Learning, Optimization, and Control for Large-scale Autonomous Systems

Yang Zheng

Assistant Professor,
ECE Department, UC San Diego

ECE 290 Seminar Series
October 22, 2021

UC San Diego
JACOBS SCHOOL OF ENGINEERING
Electrical and Computer Engineering

SOC Lab

Acknowledgements

Tsinghua University


Keqiang Li


Jiawei Wang

University of Oxford


Antonis Papachristodoulou

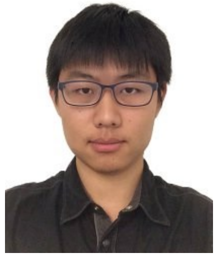

Paul Goulart


Aivar Sootla


Ross Drummond

Harvard


Na Li


Yujie Tang

Imperial College London

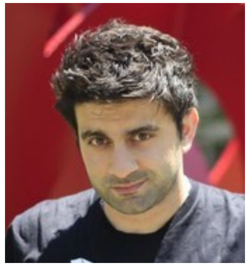

Giovanni Fantuzzi


Panagiotis Kouvaros


Ben Batten


Alessio Lomuscio

Princeton


Amir Ali Ahmadi


Georgina Hall

Northeastern University


Mario Sznaier

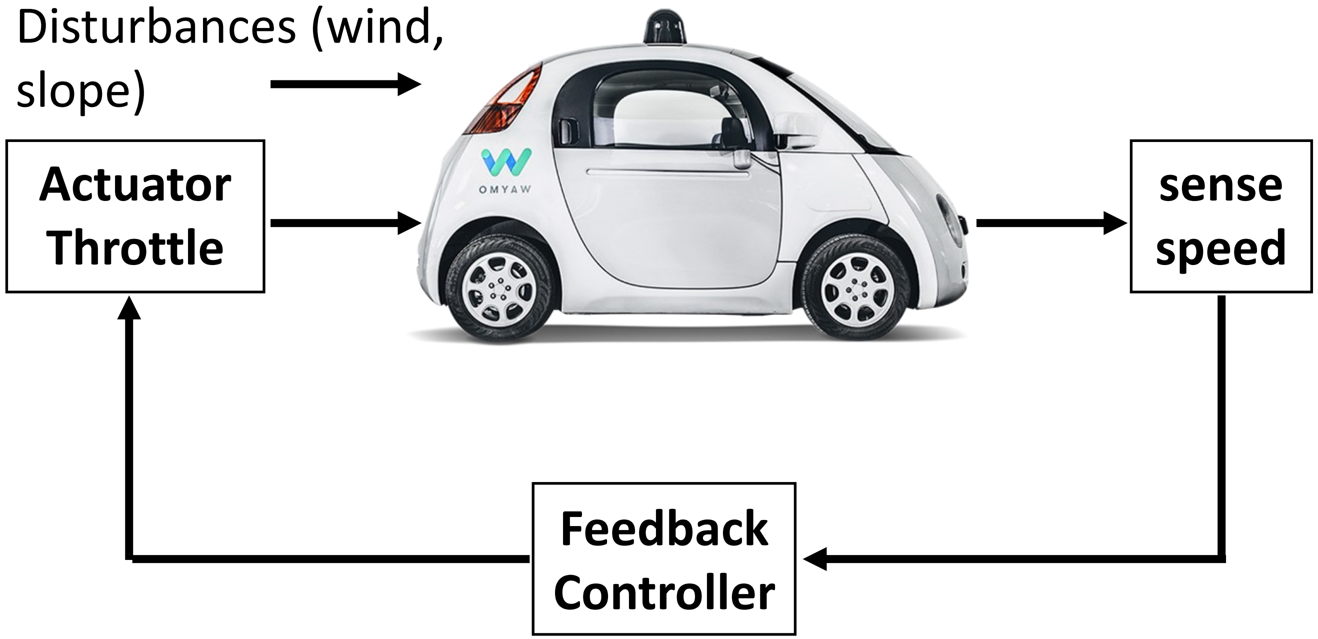
EPFL


Maryam Kamgarpour

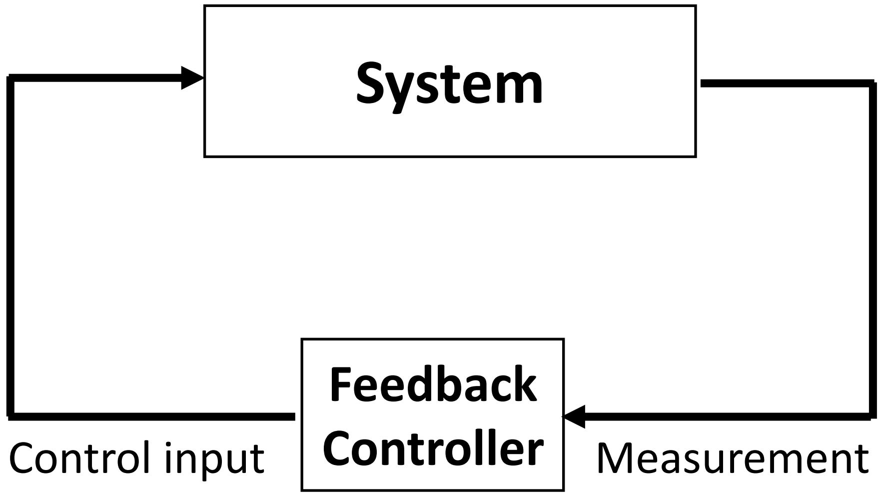

Luca Furieri

Automatic control example

Highway cruise control



Feedback Paradigm



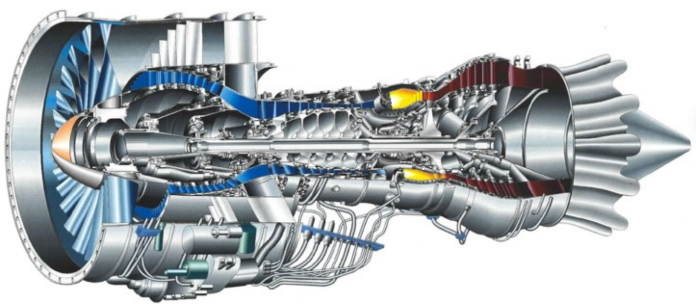
Control theory: the principled use of feedback loops and algorithms to drive a system to its goal

- "Simple" centralized control systems are well understood.
- "Complexity" can enter in different ways . . .

Complex autonomous systems

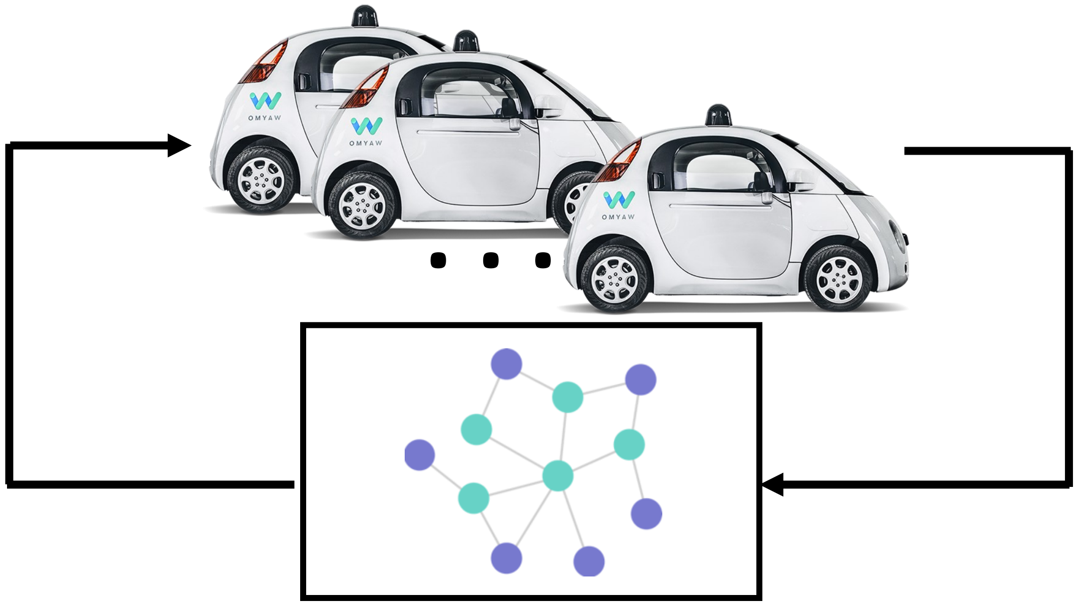
❑ Complex nonlinear dynamics

- Aircraft, jet engine, robotics



❑ Complex distributed systems

- Multiple subsystems & local commutation

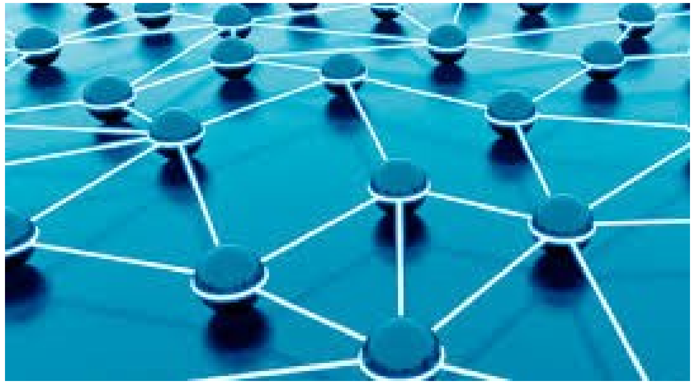


Distributed controller

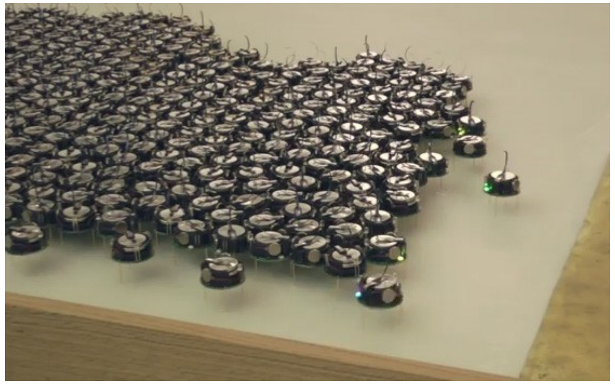
Source: https://solidmechanicsproblems.wordpress.com/;
https://www.bostondynamics.com/

Examples of large-scale autonomous systems

Drone formations

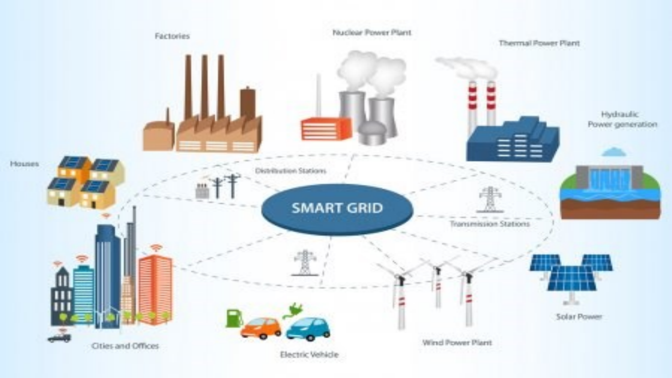


Sensor networks



Robotic networks

Transportation network



Smart grid

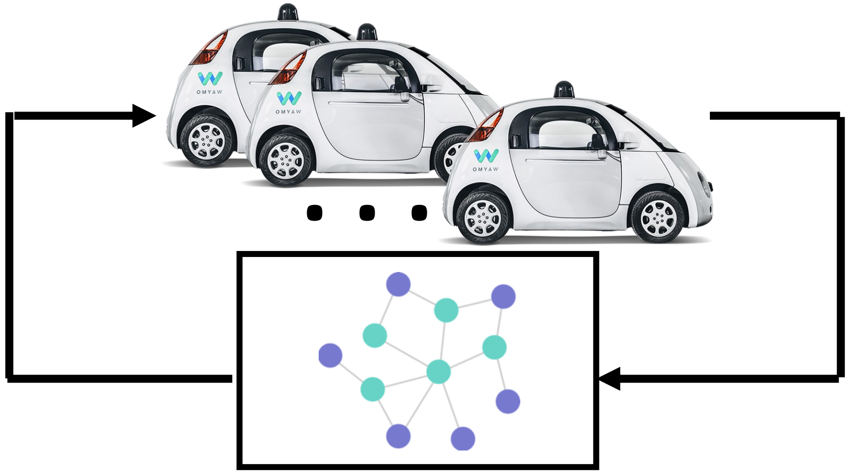
Self-organization

Distributed control laws → ***Desired collective behavior***

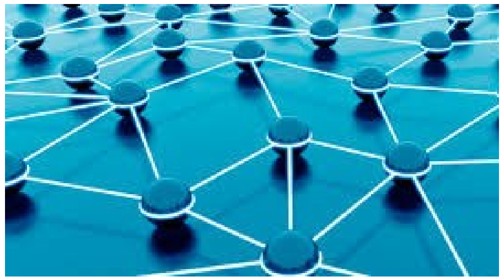
Image Credit: Getty Images; Google Images; www.pinterest.com

Challenges

- ❑ **Model uncertainty** ➡ Learning-based & Robust control
  - Model might be unknown for practical systems;
  - Model might be uncertain; **Learning-based** solutions
- ❑ **Information constraints** ➡ Distributed control
  - Large numbers of components;
  - Subsystems or components may have dynamic coupling;
  - Only **local information available** for control decision;
- ❑ **High dimensional problems** ➡ Scalable Optimization
  - A very large number of states and control variables;
  - Require to solve **large-scale optimization** efficiently;
- ❑ **Real world applications** ➡ Mixed traffic control



Distributed controller

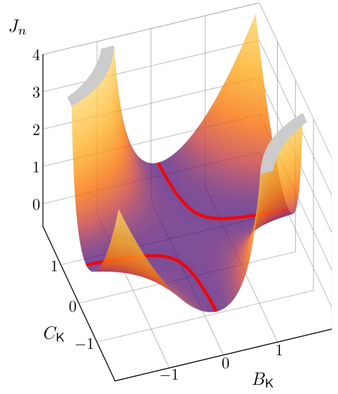


Scalable Optimization & Control (SOC) Lab

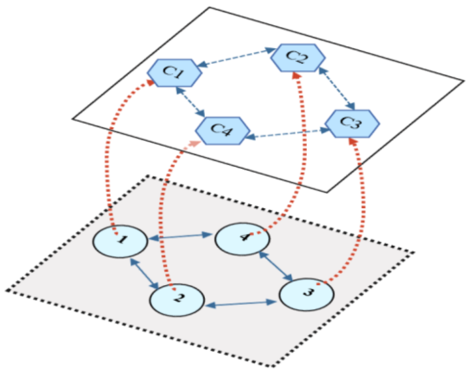
Scalable Optimization and Control (SOC) lab

SOC Lab

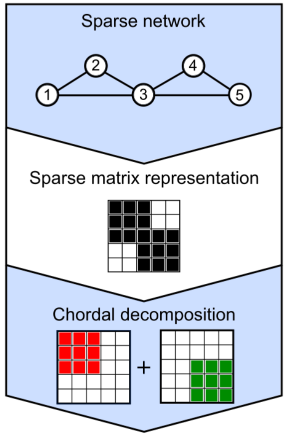
Data-driven and learning-based control



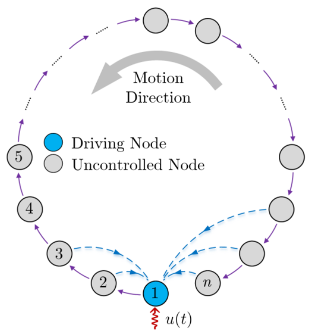
Scalable distributed control



Sparse conic optimization



Connected and autonomous vehicles (CAVs)



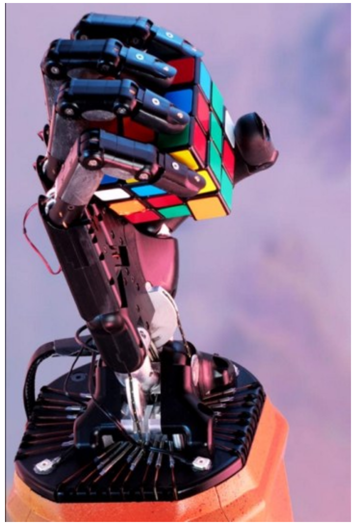
Zheng, Yang, Yujie Tang, and Na Li. "Analysis of the Optimization Landscape of Linear Quadratic Gaussian (LQG) Control." *arXiv preprint arXiv:2102.04393* (2021).

Motivation

❑ Model-free methods and data-driven control

- Use direct policy updates;
- Become very popular in both academia and practice, from game playing, robotics, and drones, etc.

DeepMind



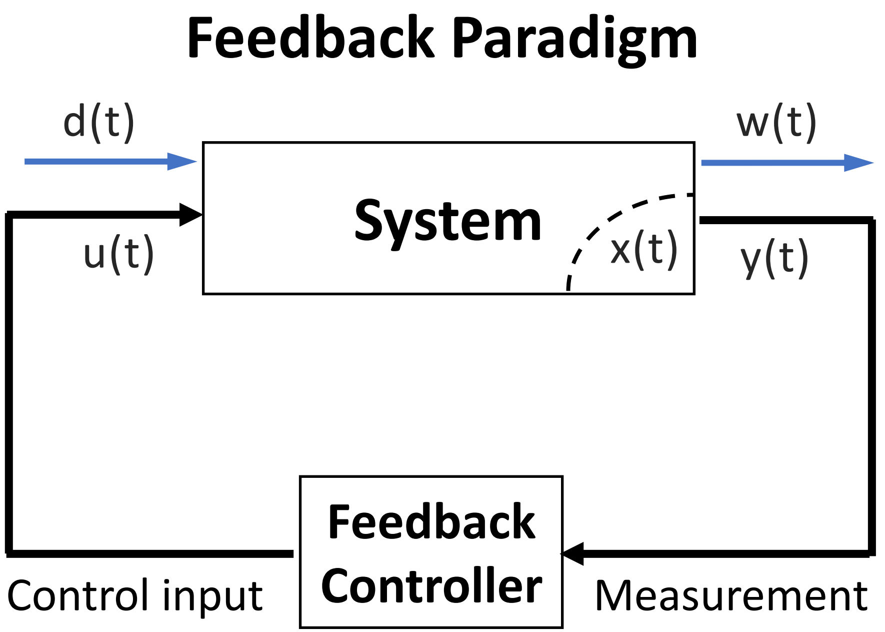
OpenAI

Applications

- lack non-asymptotic performance guarantees, such as sample complexity, safety, suboptimality, convergence etc. → **linear dynamical systems!**

This talk

Linear Quadratic Optimal control



Major challenge: how to perform optimal control when the system is unknown?

$$\min_{u_1, u_2, \ldots,} \quad \lim_{T \to \infty} \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\left(x_t^{\mathsf{T}} Q x_t + u_t^{\mathsf{T}} R u_t\right)\right]$$

$$\text{subject to} \quad x_{t+1} = Ax_t + Bu_t + w_t$$

$$y_t = Cx_t + v_t$$

- Many practical applications
- Extensive classical results (Dynamic programming, Separation principle, Riccati equations, etc)
- **Linear Quadratic Regulator (LQR)** when the state x_t is directly observable
- **LQG** when only partial output y_t is observed

Two main approaches

❑ Model-free: Direct policy iteration

- Give a parameterization of control policies; say **neural networks?** ✖
- Control theory already tells us many structural properties: **Linear feedback is sufficient for LQR**

$$u_t = K x_t$$

$$\lim_{T\to\infty} \mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\left(x_t^{\mathsf{T}} Q x_t + u_t^{\mathsf{T}} R u_t\right)\right] := J(K)$$

Set of stabilizing controllers: $K \in \mathcal{K}$

LQR as an Optimization problem

$$\min_{K} \quad J(K)$$
$$\text{s.t.} \quad K \in \mathcal{K}$$

Direct policy iteration

$$K_{i+1} = K_i - \alpha_i \nabla J(K_i)$$

✓ Good Landscape properties
- Connected feasible region
- Unique stationary point
- Gradient dominance

✓ Fast global convergence (exponential)

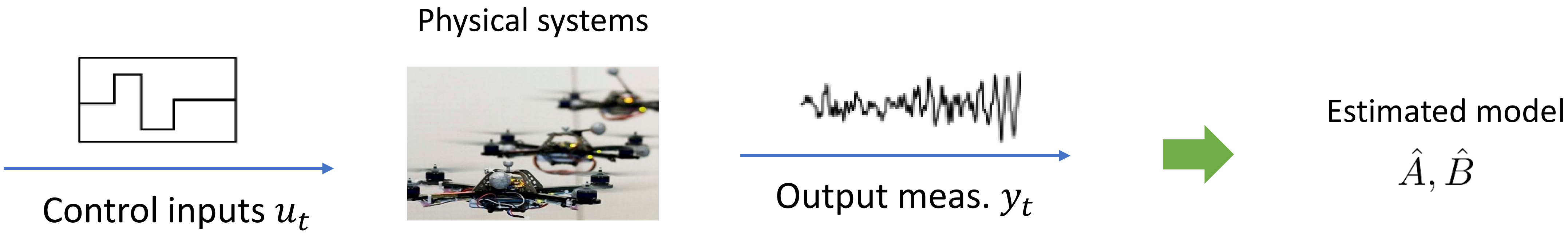
A fast-growing list of references

- Fazel et al., ICML, 2018; Malik et al., 2019; Mohammadi et al., IEEE TAC, 2020; Li et al., 2019; Zhang et al., 2019; Furieri et al., 2019; Feiran Zhao & Keyou You, 2021, and many others

Two main approaches

❑ Model-based: Sys ID + robust control

- System ID + certainty equivalent control → adaptive control (Åström & Wittenmark, 2013).



- Recent works → robust stability guarantees and sample complexity results, LQR problems (so-called system-level parameterization, Wang, Matni & Doyle, TAC, 2019)

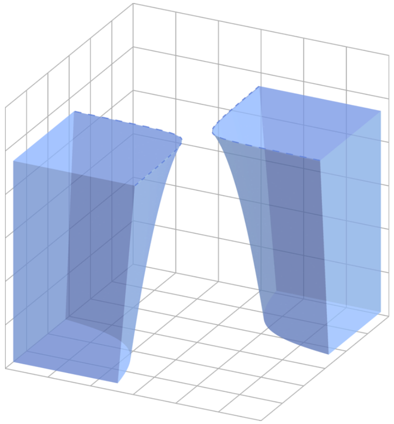
Estimated model + uncertainty $\hat{A} + \Delta A, \quad \hat{B} + \Delta B, \qquad \|\Delta A\| \leq \epsilon_A, \|\Delta B\| \leq \epsilon_B,$

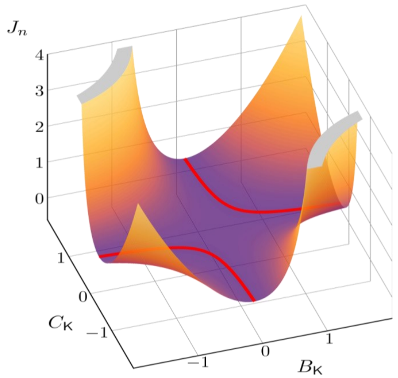
✓ Dean et al., 2020; Berberich et al., 2020; Boczar et al., 2018; Tsiamis et al., 2020; Umenberger et al., 2019; Yiwen Lu and Yilin Mo, 2021, and many others

Challenges for partially observed LQG

❑ Results on model-free or model-based LQG control are much fewer

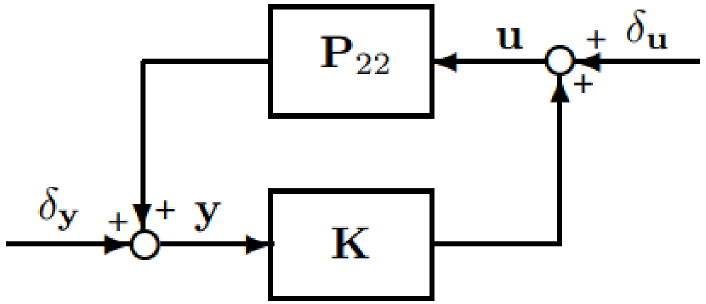
- LQG is more sophisticated than LQR
- Requires dynamical controllers
- Its landscape properties are much richer and more complicated than LQR





Topic 1 Landscape Analysis

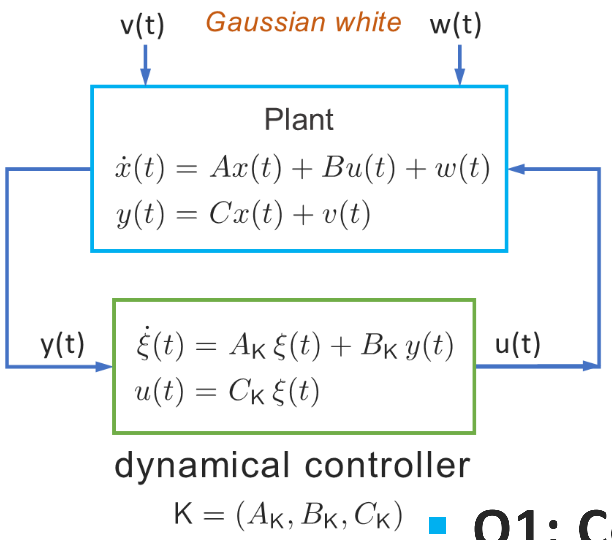
- The underlying technique, **system-level parameterization**, becomes non-trivial to use for the LQG case
- New techniques based on Input-output parameterization (IOP) (Furieri et al., 2019), are used for learning a robust LQG controller



Topic 2 Sample complexity

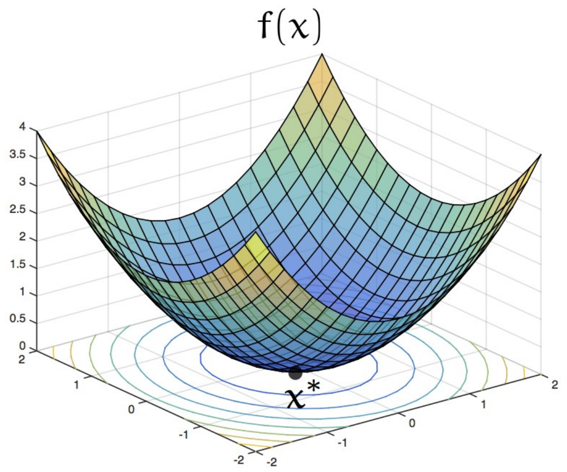
Zheng, Y., Furieri, L., Kamgarpour, M., & Li, N. (2021, May). Sample complexity of linear quadratic gaussian (LQG) control for output feedback systems. In *Learning for Dynamics and Control* (pp. 559-570). PMLR.

Model-free Optimization formulation



LQG as an Optimization problem

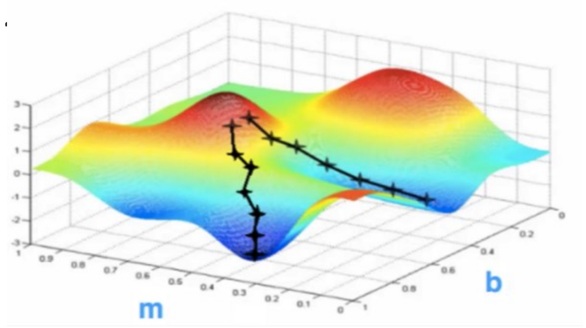
$$\min_{\mathsf{K}} \quad J(\mathsf{K})$$
$$\text{s.t.} \quad \mathsf{K} = (A_{\mathsf{K}}, B_{\mathsf{K}}, C_{\mathsf{K}}) \in \mathcal{C}_{\text{full}}$$



- **Q1: Connectivity of the feasible region $\mathcal{C}_{\text{full}}$**
  - Is it connected?
  - If not, how many connected components can it have?

Landscape Analysis

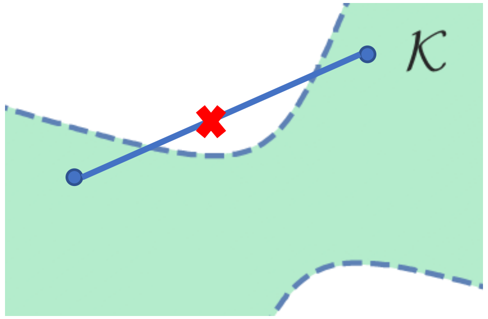
- **Q2: Structure of stationary points of $J(\mathsf{K})$**
  - Are there spurious (strictly suboptimal, saddle) stationary points?
  - How to check if a stationary point is globally optimal?



Connectivity of the feasible region

- **Simple observation: non-convex and unbounded**

Lemma 1: the set $\mathcal{C}_{\text{full}}$ is non-empty, unbounded, and can be non-convex.



Example:

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

$$\mathcal{C}_{\text{full}} = \left\{ \mathsf{K} = \begin{bmatrix} 0 & C_{\mathsf{K}} \\ B_{\mathsf{K}} & A_{\mathsf{K}} \end{bmatrix} \in \mathbb{R}^{2\times 2} \,\middle|\, \begin{bmatrix} 1 & C_{\mathsf{K}} \\ B_{\mathsf{K}} & A_{\mathsf{K}} \end{bmatrix} \text{ is stable} \right\}.$$

$$\mathsf{K}^{(1)} = \begin{bmatrix} 0 & 2 \\ -2 & -2 \end{bmatrix}, \qquad \mathsf{K}^{(2)} = \begin{bmatrix} 0 & -2 \\ 2 & -2 \end{bmatrix}$$

Stabilize the plant, and thus belong to $\mathcal{C}_{\text{full}}$

$$\hat{\mathsf{K}} = \frac{1}{2}\left(\mathsf{K}^{(1)} + \mathsf{K}^{(2)}\right) = \begin{bmatrix} 0 & 0 \\ 0 & -2 \end{bmatrix}$$

Fails to stabilize the plant, and thus outside $\mathcal{C}_{\text{full}}$

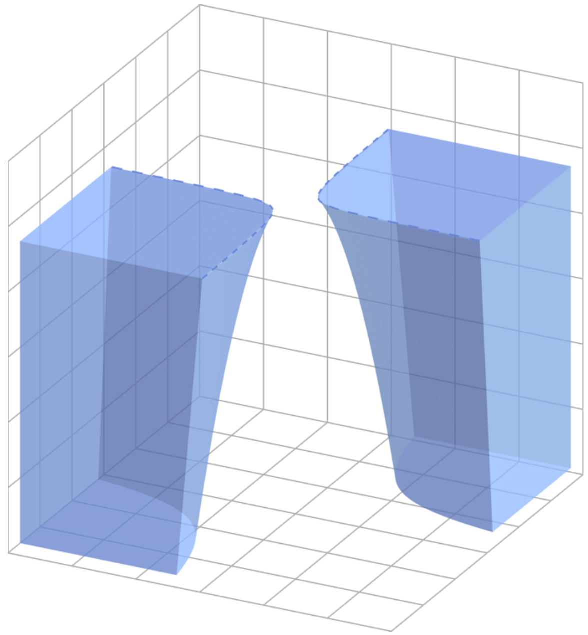
Connectivity of the feasible region

- **Main Result 1: dis-connectivity**

Theorem 1: The set $\mathcal{C}_{\rm full}$ can be disconnected but has at most 2 connected components.

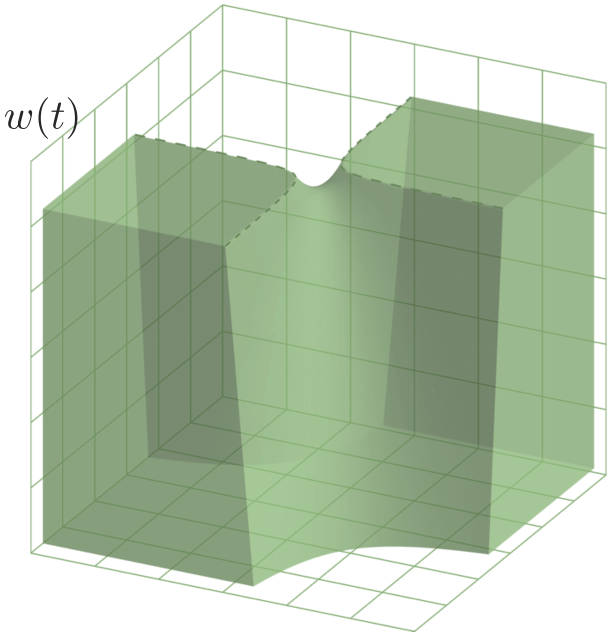
Example 1

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



Example 2

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

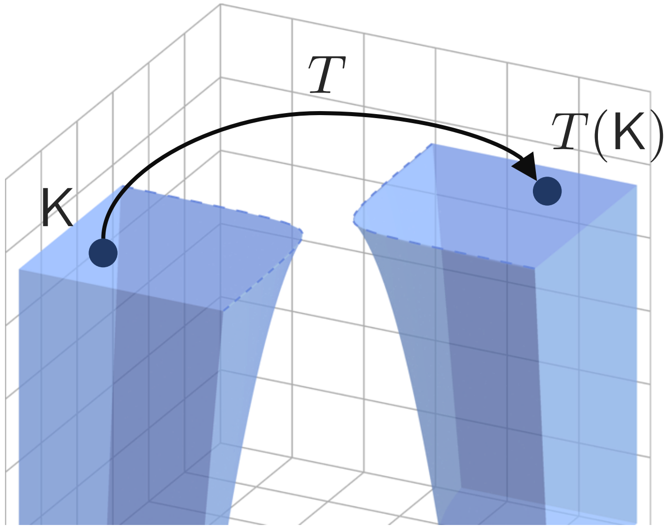


- ✓ Different from the connectivity of static stabilizing state-feedback controllers, which is always connected!
- ✓ Is this a negative result for gradient-based algorithms? → **No**

Connectivity of the feasible region

- **Main Result 2: dis-connectivity**

Theorem 2: If $\mathcal{C}_{\text{full}}$ has 2 connected components, then there is a smooth bijection T between the 2 connected components that has the same cost function value



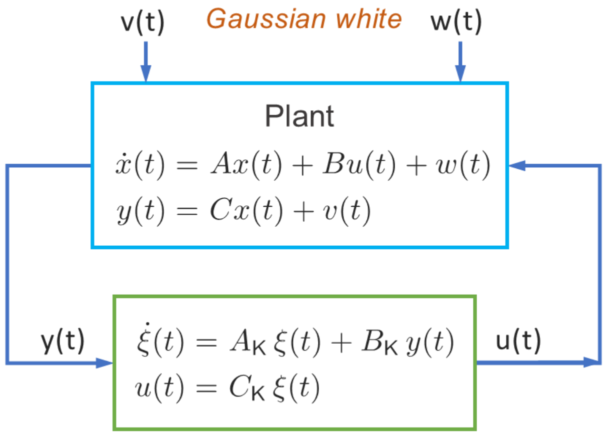
$$J(\mathsf{K}) = J(T(\mathsf{K}))$$

✓ In fact, the bijection T is defined by a similarity transformation (change of controller state coordinate)

$$\mathscr{T}_T(\mathsf{K}) := \begin{bmatrix} D_{\mathsf{K}} & C_{\mathsf{K}}T^{-1} \\ TB_{\mathsf{K}} & TA_{\mathsf{K}}T^{-1} \end{bmatrix}.$$

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

Model-free Optimization formulation



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

Landscape Analysis

LQG as an Optimization problem

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

- Q1: Connectivity of the feasible region $\mathcal{C}_{\text{full}}$
  - Is it connected? **No**
  - How many connected components can it have? **Two**
- Q2: Structure of stationary points of $J(\mathsf{K})$
  - Are there spurious (strictly suboptimal, saddle) stationary points?
  - How to check if a stationary point is globally optimal?

Structure of Stationary Points

LQG as an Optimization problem

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

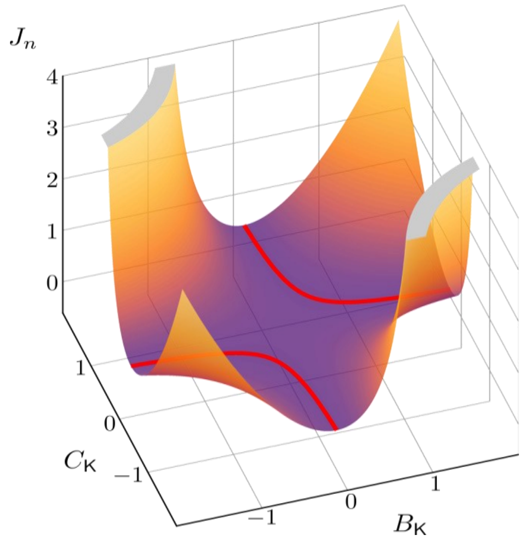
- **Simple observations**

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

Similarity transformation

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

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



- $J(\mathsf{K})$ is invariant under similarity transformations.
- It has many stationary points, unlike the LQR with a unique stationary point

Structure of Stationary Points

- Gradient computation

Lemma 1: For every $\mathsf{K} = (A_{\mathsf{K}}, B_{\mathsf{K}}, C_{\mathsf{K}}) \in \mathcal{C}_{\text{full}}$, we have

$$\frac{\partial J(\mathsf{K})}{\partial A_{\mathsf{K}}} = 2\left(Y_{12}^{\mathsf{T}} X_{12} + Y_{22} X_{22}\right),$$
$$\frac{\partial J(\mathsf{K})}{\partial B_{\mathsf{K}}} = 2\left(Y_{22} B_{\mathsf{K}} V + Y_{22} X_{12}^{\mathsf{T}} C^{\mathsf{T}} + Y_{12}^{\mathsf{T}} X_{11} C^{\mathsf{T}}\right),$$
$$\frac{\partial J(\mathsf{K})}{\partial C_{\mathsf{K}}} = 2\left(R C_{\mathsf{K}} X_{22} + B^{\mathsf{T}} Y_{11} X_{12} + B^{\mathsf{T}} Y_{12} X_{22}\right),$$

where $X_{\mathsf{K}} = \begin{bmatrix} X_{11} & X_{12} \\ X_{12}^{\mathsf{T}} & X_{22} \end{bmatrix}, \qquad Y_{\mathsf{K}} = \begin{bmatrix} Y_{11} & Y_{12} \\ Y_{12}^{\mathsf{T}} & Y_{22} \end{bmatrix}$

are the unique solutions to two Lyapunov equations

LQG as an Optimization problem

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

How does the set of Stationary Points look like?

$$\left\{ \mathsf{K} \in \mathcal{C}_{\text{full}} \;\middle|\; \frac{\partial J(\mathsf{K})}{\partial A_{\mathsf{K}}} = 0, \; \frac{\partial J(\mathsf{K})}{\partial B_{\mathsf{K}}} = 0, \; \frac{\partial J(\mathsf{K})}{\partial C_{\mathsf{K}}} = 0, \right\}$$

- Non-unique, non-isolated
- Local minimum, local maximum, saddle points, or globally minimum?

Structure of Stationary Points

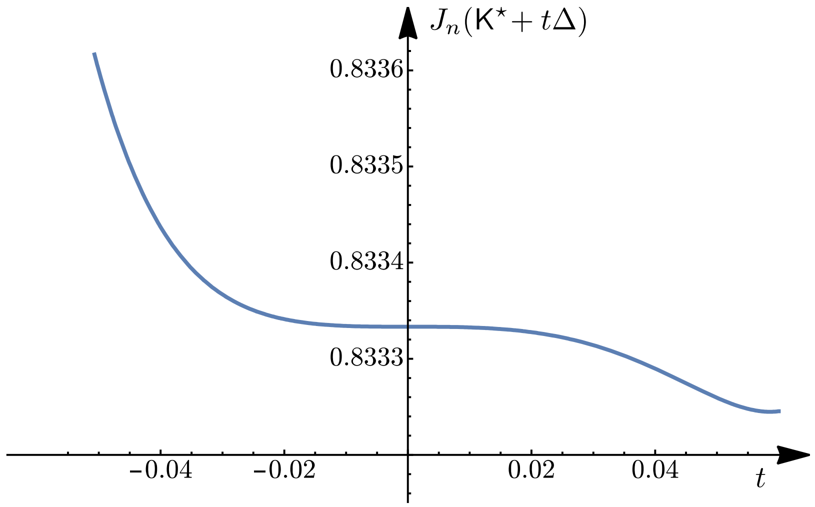
Main Result

Theorem 4: Consider any open-loop stable plant. The zero controller with any stable A_{K}

$$\mathsf{K} = (A_{\mathsf{K}}, 0, 0) \in \mathcal{C}_{\text{full}}$$

is a stationary point. Furthermore, the corresponding hessian is either indefinite (strict saddle point) or equal to zero.

Another example with zero Hessian



All bad stationary points correspond to non-minimal controllers

$$\left\{ \mathsf{K} \in \mathcal{C}_{\text{full}} \;\middle|\; \begin{aligned} \frac{\partial J(\mathsf{K})}{\partial A_{\mathsf{K}}} &= 0, \\ \frac{\partial J(\mathsf{K})}{\partial B_{\mathsf{K}}} &= 0, \\ \frac{\partial J(\mathsf{K})}{\partial C_{\mathsf{K}}} &= 0, \end{aligned} \right\}$$

Structure of Stationary Points

- **Main Result**

Theorem 5: **All stationary points corresponding to controllable and observable controllers are globally minimal!!**

$$\left\{ \mathsf{K} \in \mathcal{C}_{\text{full}} \;\middle|\; \frac{\partial J(\mathsf{K})}{\partial A_{\mathsf{K}}} = 0, \; \frac{\partial J(\mathsf{K})}{\partial B_{\mathsf{K}}} = 0, \; \frac{\partial J(\mathsf{K})}{\partial C_{\mathsf{K}}} = 0, \right\}$$

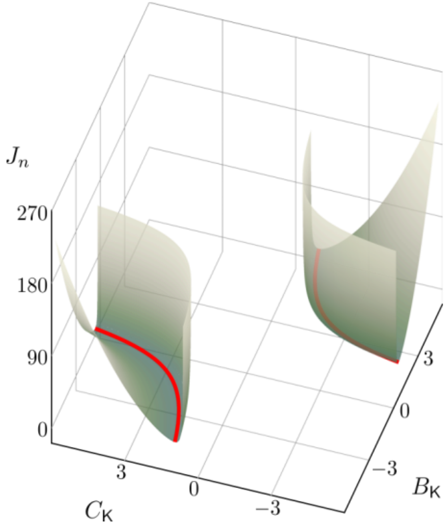
Particularly, given a stationary point that is a **minimal** controller

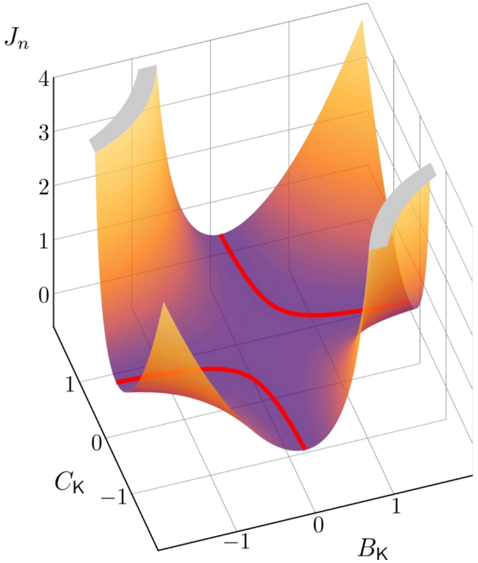
1) This stationary point is a global optimum of $J(\mathsf{K})$
2) The set of all global optima forms a manifold with 2 connected components. They are connected by a similarity transformation.

Example 1

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

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





Example 2

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

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

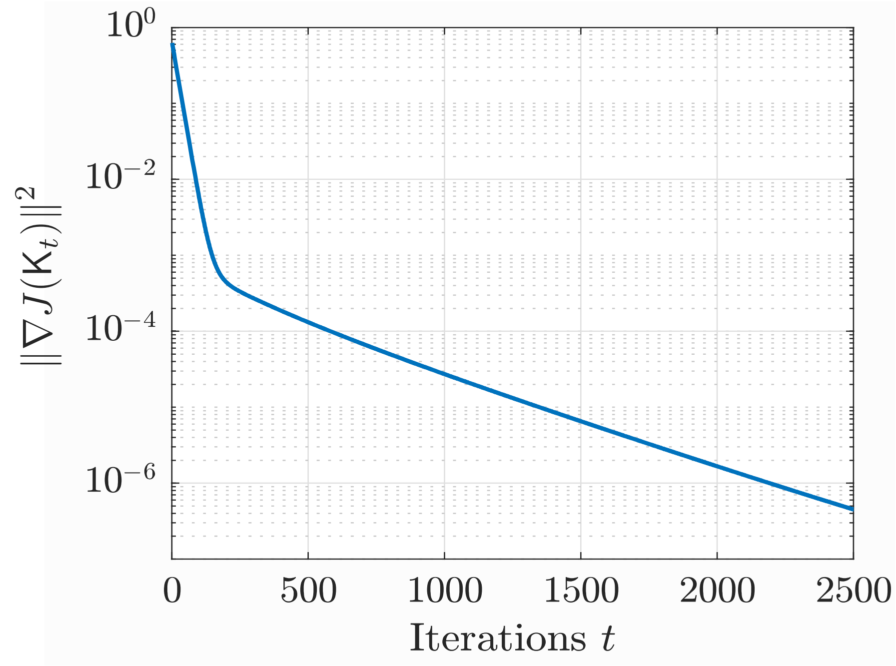
Structure of Stationary Points

❑ Implication

Consider gradient descent iterations

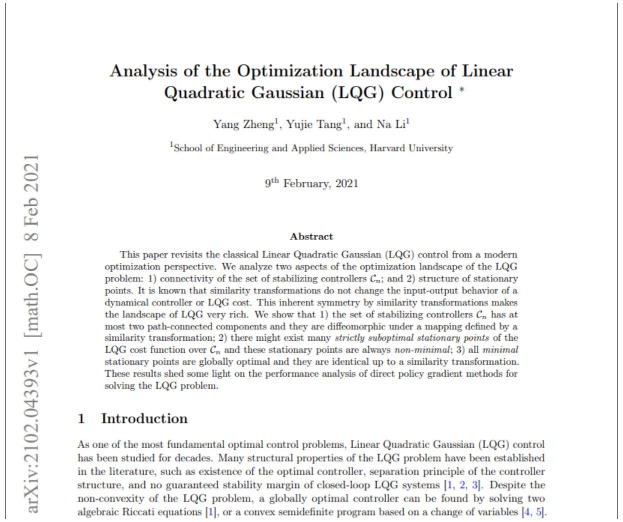
$$\mathsf{K}_{t+1} = \mathsf{K}_t - \alpha \nabla J(\mathsf{K}_t)$$

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



Open questions:

- ✓ Convergence conditions?
- ✓ Convergence speed?
- ✓ Alternative model-free parameterization



arXiv:2102.04393v1 [math.OC] 8 Feb 2021

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

Yang Zheng[1], Yujie Tang[1], and Na Li[1]

[1]School of Engineering and Applied Sciences, Harvard University

9th February, 2021

Abstract

This paper revisits the classical Linear Quadratic Gaussian (LQG) control from a modern optimization perspective. We analyze two aspects of the optimization landscape of the LQG problem: 1) connectivity of the set of stabilizing controllers $\mathcal{C}_n$; and 2) structure of stationary points. It is known that similarity transformations do not change the input-output behavior of a dynamical controller or LQG cost. This inherent symmetry by similarity transformations makes the landscape of LQG very rich. We show that 1) the set of stabilizing controllers $\mathcal{C}_n$ has at most two path-connected components and they are diffeomorphic under a mapping defined by a similarity transformation; 2) there might exist many *strictly suboptimal stationary points* of the LQG cost function over $\mathcal{C}_n$ and these stationary points are always *non-minimal*; 3) all *minimal* stationary points are globally optimal and they are identical up to a similarity transformation. These results shed some light on the performance analysis of direct policy gradient methods for solving the LQG problem.

1 Introduction

As one of the most fundamental optimal control problems, Linear Quadratic Gaussian (LQG) control has been studied for decades. Many structural properties of the LQG problem have been established in the literature, such as existence of the optimal controller, separation principle of the controller structure, and no guaranteed stability margin of closed-loop LQG systems [1, 2, 3]. Despite the non-convexity of the LQG problem, a globally optimal controller can be found by solving two algebraic Riccati equations [1], or a convex semidefinite program based on a change of variables [4, 5].

72 pages

Zheng, Yang, Yujie Tang, and Na Li. "Analysis of the Optimization Landscape of Linear Quadratic Gaussian (LQG) Control." *arXiv preprint arXiv:2102.04393* (2021).

Some other recent results

Model-based Learning LQG controller

2011.09929v2 [math.OC] 25 Jun 2021

Sample Complexity of Linear Quadratic Gaussian (I
for Output Feedback Systems

Yang Zheng * ZHEN(
School of Engineering and Applied Sciences, Harvard University, USA

Luca Furieri * LU(
Automatic Control Laboratory, ETH Zurich, Switzerland,
Laboratoire d'Automatique, EPFL, Switzerland

Maryam Kamgarpour MA
Electrical and Computer Engineering, University of British Columbia, Canada

Na Li NALI(
School of Engineering and Applied Sciences, Harvard University, USA

Editors: A. Jadbabaie, J. Lygeros, G. J. Pappas, P. A. Parrilo, B. Recht, C. J. Tomli

Abstract

This paper studies a class of partially observed Linear Quadratic Gaussi with *unknown* dynamics. We establish an end-to-end sample complexity bound LQG controller for open-loop stable plants. This is achieved using a robust s where we first estimate a model from a single input-output trajectory of finite le infinity bound on the estimation error, and then design a robust controller using and its quantified uncertainty. Our synthesis procedure leverages a recent contr Output Parameterization (IOP) that enables robust controller design using c For open-loop stable systems, we prove that the LQG performance degrades *l* to the model estimation error using the proposed synthesis procedure. Despi in the LQG problem, the achieved scaling matches previous results on learnin Regulator (LQR) controllers with full state observations.

Model-free Learning LQG controller in finite horizon

arXiv:1912.08774v3 [eess.SY] 30 May 2020

Pages::1–22, 2020

Learning the Globally Optimal Distributed LQ Regula

Luca Furieri FURIERIL@CONTRO
Automatic Control Laboratory, ETH Zurich, Switzerland

Yang Zheng ZHENGY@G.
School of Engineering and Applied Sciences, Harvard University, USA

Maryam Kamgarpour * MKAMGAR@CONTRO
Automatic Control Laboratory, ETH Zurich, Switzerland

Abstract

We study model-free learning methods for the output-feedback Linear Quadratic (LQ) co lem in finite-horizon subject to subspace constraints on the control policy. Subspace c naturally arise in the field of distributed control and present a significant challenge in the standard model-based optimization and learning leads to intractable numerical prograr eral. Building upon recent results in zeroth-order optimization, we establish model-fre complexity bounds for the class of distributed LQ problems where a local gradient domir stant exists on any sublevel set of the cost function. We prove that a fundamental class of control problems—commonly referred to as Quadratically Invariant (QI) problems—as w ers possess this property. To the best of our knowledge, our result is the first sample-c bound guarantee on learning globally optimal distributed output-feedback control polici

1. Introduction

Recent years have witnessed significant attention and progress in controlling unkno systems solely based on system trajectory observations. This shift from classical contr to data-driven ones is motivated by the ever increasing complexity of critical emergi systems, whose mathematical models may be unreliable or simply not available (H 2013). When it comes to learning an optimal control policy, the available approaches divided into two categories. The first class of methods is denoted as *model-based*, wh ical system data is exploited to build an approximation of the nominal system and cla robust control is then used on this system approximation. The second class of methods *model-free*, where reinforcement learning is used to directly learn an optimal control on the observed costs, without explicitly constructing a model for the system.

Non-asymptotic System Identification

arXiv:2009.00739v3 [math.OC] 8 Nov 2020

Non-asymptotic Identification of Linear Dynamical Systems Using Multiple Trajectories*

Yang Zheng[1,2] and Na Li[1,2]

[1]School of Engineering and Applied Sciences, Harvard University
[2]Harvard Center for Green Buildings and Cites, Harvard University

November 10, 2020

Abstract

This paper considers the problem of linear time-invariant (LTI) system identification using input/output data. Recent work has provided non-asymptotic results on partially observed LTI system identification using a single trajectory but is only suitable for stable systems. We provide finite-time analysis for learning Markov parameters based on the ordinary least-squares (OLS) estimator using multiple trajectories, which covers both stable and unstable systems. For unstable systems, our results suggest that the Markov parameters are harder to estimate in the presence of process noise. Without process noise, our upper bound on the estimation error is independent of the spectral radius of system dynamics with high probability. These two features are different from fully observed LTI systems for which recent work has shown that unstable systems with a bigger spectral radius are easier to estimate. Extensive numerical experiments demonstrate the performance of our OLS estimator.

1 Introduction

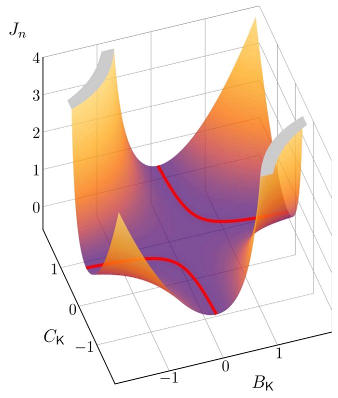
System identification estimates the models of dynamical systems from observed input-output data [1], which is an important topic in time-series analysis, control theory, robotics, and reinforcement learning. There is an extensive literature on theoretical and algorithmic developments of system identification, with many excellent textbooks [1, 2] and surveys [3, 4, 5] available. Classical results often offer asymptotic convergence guarantees for learning system models from observed data [1, 5]. There has been an increasing interest in finite sample complexity and non-asymptotic analysis, since good error bounds are essential for designing high-performance robust control systems as well as for establishing end-to-end performance guarantees [6, 7, 8].

In this paper, we consider the problem of identifying a discrete-time linear time-invariant (LTI) system

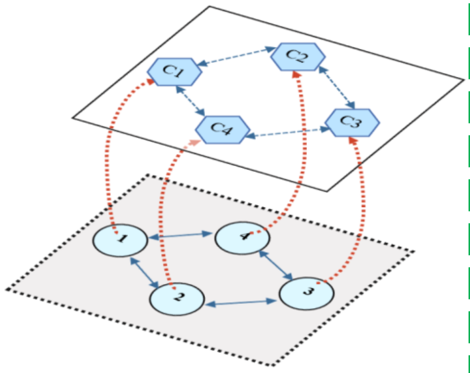
$$x_{t+1} = Ax_t + Bu_t + B_w w_t$$
$$y_t = Cx_t + Du_t + D_v v_t, \quad (1)$$

Scalable Optimization and Control (SOC) lab

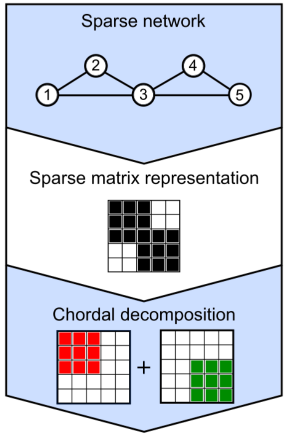
Data-driven and learning-based control



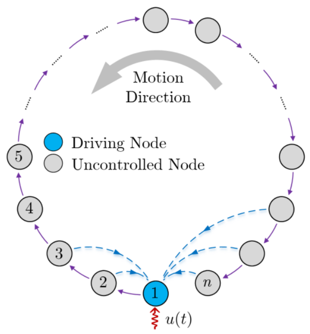
Scalable distributed control



Sparse conic optimization

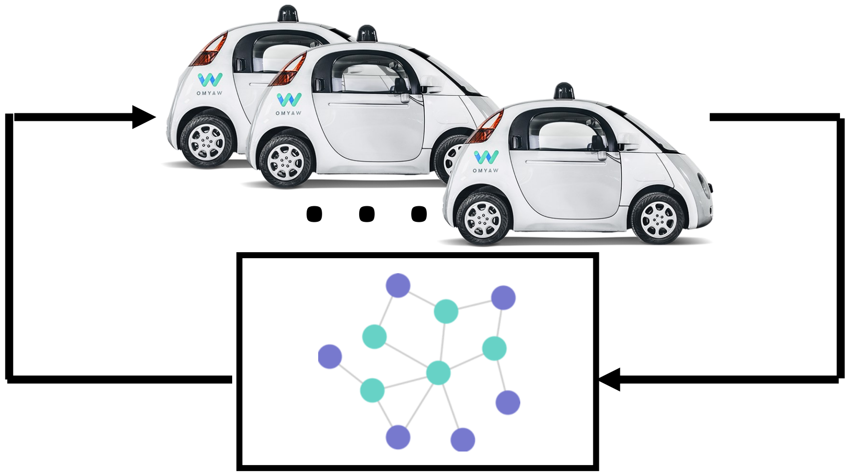


Connected and autonomous vehicles (CAVs)



Check out our webpage: https://zhengy09.github.io/soclab.html

General procedure



Distributed controller

Control problems (analysis/synthesis)

↓

Convex reformulation as LMI or SDP

↓

Call a numerical solver

Challenge 1: Model-free or Model-based design

Challenge 2: How to recover convexity

Challenge 3: How to solve large-scale semidefinite programs (SDPs)

Centralized Controller

Optimization perspective: static

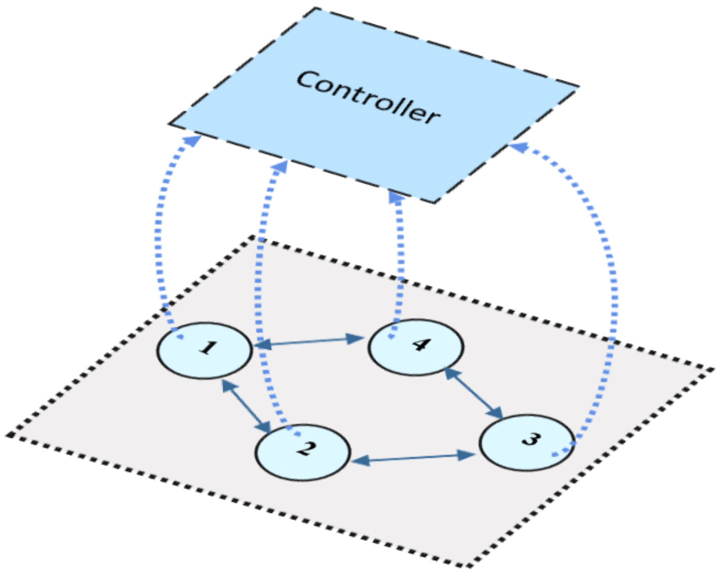
$$\begin{aligned} \text{minimize} \quad & J_1(K) \\ \text{subject to} \quad & K \in \mathcal{C}_{\text{stab},1}. \end{aligned}$$

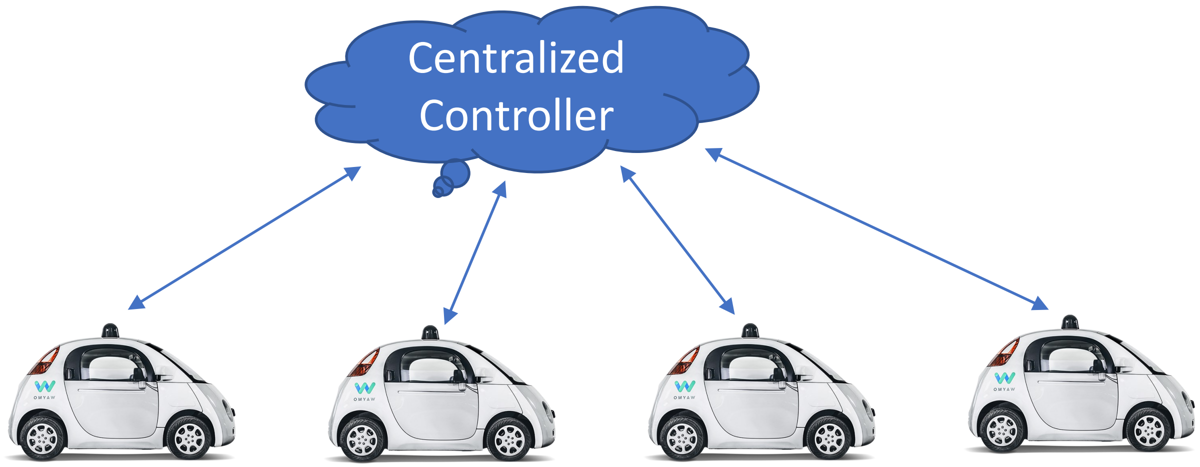
- $\mathcal{C}_{\text{stab},1}$ is the set of stabilizing **static** controllers.

Optimization perspective: dynamic

$$\begin{aligned} \text{minimize} \quad & J_2(\mathbf{K}) \\ \text{subject to} \quad & \mathbf{K} \in \mathcal{C}_{\text{stab},2}. \end{aligned}$$

- $\mathcal{C}_{\text{stab},2}$ is the set of stabilizing **dynamic** controllers

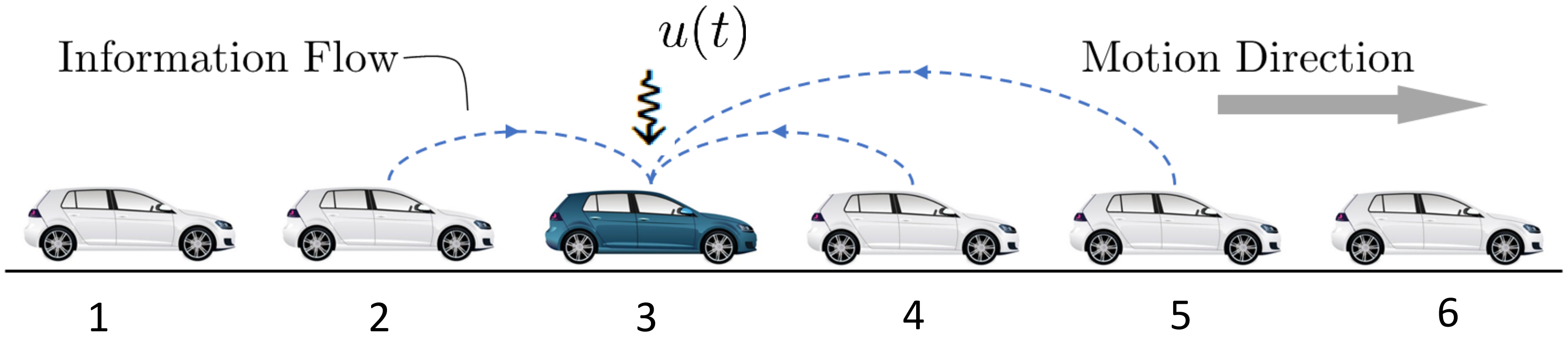




Design of Distributed Controller

❑ Why distributed?

- No need of a centralized coordinator
- Allow for local communication



$$u(t) = -(k_1 x_1 + k_2 x_2 + k_3 x_3 + k_4 x_4 + k_5 x_5 + k_6 x_6)$$

$$k_1 = k_6 = 0$$

❑ Sparsity constraint

$$K \in \text{Sparse}\left(S = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}\right) = \left\{\begin{bmatrix} 0 & * & * & * & * & 0 \end{bmatrix}\right\}$$

The binary matrix S encodes local communication

Design of Distributed Controller

Optimization perspective: static

$$
\begin{aligned}
\text{minimize} \quad & J_1(K) \\
\text{subject to} \quad & K \in \mathcal{C}_{\text{stab},1}, \\
& K \in \text{Sparse}(S)
\end{aligned}
$$

- $\mathcal{C}_{\text{stab},1}$ is the set of static stabilizing controllers

Optimization perspective: dynamic

$$
\begin{aligned}
\text{minimize} \quad & J_2(\mathbf{K}) \\
\text{subject to} \quad & \mathbf{K} \in \mathcal{C}_{\text{stab},2}, \\
& \mathbf{K} \in \text{Sparse}(S)
\end{aligned}
$$

- $\mathcal{C}_{\text{stab},2}$ is the set of dynamic stabilizing controllers

✓ **Geometrical properties**: Han & Lavei, ACC 2019; Bu et al, 2019;

✓ **Convex restriction**: Geromel et al., 1994; Conte et al.,2012; Rubio et al., 2013;

✓ **Non-convex optimization**: Lin, Fardad, Jovanovic, TAC 2011; Dörfler, et al, IEEE TPS 2014

✓ **Special cases**: Polyak, Khlebnikov, & Shcherbakov, ECC 2013;

✓ **Exact solutions for special classes of systems: Quadratic Invariance** (Rotkowitz & Lall, TAC 2005); Partially ordered sets (Shah & Parrilo, TAC 2013);

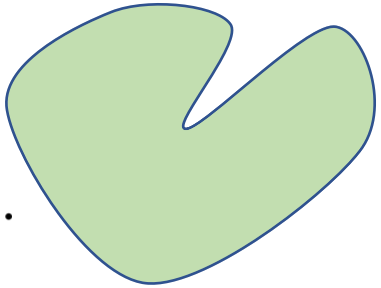
✓ **Non-smooth optimization:** Apkarian, and Dominikus IEEE TAC 2016.

✓ **Alternative formulation: system-level synthesis** (wang, Matni, Doyle, TAC 2019)

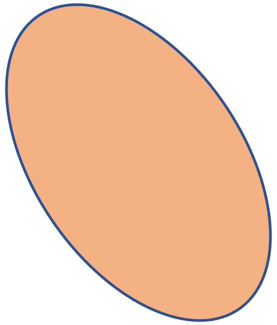
Change of Variables

- **Do not optimize the controller K directly:** Convex reformation via a change of variables (convex SDP); Boyd et al. 1994

$$\begin{aligned} \text{minimize} \quad & J_1(K) \\ \text{subject to} \quad & K \in \mathcal{C}_{\mathrm{Stab},1}. \end{aligned}$$



$$K = YX^{-1}$$



$$\begin{aligned} \text{minimize} \quad & g_1(X,Y) \\ \text{subject to} \quad & (X,Y) \in \hat{\mathcal{C}}_{\mathrm{Stab},1}. \end{aligned}$$

✓ $V(x) = x^T X^{-1} x$ defines a **Lyapunov function** for the closed-loop system.

Lyapunov inequality

$A - BK$ is stable $\iff \begin{cases} X \succ 0 \\ (A-BK)X + X(A-BK)^T \prec 0 \end{cases}$

$\iff \begin{cases} X \succ 0 \\ AX - BY + (AX-BY)^T \prec 0 \end{cases}$

Change of Variables

$$KX = Y$$

Challenges and heuristics

❑ Optimization perspective: static state feedback

$$\begin{aligned} \text{minimize} \quad & J_1(K) \\ \text{subject to} \quad & K \in \mathcal{C}_{\text{stab},1}, \\ & K \in \text{Sparse}(S) \end{aligned}$$

❑ **Method** via a change of variables

$$\begin{aligned} \text{minimize} \quad & g_1(X, Y) \\ \text{subject to} \quad & (X, Y) \in \hat{\mathcal{C}}_{\text{Stab},1} \\ & YX^{-1} \in \text{Sparse}(S). \end{aligned}$$

Non-convex constraint

One approximation strategy (Geromel et al., 1994; Conte et al.,2012; Rubio et al., 2013; Han et al., 2017)

$$X \text{ is diagonal}, \; Y \in \text{Sparse}(S)$$
$$\Rightarrow$$
$$YX^{-1} \in \text{Sparse}(S)$$

- Requires a diagonal Lyapunov function $V(x) = x^T X^{-1} x$
- May be too restrictive.

Sparsity Invariance

Sparsity invariance (SI)

$$X \in \text{Sparse}(R),\ Y \in \text{Sparse}(T)$$
$$\Rightarrow$$
$$K = YX^{-1} \in \text{Sparse}(S)$$

A new and unified framework based on **Sparsity Invariance (SI)** for **convex design** of the **largest known class** of distributed control problems

Goes beyond the well-known notion of QI

Static case

- Strictly better than the widely used **diagonal approximation strategy** (Geromel et al., 1994; Conte et al.,2012; Rubio et al., 2013; Han et al., 2017)

dynamical case

- Guaranteed to be optimal when QI holds
- Best known performance for non-QI cases (Rotkowitz & Martins, 2012)

Some other results

Sparse Invariance

(Best student paper award finalist at ECC19)

Sparsity Invariance for Co Distributed Contr

Luca Furieri, *Student Member, IEEE*, Yang Zh
Antonis Papachristodoulou, *Fellow, IEEE*, and

Abstract—**We address the problem of designing optimal linear time-invariant (LTI) sparse controllers for LTI systems, which corresponds to minimizing a norm of the closed-loop system subjected to sparsity constraints on the controller structure. This problem is NP-hard in general and motivates the development of tractable approximations. We characterize a class of convex restrictions based on a new notion of sparsity invariance (SI). The underlying idea of SI is to design sparsity patterns for transfer matrices $\mathrm{Y}(s)$ and $\mathrm{X}(s)$ such that any corresponding controller $\mathrm{K}(s) = \mathrm{Y}(s)\mathrm{X}(s)^{-1}$ exhibits the desired sparsity pattern. For sparsity constraints, the approach of SI goes beyond the notion of quadratic invariance (QI): 1) the SI approach always yields a convex restriction and 2) the solution via the SI approach is guaranteed to be globally optimal when QI holds and performs at least, considering the nearest QI subset. Moreover, the notion of SI naturally applies to designing structured static controllers, while QI is not utilizable. Numerical examples show that even for non-QI cases, SI can recover solutions that are: 1) globally optimal and 2) strictly more performing than previous methods.**

Index Terms—**Decentralized control, linear systems,**

to privacy cor
implementing
The celebra
tion can enorr
inputs. Indeed
even be linear
without full o
the lack of ful
core challenge
cases of optim
which efficien
Optimally
with distribute
linear controll
a norm of the
problem, the
to be sufficier
formulation. A
codesign was

Equivalence of three controller parameterizations

On the Equivalence of Youla, Sys Input–Output Parameteri

Yang Zheng, Luca Furieri, Antonis Papachristodoulou,

Abstract—**A convex parameterization of internally stabilizing controllers is fundamental for many controller synthesis procedures. The celebrated Youla parameterization relies on a doubly coprime factorization of the system, while the recent system-level and input–output parametrizations require no doubly coprime factorization, but a set of equality constraints for achievable closed-loop responses. In this article, we present explicit affine mappings among Youla, system-level, and input–output parameterizations. Two direct implications of these affine mappings are: 1) any convex problem in the Youla, system-level, or input–output parameters can be equivalently and convexly formulated in any other one of these frameworks, including the convex system-level synthesis; 2) the condition of quadratic invariance is sufficient and necessary for the classical distributed control problem to admit an equivalent convex reformulation in terms of either Youla, system-level, or input–output parameters.**

Index Terms—**Quadratic invariance (QI), stabilizing controller, system-level synthesis (SLS), Youla parameterization.**

I. INTRODUCTION

One of the most fundamental problems in control theory is to design a feedback controller that stabilizes a dynamical system. Additionally, one can further design an optimal controller by optimizing a certain performance measure [1]. It is well known that the set of stabilizing controllers is in general nonconvex, and hence, hard to optimize directly

parameterizatio
response) direct
problem. Also,
loop system car
convex optimiz
and optimal con
that a doubly co
preliminary step
parameterizatio
[6] were introd
trollers, with no
the system *a pri*
treat certain clo
synthesis is thu
closed-loop resp
closed-loop res
closed-loop con
The Youla pa
since they chara
their explicit rel
main objective c
parameterizatio
mappings betw
input–output pa
problem in terms of the Youla, system-level, input–output parameters can be equivalently and convexly formulated into any other one of

New controller parameterizations

[math.OC] 28 May 2020

System-level, Input-output and New Parameterizations of Stabilizing Controllers, and Their Numerical Computation

Yang Zheng [a,b], Luca Furieri [c], Maryam Kamgarpour [c,d], Na Li [a,b]

[a] School of Engineering and Applied Sciences, Harvard University, Boston, MA, 02138, U.S.
[b] Harvard Center of Smart Cities and Buildings, Harvard University, Boston, MA, 02138, U.S.
[c] Automatic Control Laboratory, ETH Zurich, Switzerland.
[d] Electrical & Computer Engineering, UBC, Vancouver, Canada.

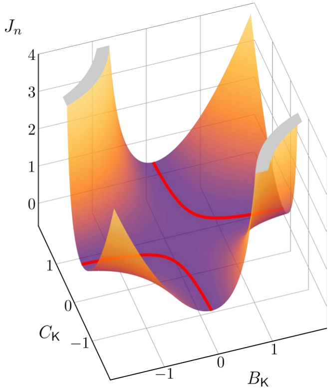
Abstract

It is known that the set of internally stabilizing controller $\mathcal{C}_{\text{stab}}$ is non-convex, but it admits convex characterizations using certain closed-loop maps: a classical result is the Youla parameterization, and two recent notions are the system-level parameterization (SLP) and the input-output parameterization (IOP). In this paper, we address the existence of new convex parameterizations and discuss potential tradeoffs of each parametrization in different scenarios. Our main contributions are: 1) We first reveal that only four groups of stable closed-loop transfer matrices are equivalent to internal stability: one of them is used in the SLP, another one is used in the IOP, and the other two are new, leading to two new convex parameterizations of $\mathcal{C}_{\text{stab}}$. 2) We then investigate the properties of these parameterizations after imposing the finite impulse response (FIR) approximation, revealing that the IOP has the best ability of approximating $\mathcal{C}_{\text{stab}}$ given FIR constraints. 3) These four parameterizations require no *a priori* doubly-coprime factorization of the plant, but impose a set of equality constraints. However, these equality constraints will never be satisfied exactly in numerical computation. We prove that the IOP is numerically robust for open-loop stable plants, in the sense that small mismatches in the equality constraints do not compromise the closed-loop stability. The SLP is known to enjoy numerical robustness in the state feedback case; here, we show that numerical robustness of the four-block SLP controller requires case-by-case analysis in the general output feedback case.

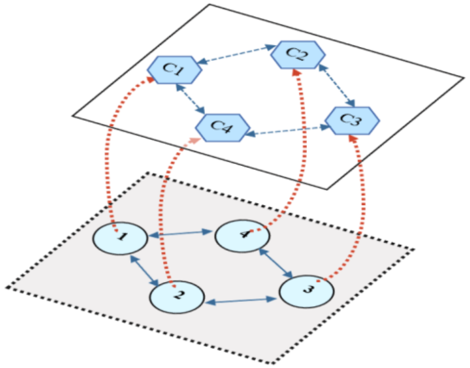
Scalable Optimization and Control (SOC) lab

SOC Lab

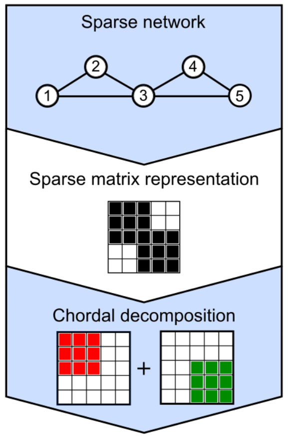
Data-driven and learning-based control



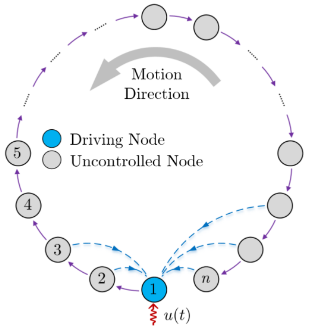
Scalable distributed control



Sparse conic optimization



Connected and autonomous vehicles (CAVs)



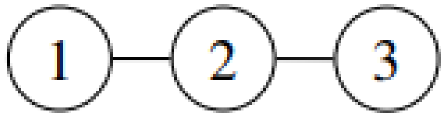
Check out our webpage: https://zhengy09.github.io/soclab.html

A simple example

$$A = \underbrace{\begin{bmatrix} 3 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 3 \end{bmatrix}}_{\succeq 0} = \underbrace{\begin{bmatrix} 3 & 1 & 0 \\ 1 & 0.5 & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{\succeq 0} + \underbrace{\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.5 & 1 \\ 0 & 1 & 3 \end{bmatrix}}_{\succeq 0}$$

Sparse **positive semidefinite (PSD) cone decomposition** (Agler et al. 1984)

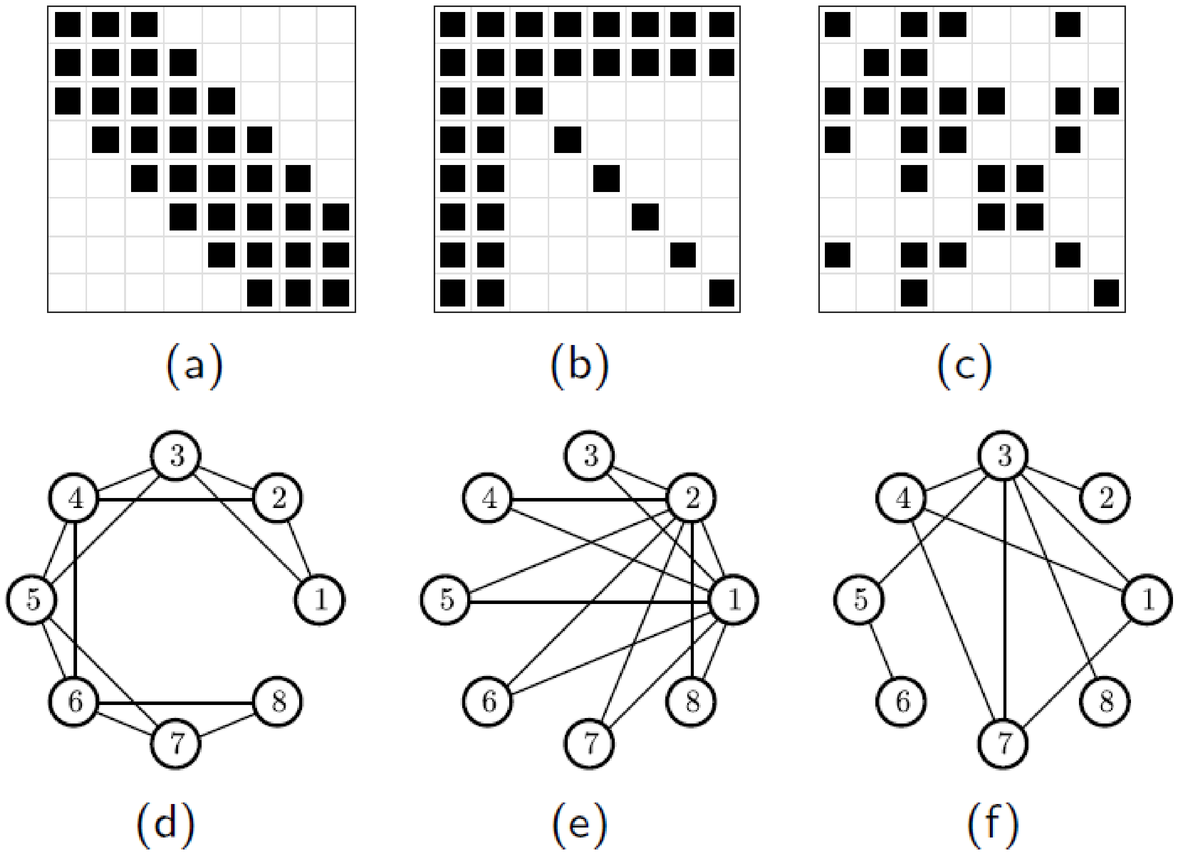
$$\underbrace{\begin{bmatrix} * & * & 0 \\ * & * & * \\ 0 & * & * \end{bmatrix}}_{\succeq 0} = \underbrace{\begin{bmatrix} * & * & 0 \\ * & * & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{\succeq 0} + \underbrace{\begin{bmatrix} 0 & 0 & 0 \\ 0 & * & * \\ 0 & * & * \end{bmatrix}}_{\succeq 0}$$



Benefits: Reduce computational complexity, and thus improve efficiency! ($3 \times 3 \rightarrow 2 \times 2$)

Sparse Matrix Decomposition

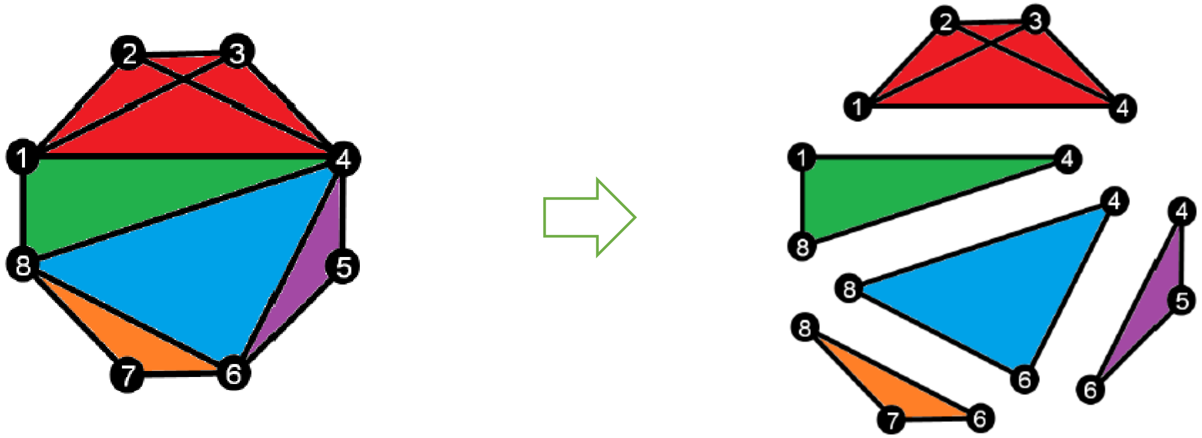
- Many other sparsity patterns admit similar matrix decompositions

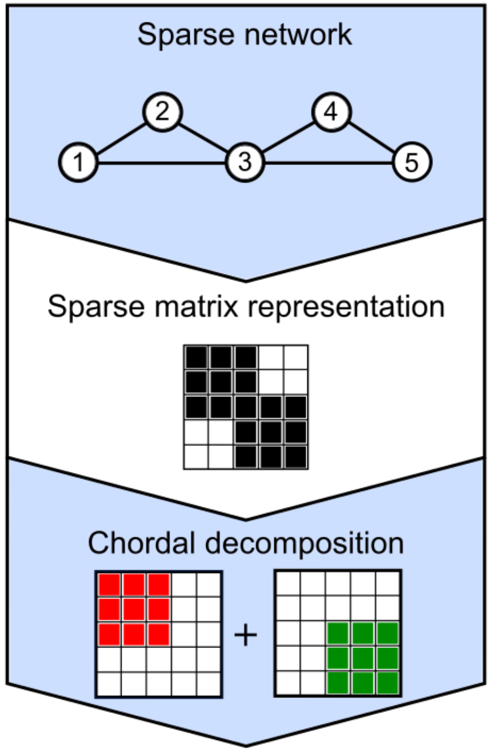


- They can be commonly characterized by **chordal graphs** (any cycle of length > 3 has a chord).

Matrix Decomposition and Chordal Graphs

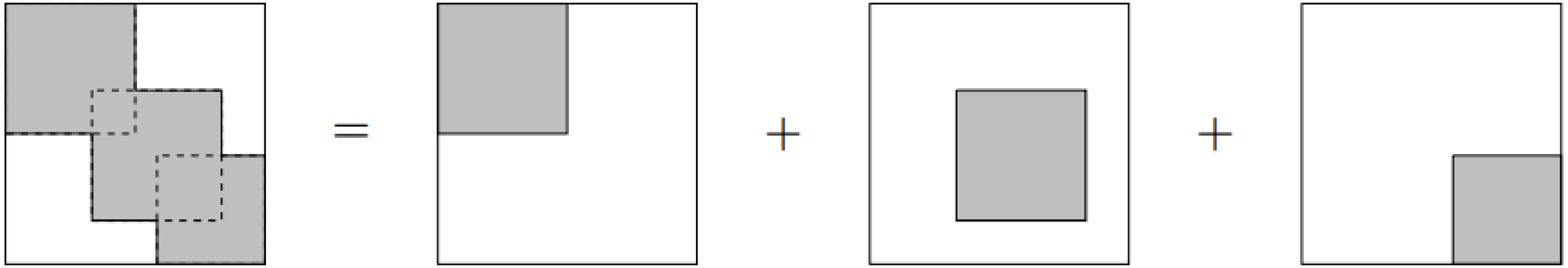
- A chordal graph can be decomposed into its maximal cliques $\mathcal{C} = \{\mathcal{C}_1, \mathcal{C}_2, \ldots, \mathcal{C}_p\}$.





Clique (Chordal) Decomposition

$$Z \in \mathbb{S}^n_+(\mathcal{E}, 0) \Leftrightarrow Z = \sum_{k=1}^{p} E_k^T Z_k E_k, \; Z_k \in \mathbb{S}_+^{|\mathcal{C}_k|}$$

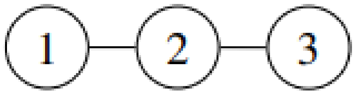


(Agler, et al., 1988; Griewank and Toint, 1984)

A Sparse SDP example

$$C = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, A_1 = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 2 \end{bmatrix}, A_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 2 \end{bmatrix}$$

$$\begin{bmatrix} * & * & 0 \\ * & * & * \\ 0 & * & * \end{bmatrix}$$



Define an SDP

$$\begin{aligned} \max_{y_1, y_2, Z} \quad & b_1 y_1 + b_2 y_2 \\ \text{subject to} \quad & y_1 A_1 + y_2 A_2 + Z = C \\ & Z \in \mathbb{S}^3_+. \end{aligned}$$

Patterns of feasible solutions

$$Z \in \begin{bmatrix} * & * & 0 \\ * & * & * \\ 0 & * & * \end{bmatrix}$$

$$\underbrace{\begin{bmatrix} * & * & 0 \\ * & * & * \\ 0 & * & * \end{bmatrix}}_{\succeq 0} = \underbrace{\begin{bmatrix} * & * & 0 \\ * & * & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{\succeq 0} + \underbrace{\begin{bmatrix} 0 & 0 & 0 \\ 0 & * & * \\ 0 & * & * \end{bmatrix}}_{\succeq 0}$$

Cone replacement

$$Z \in \mathbb{S}^3_+(\mathcal{E}, 0)$$

✓ Applying the clique decomposition (Fukuda et al., 2001; Nakata et al., 2003; Andersen et al., 2010; Madani et al., 2015; Sun, Andersen, and Vandenberghe, 2014; **Zheng** et al., 2017 & 2019)

First-order algorithms for SDPs

Work	Complexity per Iteration	Generality	Infeasibility Detection	Open-source Solver
O'Donoghue et al. 2016	$\mathcal{O}(n^3)$ + QP	**General SDP**	**Yes**	**SCS**
Dall'Anese, Zhu, Giannakis, 2013	$\mathcal{O}(\lvert C_k \rvert^3)$ + QP	Special OPF problems with Sep. constraints	No	No
Sun &Vandenberghe, 2015	$\mathcal{O}(\lvert C_k \rvert^3)$	Special SDP with no equality constraints	No	No
Sun et al, 2014	$\mathcal{O}(\lvert C_k \rvert^3)$ + IPM	General SDP	No	No
Madani et al, 2015	$\mathcal{O}(\lvert C_k \rvert^3)$ + QP	General sparse SDP with ineq. Constraints	No	No
Kalbat & Lavaei, 2015	$\mathcal{O}(\lvert C_k \rvert^3)$ + QP	Special sparse SDP with Sep. constraints	No	No
Today's talk (Zheng et al., 2017 & 2019)	$\mathcal{O}(\lvert C_k \rvert^3)$ **+ QP**	**General SDP**	**Yes**	**CDCS**

Open-source Solver: CDCS

Case 1: Test on sparse benchmark problems (from Andersen, Dahl, Vandenberghe, 2010)

Problem instance: rs1907

- PSD size 5357× 5357
- > **10 million** decision variables

✓ SeDuMi ran out of memory

✓ The first-order solver SCS took over **13 hours** to return a solution

✓ **CDCS** took **6 minutes** to get a solution; **100 × speedup!**

Exploiting sparsity achieves massive scalability in both **time** and **memory**

	rs228			rs365		
	Time (s)	# Iter.	Objective	Time (s)	# Iter.	Objective
SeDuMi (high)	1655	21	64.71	***	***	***
SeDuMi (low)	809	10	64.80	***	***	***
SCS (direct)	2338	2000	62.06	34,497	2000	44.02
CDCS-primal	94	400	64.65	321	401	63.37
CDCS-dual	84	341	64.76	240	265	63.69
CDCS-hsde	79	361	64.87	332	442	63.64

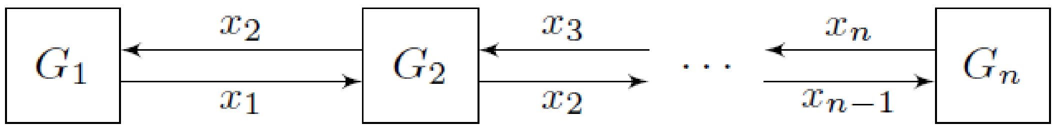
	rs1555			rs1907		
	Time (s)	# Iter.	Objective	Time (s)	# Iter.	Objective
SeDuMi (high)	***	***	***	***	***	***
SeDuMi (low)	***	***	***	***	***	***
SCS (direct)	139,314	2000	34.20	50,047	2000	45.89
CDCS-primal	1721	2000	61.22	330	349	62.87
CDCS-dual	317	317	69.54	271	252	63.30
CDCS-hsde	1413	2000	61.36	393	414	63.14

Entries marked *** indicate that the problem could not be solved due to memory limitations

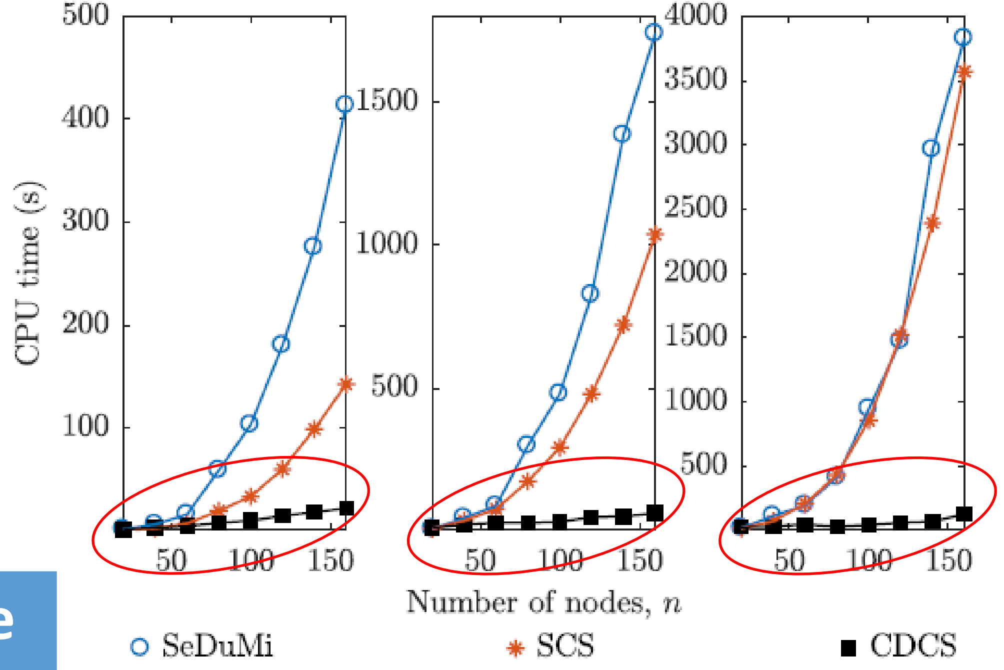
Open-source Solver: CDCS

Case 2: Test on stability/H2/Hinf analysis of linear network systems

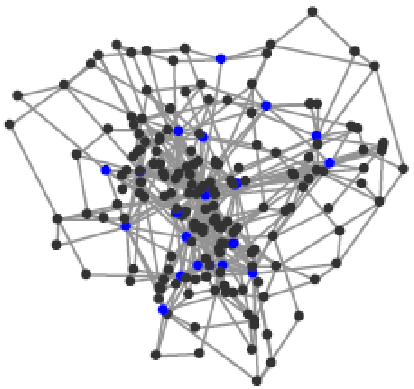
❑ Example 1: a chain of interconnected subsystems



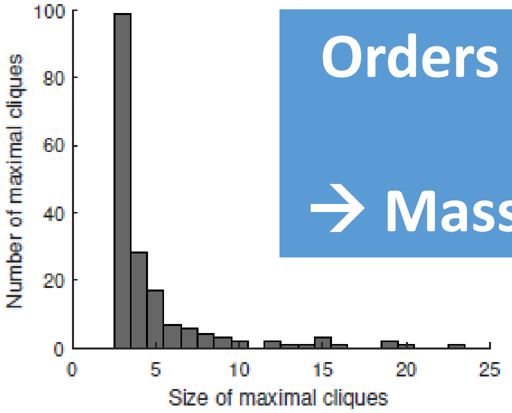
- Randomly generated stable subsystems (state dimension: 5-10).
- The graph is a line where the maximal cliques are $\mathcal{C}_\mathrm{i} = \{i, i+1\}$
- Apply a block-diagonal Lyapunov function → preserve sparsity



❑ Example 2: a sparse network system



(a) Scale free graph



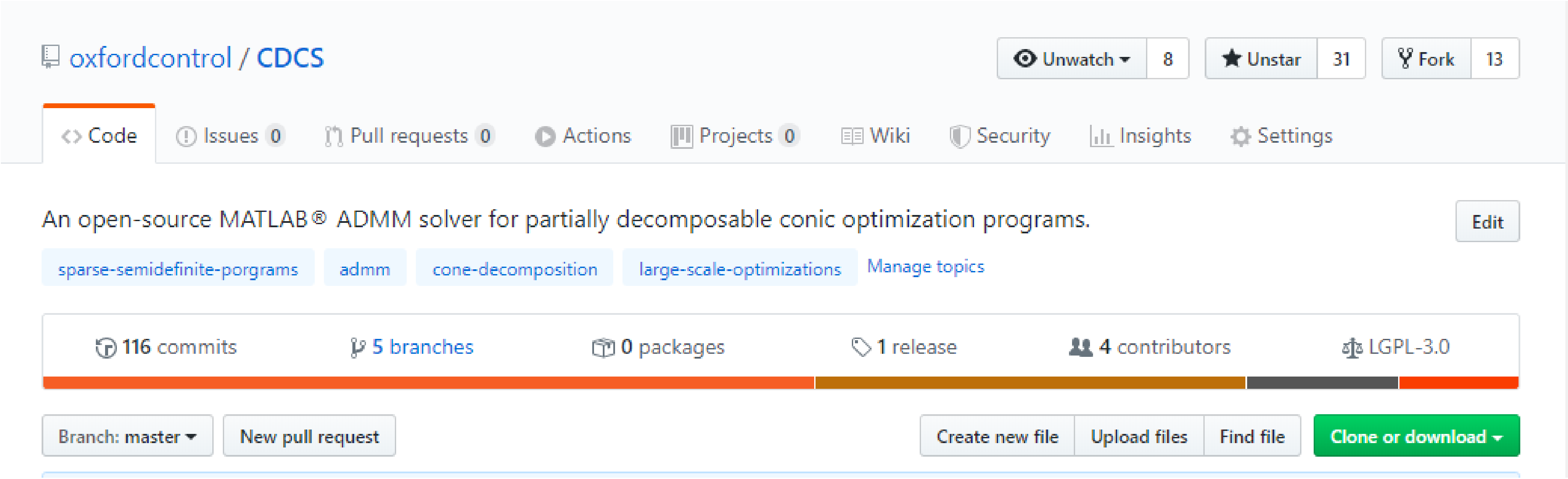
(b) Clique size distribution

Orders of magnitude faster
→ Massive Scalability

	Time (s)		
	sedumi	SCS	CDCS
Stability	115.0	108.9	40.6
$\mathcal{H}_2$	805.0	556.1	147.4
$\mathcal{H}_\infty$	3 374.8	2 130.2	172.0

Open-source Solver: CDCS

Large-scale practical problems

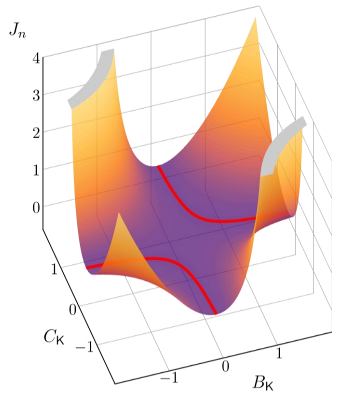


- ❑ **Signal recovery problem**
  - Fosson, S. M., & Abuabiah, M. (2019). Recovery of binary sparse signals from compressed linear measurements via polynomial optimization. *IEEE Signal Processing Letters*.
- ❑ **Optimal power flow problem**
  - Eltved, A., Dahl, J., & Andersen, M. S. (2018). On the robustness and scalability of semidefinite relaxation for optimal power flow problems. *Optimization and Engineering*, 1-18.
- ❑ **Nonlinear systems analysis**
  - Driggs & Fawzi (2019). “AnySOS: An anytime algorithm for semidefinite programming" IEEE CDC, 1-6.

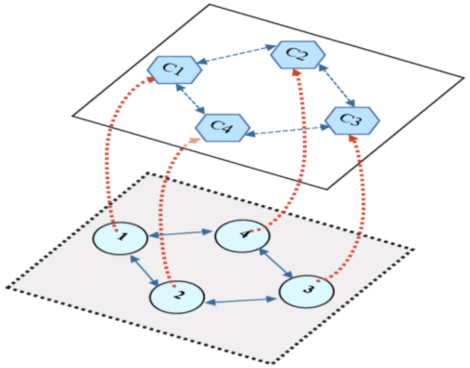
Scalable Optimization and Control (SOC) lab

SOC Lab

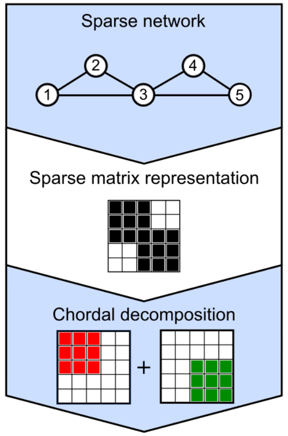
Data-driven and learning-based control



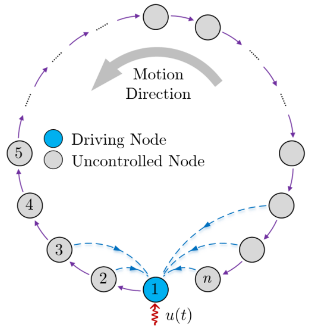
Scalable distributed control



Sparse conic optimization



Connected and autonomous vehicles (CAVs)

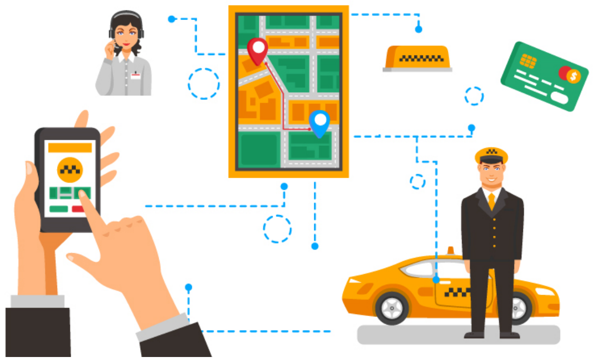


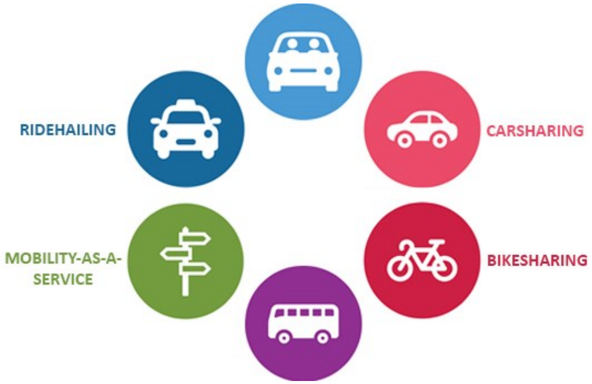
Check out our webpage: https://zhengy09.github.io/soclab.html

Autonomous Vehicles

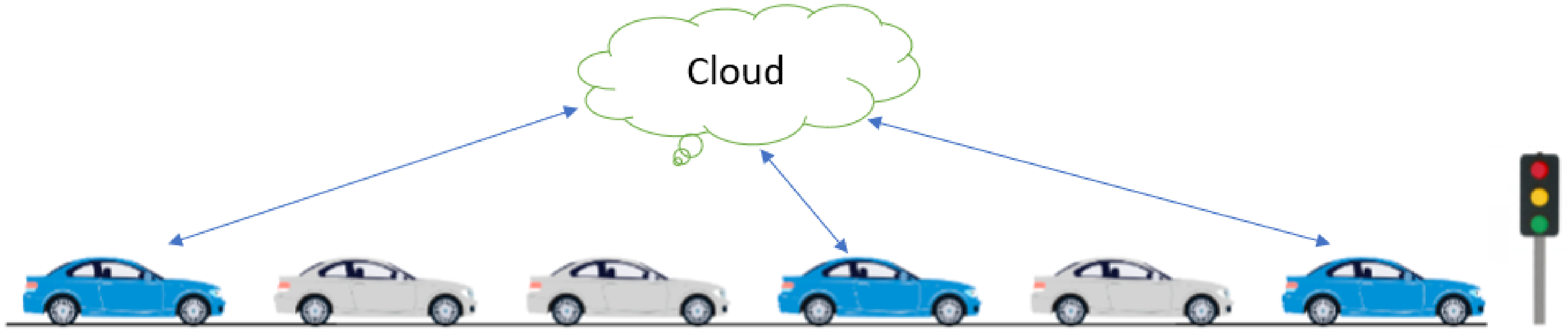
U.S. Census Bureau, 2017.

- **Reduce traffic accidents**
  - 37,000 fatalities
  - 41% deaths of young adults (ages 15-24)
  - **94%** of serious crashes caused by human error
- **Ease traffic congestion**
  - **6.9 billion hours** wasted annually
  - Cost of traffic congestion is **$1740** per person annually in US/Europe.
- **Improve energy efficiency**
  - 28% of greenhouse gas emission is from transportation
- **New mobility patterns**: on-demand mobility, mobility as service etc.





Mix-Autonomy Mobility



Mixed-autonomy mobility: a traffic condition where both autonomous vehicles and human-driven vehicles co-exist.

- **Q1:** How will **a small scale of autonomous vehicles** change traffic dynamics?
- **Q2:** How to integrate **a small scale of autonomous vehicles** to improve traffic performance?

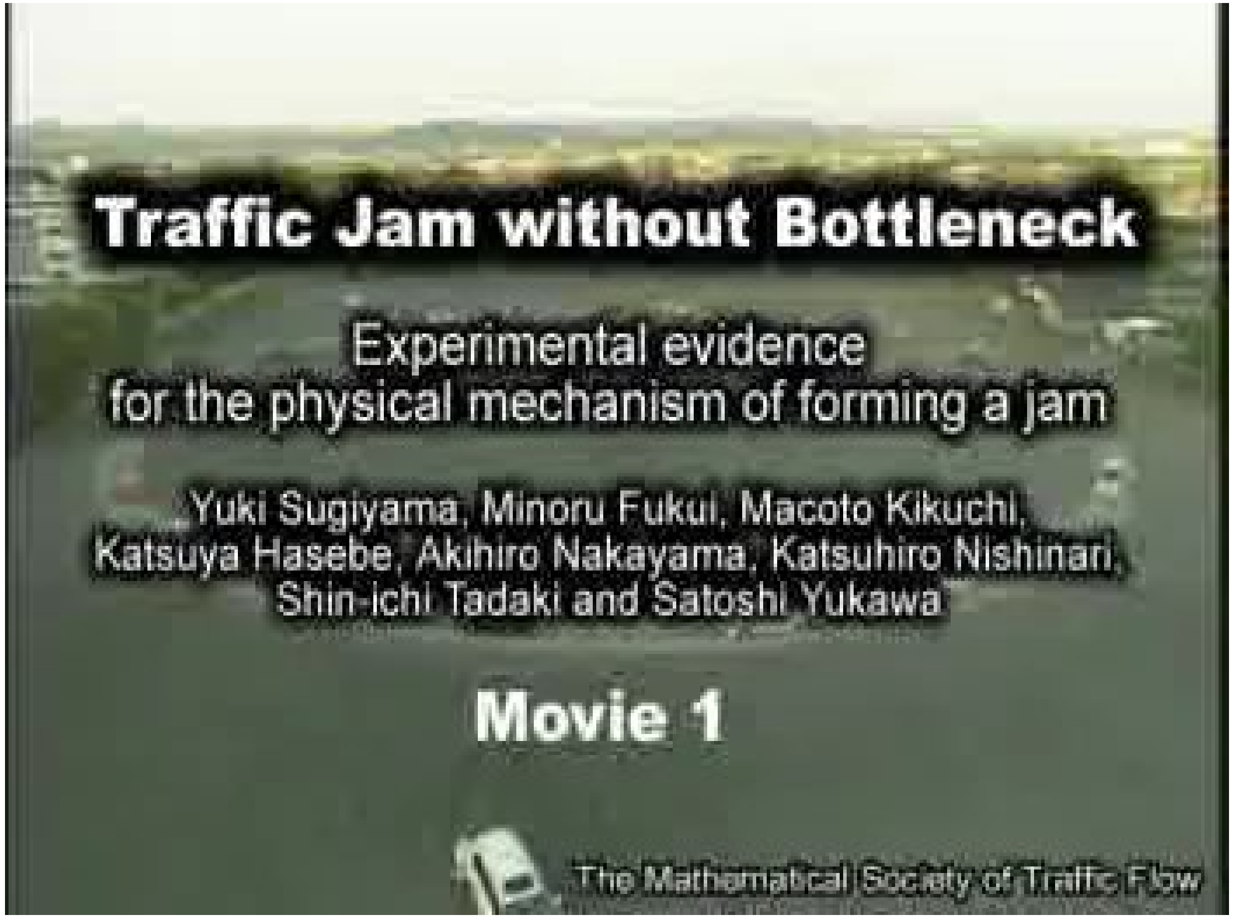
Theoretical evidence of the high potential of autonomous vehicles

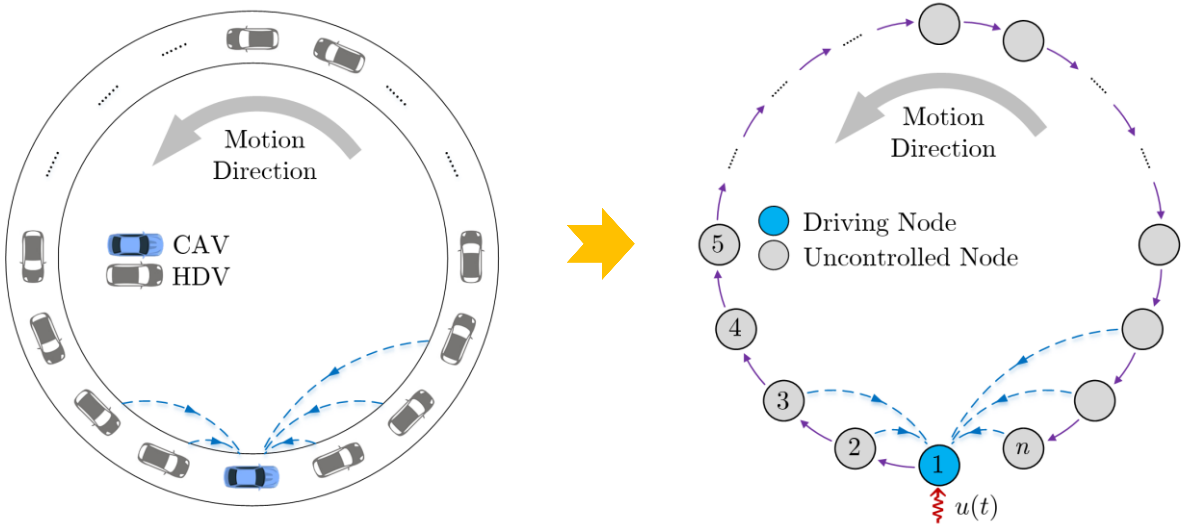
Practical design via distributed control and scalable optimization

Mixed urban mobility

Design an optimal distributed controller for autonomous vehicles to actively smooth traffic flow

Real-world Experiments in 2008 (Sugiyama, et al. 2008)





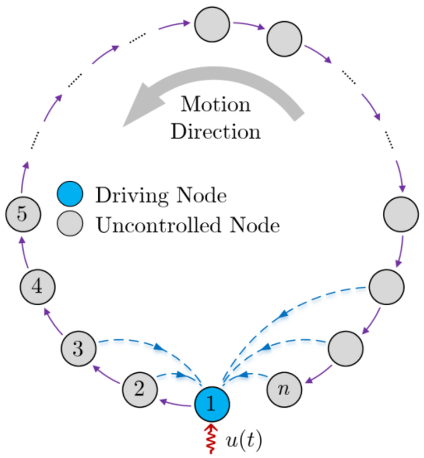
- The linearized system is **stabilizable** after introducing a single autonomous vehicle;
- Design a distributed controller;

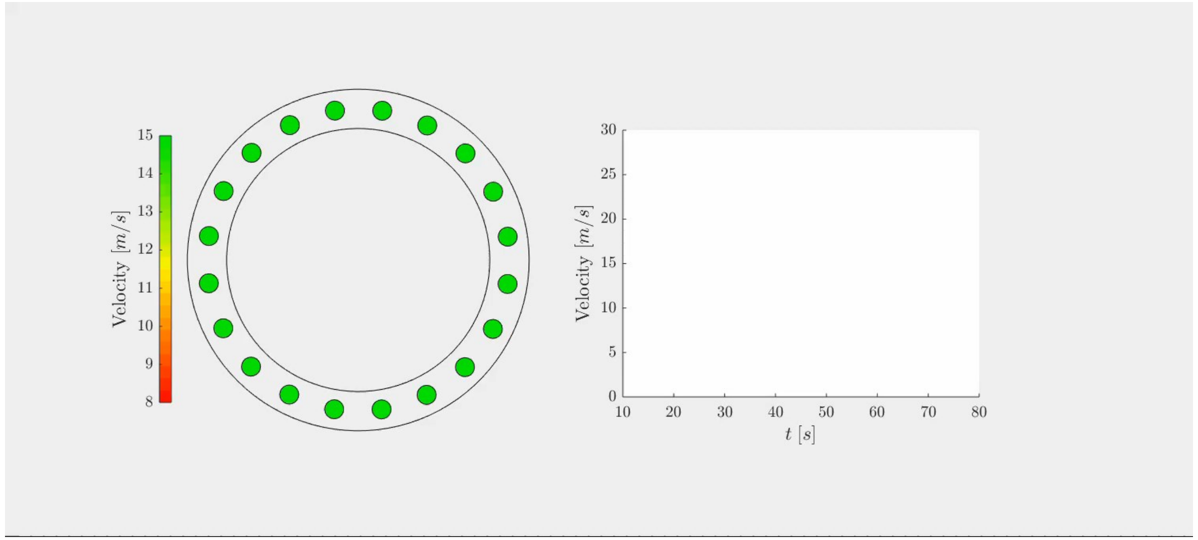
$$\begin{aligned} \text{minimize} \quad & J(K) \\ \text{subject to} \quad & K \in \mathcal{C} \cap \text{Sparse}(S). \end{aligned}$$

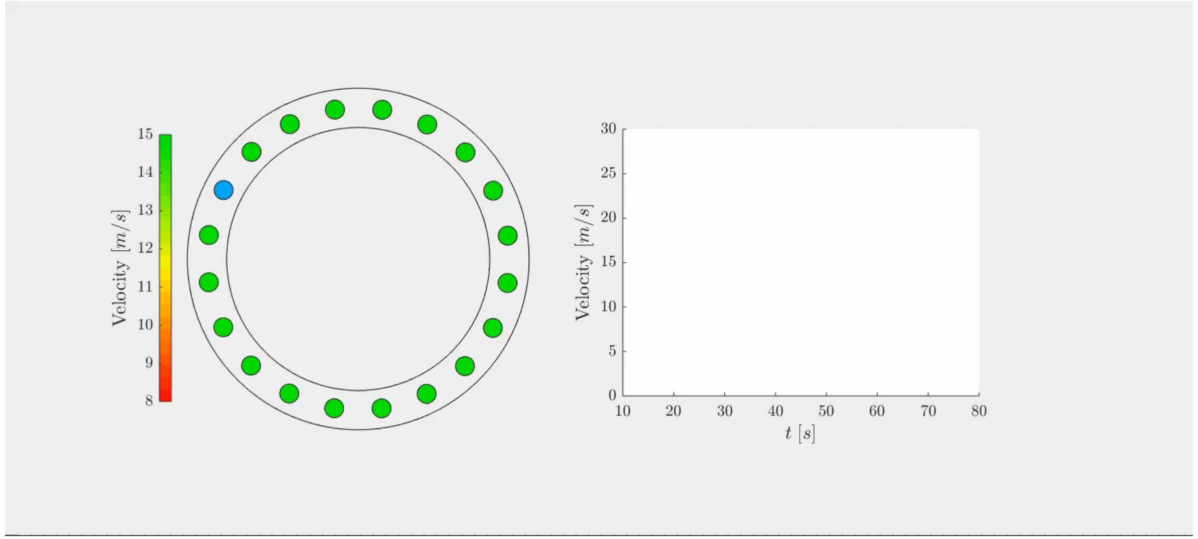
zheng et al., IEEE Journal Internet of Things, 2019, accepted

Sparsity invariance: Mixed urban mobility

Design an optimal distributed controller for autonomous vehicles to **actively smooth traffic flow**







OVM: Optimal Velocity Model

$$F_i = \alpha(V(s_i(t)) - v_i(t)) + \beta \dot{s}_i(t)$$

$$V(s) = \begin{cases} 0, & s \le s_{\mathrm{st}}, \\ f_v(s), & s_{\mathrm{st}} < s < s_{\mathrm{go}}, \\ v_{\max}, & s \ge s_{\mathrm{go}}, \end{cases}$$

$$f_v(s) = \frac{v_{\max}}{2}\left(1 - \cos(\pi \frac{s - s_{\mathrm{st}}}{s_{\mathrm{go}} - s_{\mathrm{st}}})\right).$$

zheng et al., IEEE Journal Internet of Things, 2019, accepted

Some other results

Smoothing Traffic Flow

Smoothing Traffic Fl
of Autonomous

Yang Zheng, *Member, IEEE*, Jiawei Wang, *St*

Abstract—The emergence of autonomous vehicles (AVs) is expected to revolutionize road transportation in the near future. Although large-scale numerical simulations and small-scale experiments have shown promising results, a comprehensive theoretical understanding to smooth traffic flow via AVs is lacking. In this article, from a control-theoretic perspective, we establish analytical results on the controllability, stabilizability, and reachability of a mixed traffic system consisting of human-driven vehicles and AVs in a ring road. We show that the mixed traffic system is not completely controllable, but is stabilizable, indicating that AVs can not only suppress unstable traffic waves but also guide the traffic flow to a higher speed. Accordingly, we establish the maximum traffic speed achievable via controlling AVs. Numerical results show that the traffic speed can be increased by over 6% when there are only 5% AVs. We also design an optimal control strategy for AVs to actively dampen undesirable perturbations. These theoretical findings validate the high potential of AVs to smooth traffic flow.

Index Terms—Autonomous vehicle (AV), controllability, mixed traffic flow, stabilizability.

macı desc many to in Curr locat speed signs cally Thes impo As plays past panic of au been auto to re adva

Leading Cruise Control

This article has been accepted for inclusion in a future issue of this journal. Content is final a

Leading Cruise Control in Mi
System Modeling, Con
and String Stabi

Jiawei Wang, *Graduate Student Member, IEEE*, Yan
Chaoyi Chen, *Graduate Student Member, IEEE*, Qi

Abstract—Connected and autonomous vehicles (CAVs) have great potential to improve road transportation systems. Most existing strategies for CAVs' longitudinal control focus on downstream traffic conditions, but neglect the impact of CAVs' behaviors on upstream traffic flow. In this paper, we introduce a notion of Leading Cruise Control (LCC), in which the CAV maintains car-following operations adapting to the states of its preceding vehicles, and also aims to lead the motion of its following vehicles. Specifically, by controlling the CAV, LCC aims to attenuate downstream traffic perturbations and smooth upstream traffic flow actively. We first present the dynamical modeling of LCC, with a focus on three fundamental scenarios: car-following, free-driving, and Connected Cruise Control. Then, the analysis of controllability, observability, and head-to-tail string stability reveals the feasibility and potential of LCC in improving mixed traffic flow performance. Extensive numerical studies validate that the capability of CAVs in dissipating traffic perturbations is further strengthened when incorporating the information of the vehicles behind into the CAVs' control.

High-accuracy rithms enable car-following potential of A emergence of By exploiting (V2V) or vel other vehicles thereby allowi traditional AC To coordina Control (CAC In CACC, a a platoon, fo demonstrates communicatio

Controllability Analysis

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Controllability Analysis and Optimal Control of Mixed Traffic Flow With Human-Driven and Autonomous Vehicles

Jiawei Wang, *Graduate Student Member, IEEE*, Yang Zheng, *Member, IEEE*,
Qing Xu, Jianqiang Wang, and Keqiang Li

Abstract—Connected and automated vehicles (CAVs) have a great potential to improve traffic efficiency in mixed traffic systems, which has been demonstrated by multiple numerical simulations and field experiments. However, some fundamental properties of mixed traffic flow, including controllability and stabilizability, have not been well understood. This paper analyzes the controllability of mixed traffic systems and designs a system-level optimal control strategy. Using the Popov-Belevitch-Hautus (PBH) criterion, we prove for the first time that a ring-road mixed traffic system with one CAV and multiple heterogeneous human-driven vehicles is not completely controllable, but is stabilizable under a very mild condition. Then, we formulate the design of a system-level control strategy for the CAV as a structured optimal control problem, where the CAV's communication ability is explicitly considered. Finally, we derive an upper bound for reachable traffic velocity via controlling the CAV. Extensive numerical experiments verify the effectiveness of our analytical results and the proposed control strategy. Our results validate the possibility of utilizing CAVs as mobile actuators to smooth traffic flow actively.

for traffic control rely on certain actuators at fixed locations, such as traffic signals and signs on roadside infrastructure [2]. Two typical systems are variable speed limits and variable speed advisory, which already have certain industrial applications [3]. Due to their dependence on fixed infrastructure and drivers' compliance, however, the flexibility and effectiveness of these systems might be compromised [4].

As one key ingredient of traffic systems, the motion of vehicles plays an important role in traffic efficiency. Recent advancements on control and communication technologies have led to the emergence of connected and automated vehicles (CAVs), which are expected to revolutionize road transportation systems significantly. Compared to human-driven vehicles (HDVs), the cooperative formation of multiple CAVs, *e.g.*, adaptive cruise control (ACC) and cooperative adaptive cruise control (CACC) [5], has shown very promising effects on achieving higher traffic efficiency [6], better driving

Conclusion

SOC lab at UC San Diego. Join us!

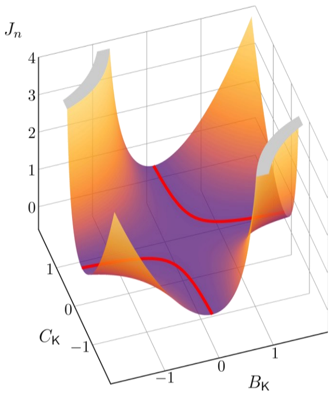
SOC Lab

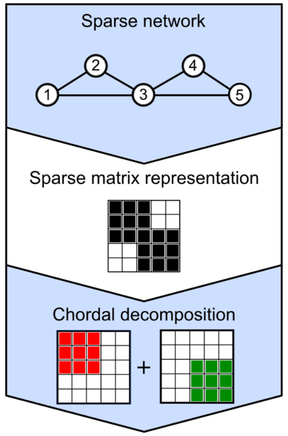
Data-driven and learning-based control

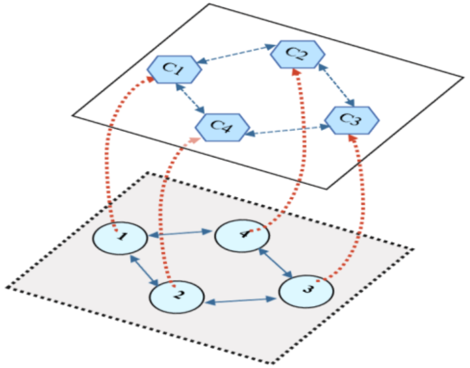
Sparse conic optimization

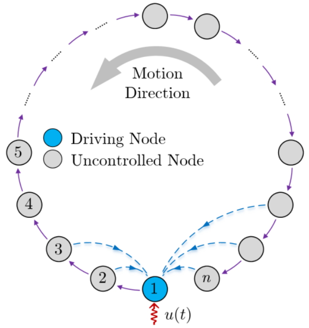
Scalable distributed control

Connected and autonomous vehicles (CAVs)









Check out our webpage: https://zhengy09.github.io/soclab.html

Thank you for your attention!

Q & A

More details. Check out our webpage: https://zhengy09.github.io/soclab.html

Extra slides

Proof idea: Lifting via Change of Variables

- **Change of variables in state-space domain: Lyapunov theory**

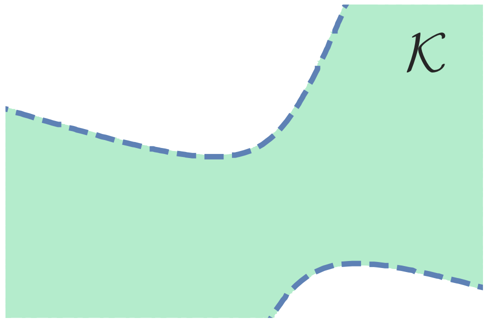
  - Connectivity of the static stabilizing state feedback gains

$$\{K \in \mathbb{R}^{m\times n} \mid A - BK \text{ is stable}\}$$

$$\iff \{K \in \mathbb{R}^{m\times n} \mid \exists P \succ 0, P(A-BK)^{\mathsf{T}} + (A-BK)P \prec 0\}$$

$$\iff \{K \in \mathbb{R}^{m\times n} \mid \exists P \succ 0, PA^{\mathsf{T}} - L^{\mathsf{T}}B^{\mathsf{T}} + AP - BL \prec 0, L = KP\}$$

$$\iff \{K = LP^{-1} \in \mathbb{R}^{m\times n} \mid \exists P \succ 0, PA^{\mathsf{T}} - L^{\mathsf{T}}B^{\mathsf{T}} + AP - BL \prec 0\}.$$



Open, connected, possibly nonconvex

  - How about the set of stabilizing dynamical controllers

$$\begin{bmatrix} A & BC_{\mathsf{K}} \\ B_{\mathsf{K}}C & A_{\mathsf{K}} \end{bmatrix} \text{ is stable}$$

$$\iff \quad \exists P \succ 0,\; P\begin{bmatrix} A & BC_{\mathsf{K}} \\ B_{\mathsf{K}}C & A_{\mathsf{K}} \end{bmatrix}^{\mathsf{T}} + \begin{bmatrix} A & BC_{\mathsf{K}} \\ B_{\mathsf{K}}C & A_{\mathsf{K}} \end{bmatrix} P \prec 0,$$

Change of variables for output feedback control is highly non-trivial

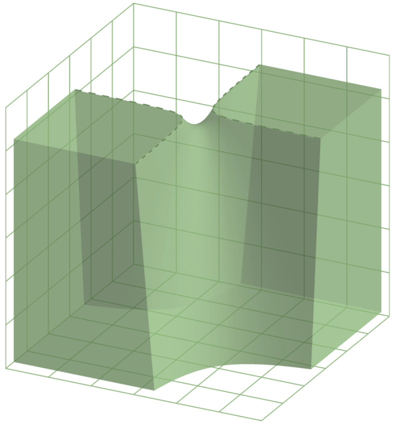
[Scherer et al., IEEE TAC 1997]
[Gahinet and Apkarian, 1994]

Proof idea: Lifting via Change of Variables

- **Change of variables in state-space domain: Lyapunov theory**

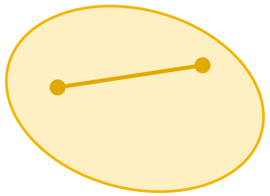
[Scherer et al., IEEE TAC 1997]
[Gahinet and Apkarian, 1994]

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



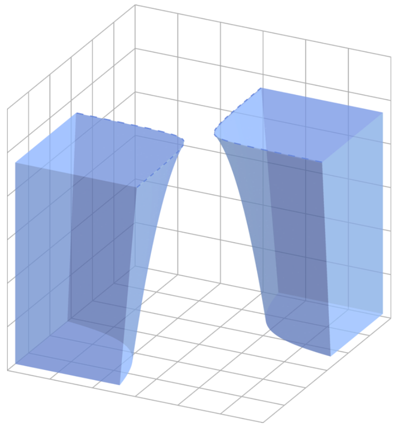
at most 2 connected components

$$\mathcal{F} \times \mathrm{GL} \xrightarrow[\text{surjective}]{\Phi} \mathcal{C}_{\text{full}}$$



Convex thus connected

General linear group: the set of invertible matrices (similarity transformation)



2 connected components

$$\mathrm{GL}_n^+ = \{\Pi \in \mathbb{R}^{n\times n} \mid \det \Pi > 0\},$$
$$\mathrm{GL}_n^- = \{\Pi \in \mathbb{R}^{n\times n} \mid \det \Pi < 0\}.$$

Comparison with LQR

	LQR as an Optimization problem $$\min_{K} \; J(K) \quad \text{s.t.} \; K \in \mathcal{K}$$	LQG as an Optimization problem $$\min_{\mathsf{K}} \; J(\mathsf{K}) \quad \text{s.t.} \; \mathsf{K} = (A_{\mathsf{K}}, B_{\mathsf{K}}, C_{\mathsf{K}}) \in \mathcal{C}_{\text{full}}$$
Connectivity of feasible region	❖ Always connected	❖ Disconnected, but at most 2 connected comp. ❖ They are almost identical to each other
Stationary points	❖ Unique	❖ Non-unique, non-isolated stationary points ❖ Spurious stationary points (saddle, nonminimal controller) ❖ **All mini. stationary points are globally optimal**
Gradient Descent	❖ Gradient dominance ❖ Global fast convergence (like strictly convex)	❖ No gradient dominance ❖ Local convergence/speed (**unknown**) ❖ **Many open questions**
References	**Fazel et al., ICML, 2018**; Malik et al., 2019; Mohammadi et al., IEEE TAC, 2020; Li et al., 2019; Zhang et al., 2019; Furieri et al., 2019; Feiran Zhao & Keyou You, 2021, and many others	Zheng, Tang, Li. 2021, link

Comparison with LQR

	LQR	IOP
	$\min_{\mathbf{K}} \sup_{\|\Delta_A\|,\|\Delta_B\|<\epsilon} \lim_{T\to\infty} \mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T}(x_t^{\mathsf{T}}Qx_t + u_t^{\mathsf{T}}Ru_t)\right]$ subject to $x_{t+1} = (\hat{A}+\Delta A)x_t + (\hat{B}+\Delta B)u_t + v_t$, $\mathbf{u} = \mathbf{K}\mathbf{x}$	$\min_{\mathbf{K}} \sup_{\|\mathbf{\Delta}\|_\infty<\epsilon} \lim_{T\to\infty} \mathbb{E}\left[\frac{1}{T}\sum_{t=0}^{T}(y_t^{\mathsf{T}}Qy_t + u_t^{\mathsf{T}}Ru_t)\right]$ subject to $\mathbf{y} = (\hat{\mathbf{G}}+\mathbf{\Delta})\mathbf{u} + \mathbf{v}$, $\mathbf{u} = \mathbf{K}\mathbf{y} + \mathbf{w}$,
Sys ID methods	❖ Least squares $\|\hat{A} - A_\star\| \leq \epsilon_A, \|\hat{B} - B_\star\| \leq \epsilon_B,$	❖ Least squares $\|\mathbf{\Delta}\|_\infty := \|\mathbf{G}_\star - \hat{\mathbf{G}}\|_\infty < \epsilon$
Synthesis Technique	❖ Frequency domain ❖ System-level synthesis, SLS (Wang et al., 2019) ❖ Taylor expansion	❖ Frequency domain ❖ Input-output parameterization, IOP, (Furieri et al., 2019) ❖ Taylor expansion
Sample Complexity	❖ both stable and unstable systems $\frac{J(\hat{K}) - J_\star}{J_\star} \sim \mathcal{O}\left(\frac{1}{\sqrt{N}}\right),$	❖ Only for open-loop stable system $\frac{J(\hat{\mathbf{K}}) - J_\star}{J_\star} \sim \mathcal{O}\left(\frac{1}{\sqrt{N}}\right),$
References	✓ Dean et al., 2020; Berberich et al., 2020; Boczar et al., 2018; Tsiamis et al., 2020; Umenberger et al., 2019; and many others	• Zheng, Furieri, Kamgarpour, & Li, (2021, May). link