t) < \dim x(t)$ reduced-order

minimal controller

The input-output behavior cannot be replicated by a lower order controller.

* (A_K, B_K, C_K) controllable and observable

Objective Function and Domain



dynamic controller

$\mathsf{K} = (A_{\mathsf{K}}, B_{\mathsf{K}}, C_{\mathsf{K}})$

- Objective function $J(\mathsf{K}) : \mathcal{C}_{\text{full}} \to \mathbb{R}$

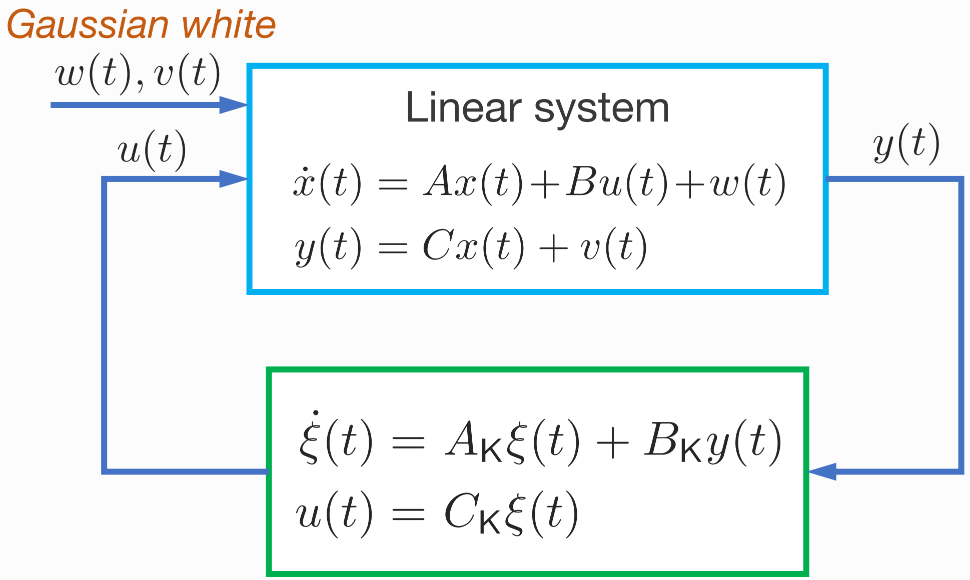
  Set of **full-order**, **stabilizing** dynamic controllers

- When does K stabilize the system?
  - Dynamics of the closed-loop system:

$$\frac{d}{dt}\begin{bmatrix} x \\ \xi \end{bmatrix} = \begin{bmatrix} A & BC_{\mathsf{K}} \\ B_{\mathsf{K}}C & A_{\mathsf{K}} \end{bmatrix}\begin{bmatrix} x \\ \xi \end{bmatrix} + \begin{bmatrix} I & 0 \\ 0 & B_{\mathsf{K}} \end{bmatrix}\begin{bmatrix} w \\ v \end{bmatrix}$$

$$\begin{bmatrix} y \\ u \end{bmatrix} = \begin{bmatrix} C & 0 \\ 0 & C_{\mathsf{K}} \end{bmatrix}\begin{bmatrix} x \\ \xi \end{bmatrix} + \begin{bmatrix} v \\ 0 \end{bmatrix}$$

Objective Function and Domain



dynamic controller

$\mathsf{K} = (A_{\mathsf{K}}, B_{\mathsf{K}}, C_{\mathsf{K}})$

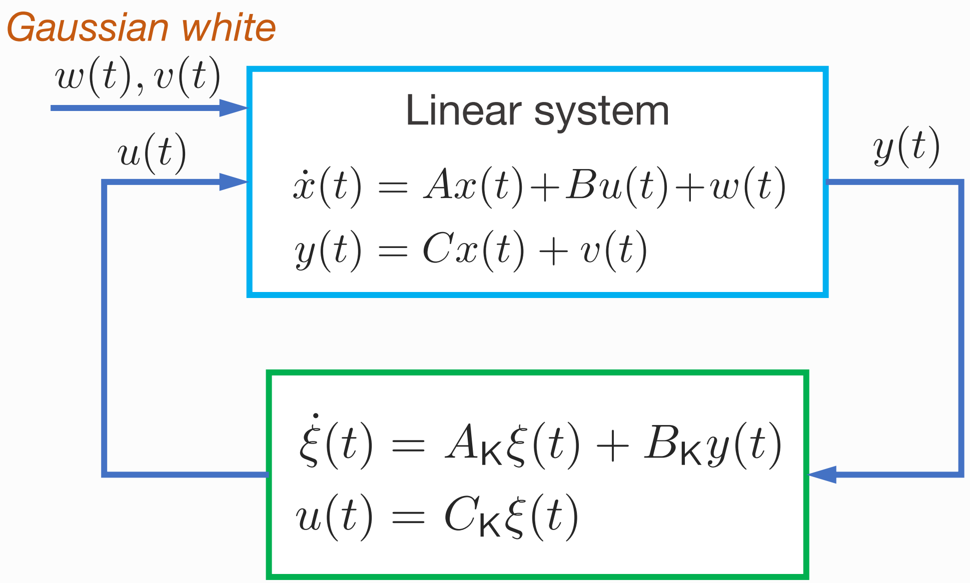
- Objective function $J(\mathsf{K}) : \mathcal{C}_{\text{full}} \to \mathbb{R}$

  Set of **full-order**, **stabilizing** dynamic controllers

- When does K stabilize the system?

$$\mathcal{C}_{\text{full}} = \left\{ \mathsf{K} \,\middle|\, \mathsf{K} = (A_{\mathsf{K}}, B_{\mathsf{K}}, C_{\mathsf{K}}) \text{ is full-order, } \begin{bmatrix} A & BC_{\mathsf{K}} \\ B_{\mathsf{K}}C & A_{\mathsf{K}} \end{bmatrix} \text{ is Hurwitz stable} \right\}$$

Objective Function and Domain



dynamic controller

$\mathsf{K} = (A_{\mathsf{K}}, B_{\mathsf{K}}, C_{\mathsf{K}})$

$$\begin{aligned} \min_{\mathsf{K}} \quad & J(\mathsf{K}) \\ \text{s.t.} \quad & \mathsf{K} = (A_{\mathsf{K}}, B_{\mathsf{K}}, C_{\mathsf{K}}) \in \mathcal{C}_{\text{full}} \end{aligned}$$

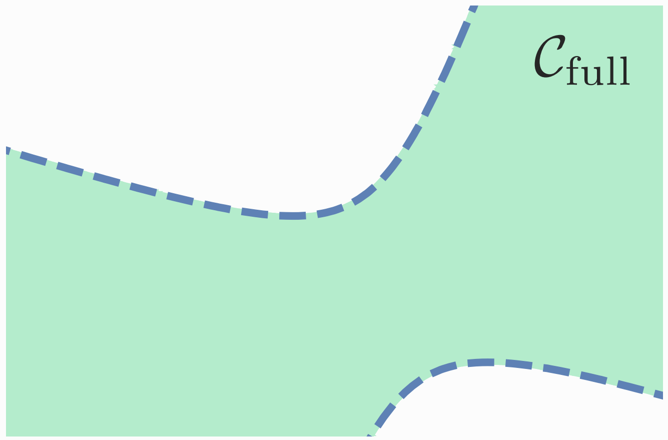
Objective: $J(\mathsf{K})$ The accumulated cost

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}\left[x(t)^\top Q\, x(t) + u(t)^\top R\, u(t)\right]$$

Domain: $\mathcal{C}_{\text{full}}$ The set of **full-order, stabilizing** dynamic controllers

Preliminary Results on the Domain

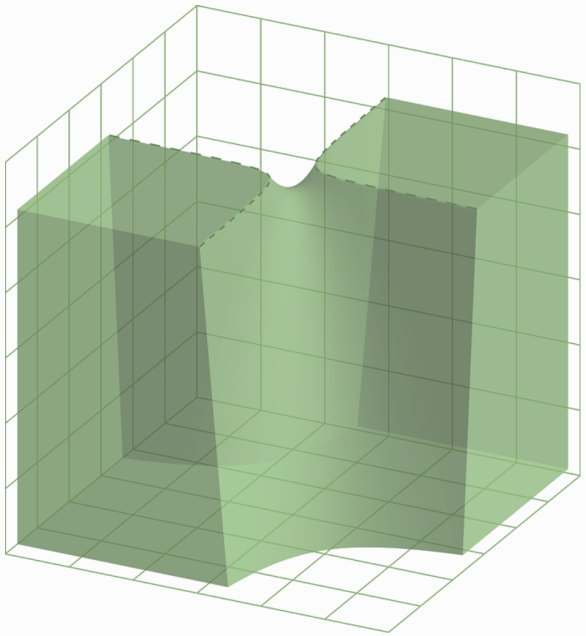
Proposition. The domain $\mathcal{C}_{\text{full}}$ is open, unbounded, and can be nonconvex.

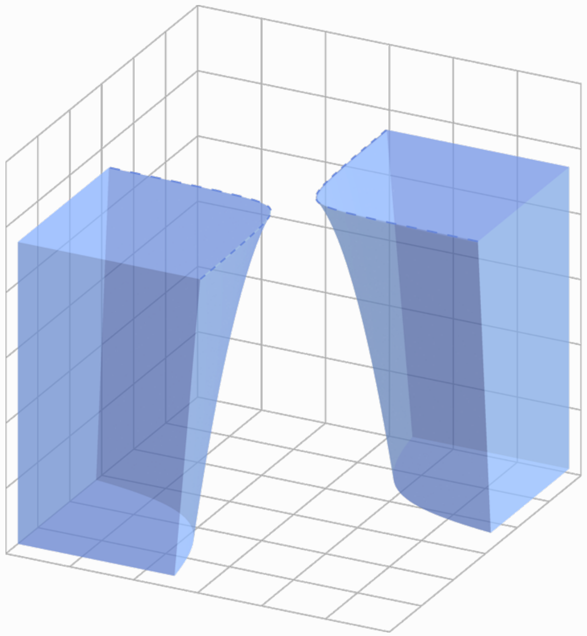


Connectivity of the Domain

Theorem 1. Under some standard assumptions,

1) The set $\mathcal{C}_{\mathrm{full}}$ can be disconnected, but has at most 2 connected components.





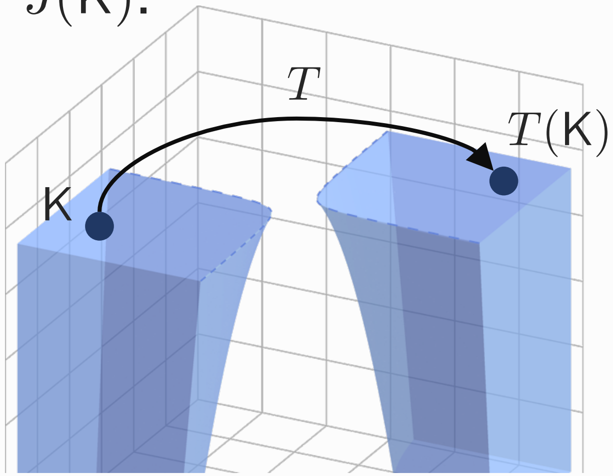
Connectivity of the Domain

Theorem 1. Under some standard assumptions,

1) The set $\mathcal{C}_{\text{full}}$ can be disconnected, but has at most 2 connected components.
2) If $\mathcal{C}_{\text{full}}$ has 2 connected components, then the mapping

$$(A_{\mathsf{K}}, B_{\mathsf{K}}, C_{\mathsf{K}}) \mapsto (A_{\mathsf{K}}, -B_{\mathsf{K}}, -C_{\mathsf{K}})$$

is a bijection between the 2 connected components that does not change the value of $J(\mathsf{K})$.



$J(\mathsf{K}) = J(T(\mathsf{K}))$

For gradient-based local search methods, it makes no difference to search over either connected component.

Connectivity of the Domain

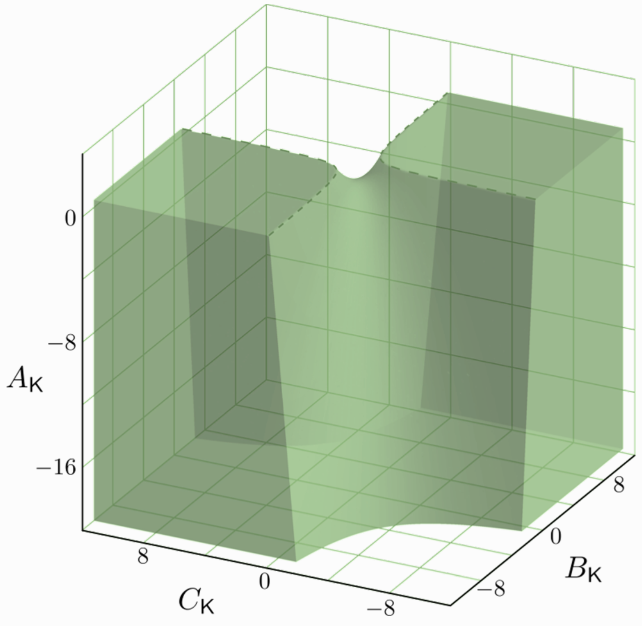
Theorem 2. Under some standard assumptions,

1) $\mathcal{C}_{\text{full}}$ is connected if the plant is open-loop stable or there exists a reduced-order stabilizing controller.

2) The sufficient condition of connectivity in 1) becomes necessary if the plant is single-input or single-output.

Example 1.
$$\dot{x}(t) = -x(t) + u(t) + w(t) \qquad x(t) \in \mathbb{R}$$
$$y(t) = x(t) + v(t)$$

- open-loop stable



Connectivity of the Domain

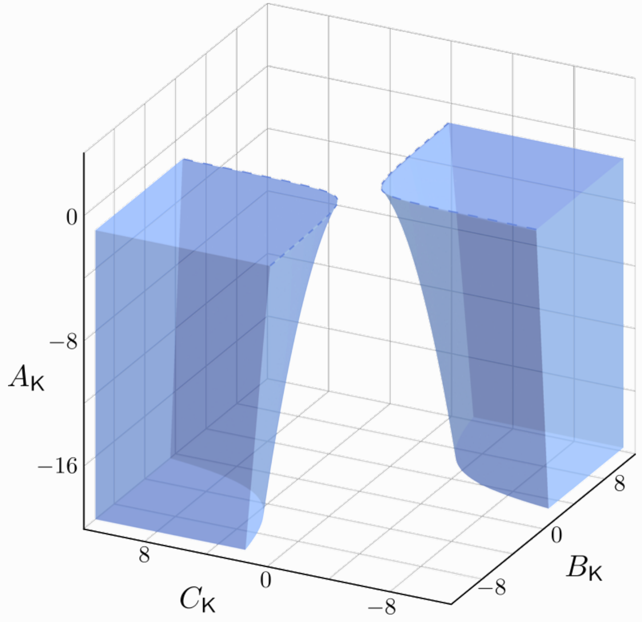
Theorem 2. Under some standard assumptions,

1) $\mathcal{C}_{\mathrm{full}}$ is connected if the plant is open-loop stable or there exists a reduced-order stabilizing controller.

2) The sufficient condition of connectivity in 1) becomes necessary if the plant is single-input or single-output.

Example 2.
$$\dot{x}(t) = x(t) + u(t) + w(t) \qquad x(t) \in \mathbb{R}$$
$$y(t) = x(t) + v(t)$$

- **not** open-loop stable
- **no** reduced-order stabilizing controller
- single-input single-output



Connectivity of the Domain

Theorem 2. Under some standard assumptions,

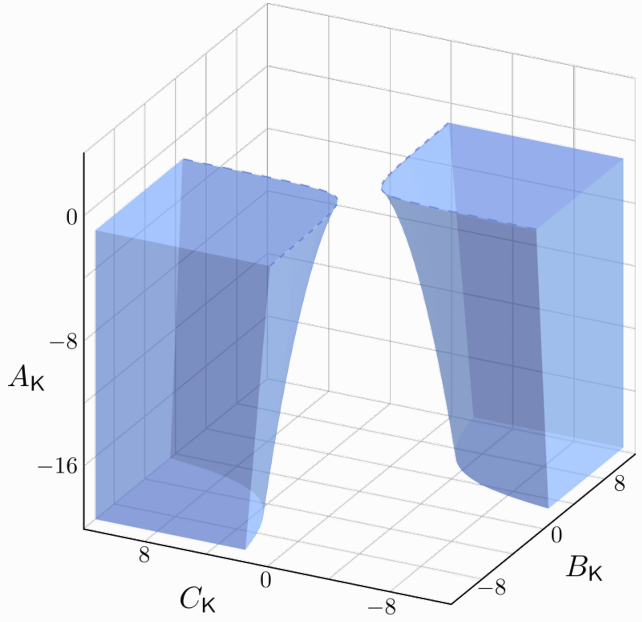
1) $\mathcal{C}_{\text{full}}$ is connected if the plant is open-loop stable or there exists a reduced-order stabilizing controller.

2) The sufficient condition of connectivity in 1) becomes necessary if the plant is single-input or single-output.

Example 2.
$$\dot{x}(t) = x(t) + u(t) + w(t) \qquad x(t) \in \mathbb{R}$$
$$y(t) = x(t) + v(t)$$

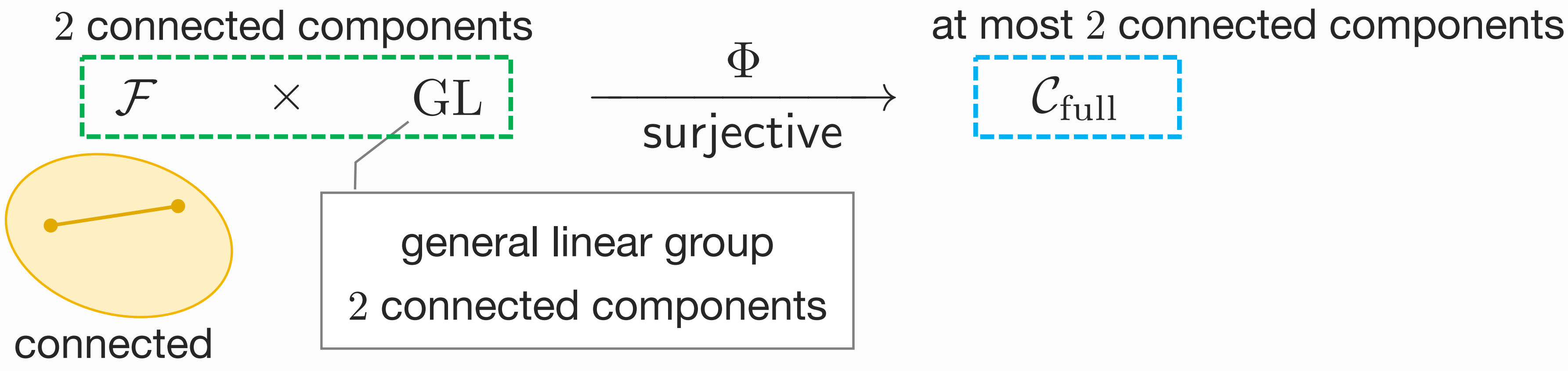
The two connected components:

$$\mathcal{C}_1^+ = \{(A_\mathsf{K}, B_\mathsf{K}, C_\mathsf{K}) \in \mathbb{R}^3 | A_\mathsf{K} < -1, B_\mathsf{K} C_\mathsf{K} < A_\mathsf{K}, B_\mathsf{K} > 0\}$$
$$\mathcal{C}_1^- = \{(A_\mathsf{K}, B_\mathsf{K}, C_\mathsf{K}) \in \mathbb{R}^3 | A_\mathsf{K} < -1, B_\mathsf{K} C_\mathsf{K} < A_\mathsf{K}, B_\mathsf{K} < 0\}$$



Connectivity of the Domain – Proof Idea

Proof idea: Construct a convex set $\mathcal{F}$ and a continuous mapping Φ such that



How to construct $\mathcal{F}$ and Φ?

$$\mathcal{F} = \left\{ (X, Y, M, H, F) \middle| X, Y \in \mathbb{S}^n, M \in \mathbb{R}^{n\times n}, H \in \mathbb{R}^{n\times p}, F \in \mathbb{R}^{m\times n}, \begin{bmatrix} X & I \\ I & Y \end{bmatrix} \succ 0, \begin{bmatrix} AX+BF & A \\ M & YA+HC \end{bmatrix} + \begin{bmatrix} AX+BF & A \\ M & YA+HC \end{bmatrix}^\top \prec 0 \right\}$$

Inspired by convex reformulation of LQG in control theory [Scherer et al. 1997]

$$\begin{bmatrix} 0 & \Phi_C(\mathsf{Z}) \\ \Phi_B(\mathsf{Z}) & \Phi_A(\mathsf{Z}) \end{bmatrix} = \begin{bmatrix} I & 0 \\ YB & \Xi \end{bmatrix}^{-1} \begin{bmatrix} 0 & H \\ F & M - YAX \end{bmatrix} \begin{bmatrix} I & CX \\ 0 & \Xi^{-1}(I - YX) \end{bmatrix}$$

LQG as an Optimization Problem

$$\begin{aligned} \min_{\mathsf{K}} \quad & J(\mathsf{K}) \\ \text{s.t.} \quad & \mathsf{K} = (A_{\mathsf{K}}, B_{\mathsf{K}}, C_{\mathsf{K}}) \in \mathcal{C}_{\text{full}} \end{aligned}$$

- **Connectivity of the domain $\mathcal{C}_{\text{full}}$**
  - Is it connected? Not necessarily.
  - If not, how many connected components can it have? Two.
- **Structure of stationary points of $J(\mathsf{K})$**
  - Are there spurious (strictly suboptimal) stationary points?
  - How to check if a stationary point is globally optimal?

LQG as an Optimization Problem

$$\begin{aligned} \min_{\mathsf{K}} \quad & J(\mathsf{K}) \\ \text{s.t.} \quad & \mathsf{K} = (A_{\mathsf{K}}, B_{\mathsf{K}}, C_{\mathsf{K}}) \in \mathcal{C}_{\text{full}} \end{aligned}$$

- Connectivity of the domain $\mathcal{C}_{\text{full}}$
  - Is it connected? Not necessarily.
  - If not, how many connected components can it have? Two.
- **Structure of stationary points of $J(\mathsf{K})$**
  - Are there spurious (strictly suboptimal) stationary points?
  - How to check if a stationary point is globally optimal?

Structure of Stationary Points

Proposition.

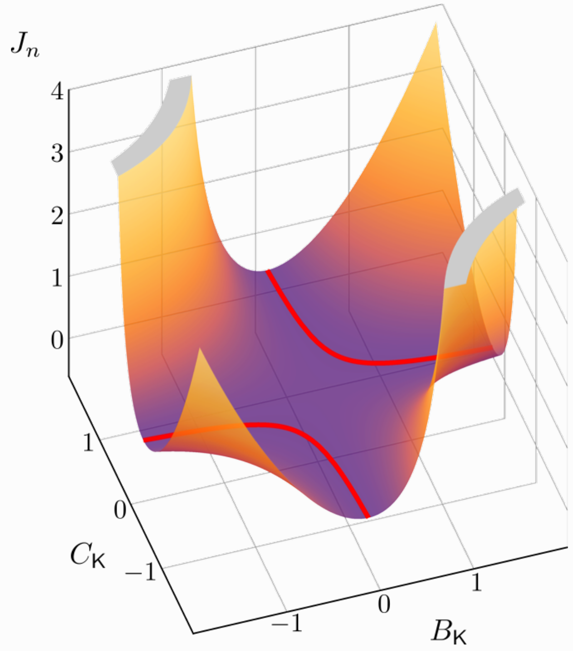
1) $J(\mathsf{K})$ is a real analytic function over its domain
2) $J(\mathsf{K})$ has **non-unique** and **non-isolated** global optima

Similarity transformation

$$(A_{\mathsf{K}}, B_{\mathsf{K}}, C_{\mathsf{K}}) \mapsto (TA_{\mathsf{K}}T^{-1}, TB_{\mathsf{K}}, C_{\mathsf{K}}T^{-1})$$

$$\dot{\xi}(t) = A_{\mathsf{K}}\,\xi(t) + B_{\mathsf{K}}\,y(t)$$
$$u(t) = C_{\mathsf{K}}\,\xi(t)$$

- $J(\mathsf{K})$ is invariant under similarity transformations.

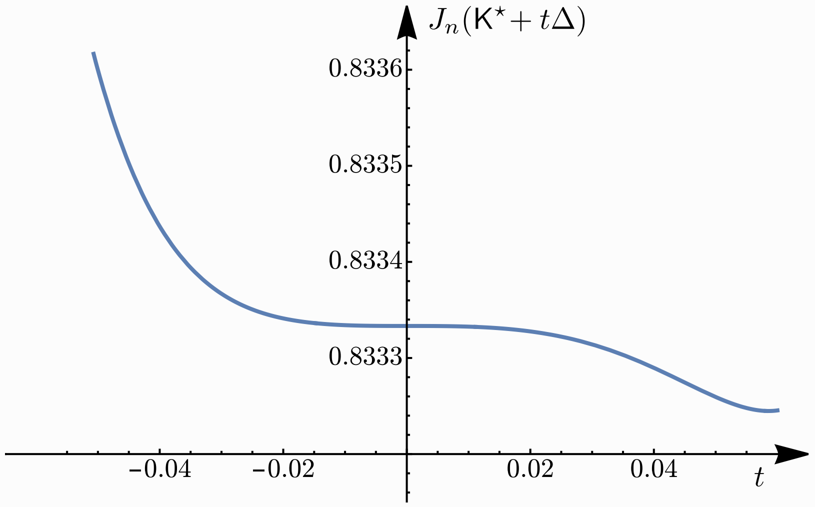


Structure of Stationary Points

Proposition.

1) $J(\mathsf{K})$ is a real analytic function over its domain
2) $J(\mathsf{K})$ has **non-unique** and **non-isolated** global optima
3) $J(\mathsf{K})$ will have **spurious** stationary points if the system is open-loop stable

➢ There may even exist saddle points with a vanishing Hessian.



Structure of Stationary Points

Proposition.

1) $J(\mathrm{K})$ is a real analytic function over its domain
2) $J(\mathrm{K})$ has **non-unique** and **non-isolated** global optima
3) $J(\mathrm{K})$ will have **spurious** stationary points if the system is open-loop stable
4) $J(\mathrm{K})$ is not coercive

Structure of Stationary Points

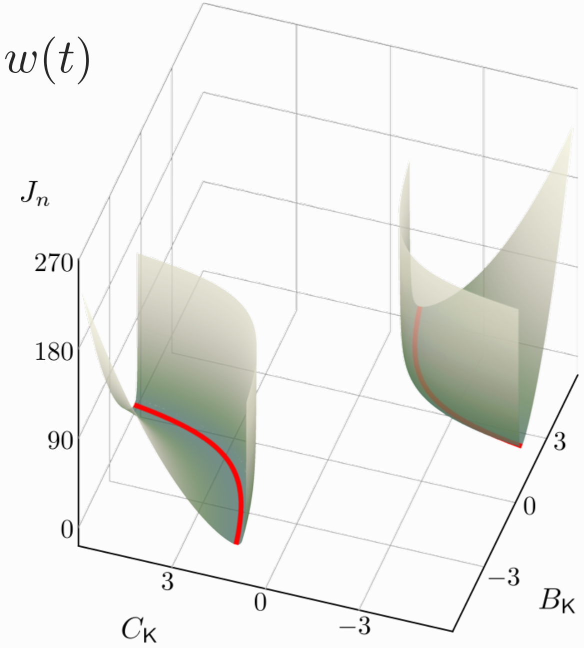
Theorem 3. Suppose there exists a stationary point that is a **minimal** controller. Then

1) This stationary point is a global optimum of $J(\mathsf{K})$
2) The set of all global optima forms a manifold with 2 connected components.

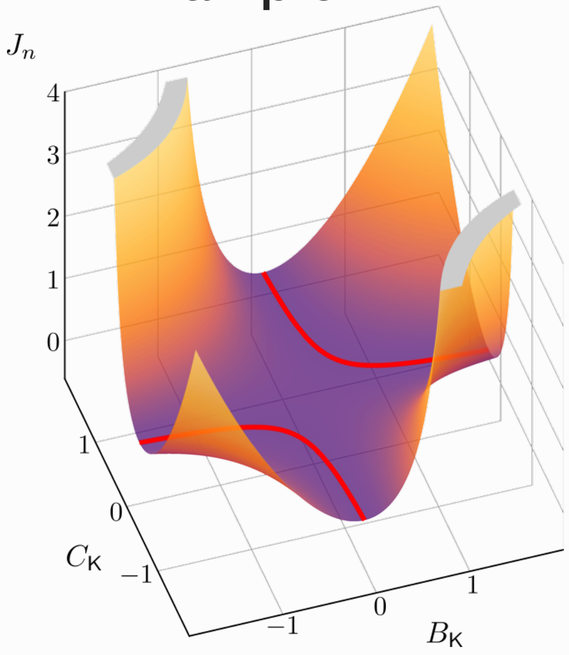
Example 1

$$\dot{x}(t) = x(t) + u(t) + w(t)$$
$$y(t) = x(t) + v(t)$$

$$x(t) \in \mathbb{R}$$



Example 2



$$\dot{x}(t) = -x(t) + u(t) + w(t)$$
$$y(t) = x(t) + v(t)$$

$$x(t) \in \mathbb{R}$$

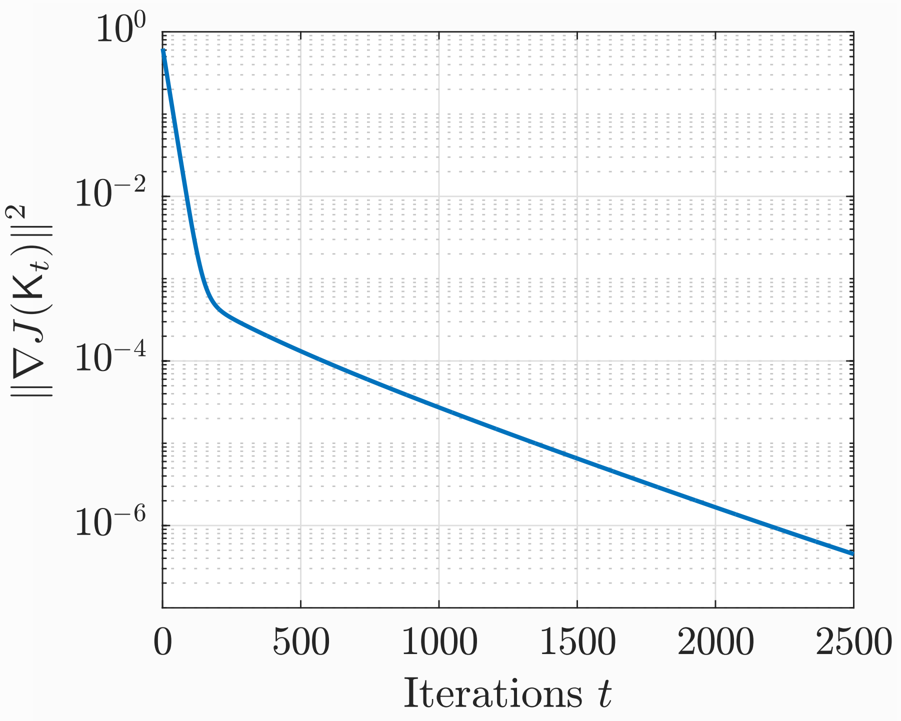
Structure of Stationary Points

Implication.

Consider gradient descent iterations

$$\mathsf{K}_{t+1} = \mathsf{K}_t - \alpha \nabla J(\mathsf{K}_t)$$

If the iterates converge to a minimal controller, then this minimal controller is a global optimum.



---

* How to check if a controller is minimal?

- Check its controllability and observability.

Summary

LQG as an optimization problem

Partial & noisy system measurement

$$\min_{\mathsf{K}} \quad J(\mathsf{K})$$
$$\text{s.t.} \quad \mathsf{K} = (A_{\mathsf{K}}, B_{\mathsf{K}}, C_{\mathsf{K}}) \in \mathcal{C}_{\text{full}}$$

Connectivity of domain

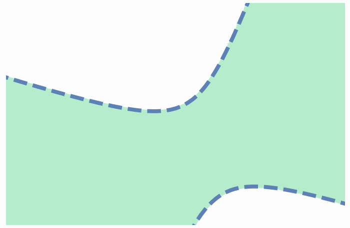
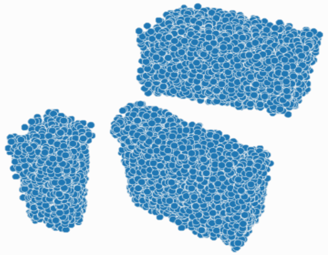
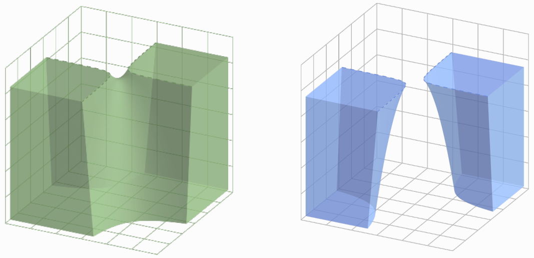
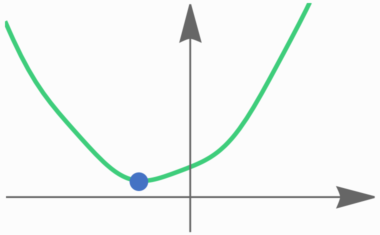
- At most **two** connected components
- The two connected components mirror each other
- Conditions for being connected

Stationary points

- **Non-unique** global optima, **spurious** stationary points
- **Minimal** stationary points are globally optimal

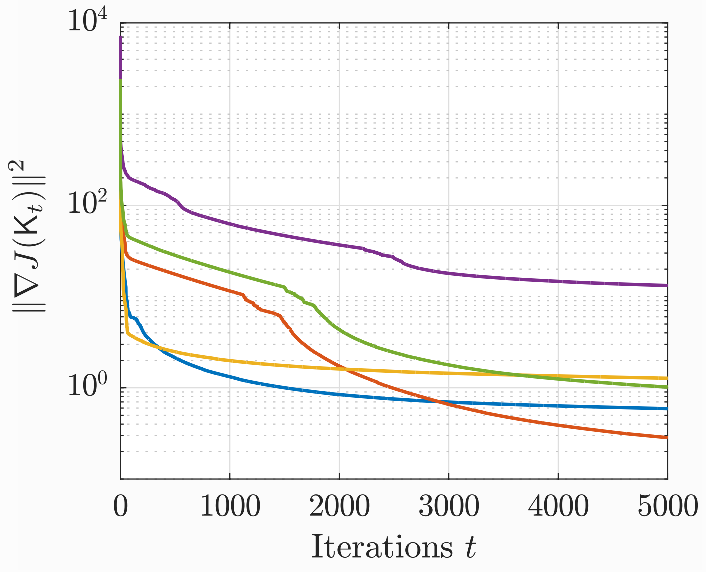
More results are presented in arXiv:2102.04393.

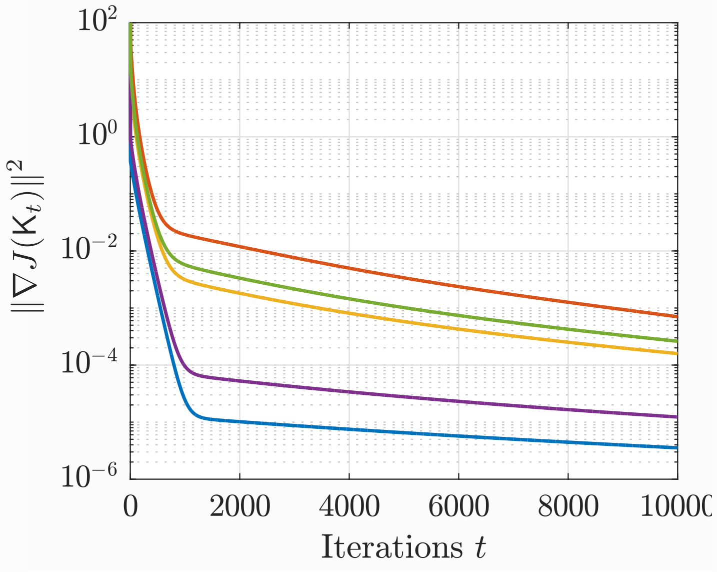
Summary

	Centralized LQR	Single-agent, partial measurement, $u(t) = K\,y(t)$	Single-agent, partial & noisy measurement, dynamic controller
Domain	Nonconvex, connected 	Multiple connected components  [Feng & Lavaei 2019]	Nonconvex, at most 2 connected components 
$J(\mathrm{K})$	• Coercive • Gradient dominance • Unique stationary point 	• Coercive • Not gradient dominance • Multiple stationary points • Lacks good properties	• Not coercive • Spurious stationary points, non-strict saddle points • Sufficient condition for checking global optimality

Future Directions

- A comprehensive classification of stationary points
- Conditions for existence of minimal globally optimal controllers
- Saddle points with vanishing Hessians may exist. How to deal with them?
- Alternative model-free parametrization of dynamic controllers
  - Better optimization landscape structures, smaller dimension





Future Directions

- A comprehensive classification of stationary points
- Conditions for existence of minimal globally optimal controllers
- Saddle points with vanishing Hessians may exist. How to deal with them?
- Alternative model-free parametrization of dynamic controllers
  - Better optimization landscape structures, smaller dimension
- Extension to multi-agent settings?
  - Should agents also exchange their measurements $y_i(t)$?
  - Effects of delays?

References

[Fazel et al. 2018] M. Fazel, R. Ge, S. Kakade, and M. Mesbahi. Global convergence of policy gradient methods for the linear quadratic regulator. In *Proceedings of the 35th International Conference on Machine Learning*, pages 1467–1476, 2018.

[Malik et al. 2019] D. Malik, A. Pananjady, K. Bhatia, K. Khamaru, P. Bartlett, and M. Wainwright. Derivative-free methods for policy optimization: Guarantees for linear quadratic systems. In *the 22nd International Conference on Artificial Intelligence and Statistics*, pages 2916–2925, 2019.

[Mohammadi et al. 2019] H. Mohammadi, A. Zare, M. Soltanolkotabi, and M. R. Jovanović. Convergence and sample complexity of gradient methods for the model-free linear quadratic regulator problem. arXiv:1912.11899, 2019.

[Zhang et al. 2019] K. Zhang, B. Hu, and T. Basar. Policy optimization for H_2 linear control with H_∞ robustness guarantee: Implicit regularization and global convergence. arXiv:1910.09496, 2019.

[Zhao & You, 2021] F. Zhao, and K. You. Primal-dual learning for the model-free risk-constrained linear quadratic regulator. In *Proceedings of the 3rd Conference on Learning for Dynamics and Control*, pages 702–714, 2021.

[Flaxman et al. 2005] A. D. Flaxman, A. T. Kalai, and H. B. McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. In Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms, pages 385–394, 2005.

[Nesterov & Spokoiny 2017] Y. Nesterov and V. Spokoiny. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, vol. 17, no. 2, pages 527–566, 2017.

[Feng & Lavaei 2019] H. Feng and J. Lavaei. On the exponential number of connected components for the feasible set of optimal decentralized control problems. In *2019 American Control Conference*, pages 1430–1437, 2019

[Bu et al. 2019] J. Bu, A. Mesbahi, M. Fazel, and M. Mesbahi. LQR through the lens of first order methods: Discretetime case. arXiv:1907.08921, 2019.

[Scherer et al. 1997] C. Scherer, P. Gahinet, and M. Chilali. Multiobjective output-feedback control via LMI optimization. IEEE Transactions on Automatic Control, vol. 42, no. 7, pages 896–911, 1997.