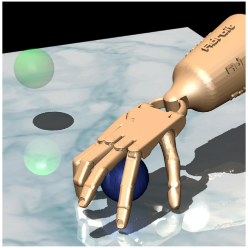
# Beyond Convexification of LQR: a Generalized Framework

Yue Sun, Maryam Fazel

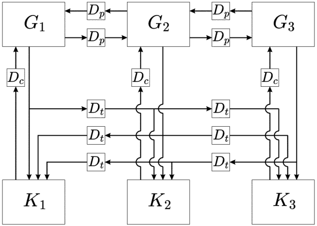
Department of Electrical and Computer Engineering
University of Washington, Seattle

May 7, 2021

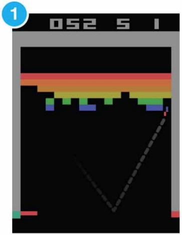
# Optimal control



Robot




Decentralized control




Game

# Linear quadratic regulator – Continuous

- Dynamics:

$$\dot{x}(t) = Ax(t) + Bu(t), \ x(0) = x_0,$$

  $x$ is state, $u$ is input (the controller).

- Loss/Objective:

$$loss(u(t)) := \mathbf{E}_{x_0 \sim \mathcal{N}(0,\Omega)} \int_0^\infty (x(t)^\top Q x(t) + u(t)^\top R u(t)) dt$$

  State tends to 0 while the energy of input is small.

# Linear quadratic regulator – Continuous

- Dynamics:

$$\dot{x}(t) = Ax(t) + Bu(t),\ x(0) = x_0,$$

  $x$ is state, $u$ is input (the controller).

- Loss/Objective:

$$loss(u(t)) := \mathbf{E}_{x_0 \sim \mathcal{N}(0,\Omega)} \int_0^{\infty} (x(t)^\top Qx(t) + u(t)^\top Ru(t))dt$$

  State tends to 0 while the energy of input is small.

- Optimal controller: state feedback

$$u = K^* x = -R^{-1}BPx,$$
$$AP + PA^\top + Q - PBR^{-1}BP = 0.$$

  Solved by Riccati equations.

# Linear quadratic regulator – Continuous

- Optimal control

$$\dot{x}(t) = Ax(t) + Bu(t),\ x(0) = x_0,$$
$$loss(u(t)) = \mathbf{E}_{x_0 \sim \mathcal{N}(0,\Omega)} \int_0^\infty (x(t)^\top Q x(t) + u(t)^\top R u(t)) dt$$

# Linear quadratic regulator – Continuous

- Optimal control

$$\dot{x}(t) = Ax(t) + Bu(t),\ x(0) = x_0,$$
$$loss(u(t)) = \mathbf{E}_{x_0 \sim \mathcal{N}(0,\Omega)} \int_0^{\infty} (x(t)^\top Q x(t) + u(t)^\top R u(t)) dt$$

- Convexification

$$\begin{aligned} \min_{Z,L,G} \ & f(L,G,Z) := \mathbf{Tr}(QG + ZR) \\ & \text{s.t., } \mathcal{A}(G) + \mathcal{B}(L) + \Omega = 0,\ G \succ 0, \\ & \begin{bmatrix} Z & L^\top \\ L & G \end{bmatrix} \succeq 0 \end{aligned}$$

And $K^* = L^* G^{*-1}$.

# Policy gradient descent – motivation and related works

**Policy gradient descent:** $K^+ = K - \eta \nabla \mathit{loss}(K)$. Gradient descent on nonconvex objective

- In today's machine learning, people run gradient descent on nonconvex functions.
- GD is more implementable than solving SDP (e.g., by interior point method).

# Policy gradient descent – motivation and related works

**Policy gradient descent:** $K^+ = K - \eta \nabla \mathit{loss}(K)$. Gradient descent on nonconvex objective

- In today's machine learning, people run gradient descent on nonconvex functions.
- GD is more implementable than solving SDP (e.g., by interior point method).

**Policy gradient descent, LQR, continuous:** Mohammadi et al. 2019; Bu, A. Mesbahi, and M. Mesbahi 2020
**Policy gradient descent, LQR, discrete:** Fazel et al. 2018; Bu, A. Mesbahi, Fazel, et al. 2019
**Policy gradient descent, robust LQR, continuous:** Zhang, Hu, and Basar 2020
**Policy gradient descent, structured finite horizon LQR, discrete:** Furieri, Zheng, and Kamgarpour 2020

# Policy gradient descent – motivation and related works

**Policy gradient descent, LQR, continuous:** Mohammadi et al. 2019; Bu, A. Mesbahi, and M. Mesbahi 2020
**Policy gradient descent, LQR, discrete:** Fazel et al. 2018; Bu, A. Mesbahi, Fazel, et al. 2019
**Policy gradient descent, robust LQR, continuous:** Zhang, Hu, and Basar 2020
**Policy gradient descent, structured finite horizon LQR, discrete:** Furieri, Zheng, and Kamgarpour 2020

**Motivation of our work:**

- Each paper solves one optimal control problem – not easy to generalize, not including optimal $\mathcal{H}_2$ control.
- The proof requires much computation, the intuition is not clear and not linked to convexification.

# Policy gradient descent – motivation and related works

**Policy gradient descent, LQR, continuous:** Mohammadi et al. 2019; Bu, A. Mesbahi, and M. Mesbahi 2020
**Policy gradient descent, LQR, discrete:** Fazel et al. 2018; Bu, A. Mesbahi, Fazel, et al. 2019
**Policy gradient descent, robust LQR, continuous:** Zhang, Hu, and Basar 2020
**Policy gradient descent, structured finite horizon LQR, discrete:** Furieri, Zheng, and Kamgarpour 2020

**Motivation of our work:**

- Each paper solves one optimal control problem – not easy to generalize, not including optimal $\mathcal{H}_2$ control.
- The proof requires much computation, the intuition is not clear and not linked to convexification.

If the optimal control problem can be convexified, does policy gradient descent converge to global optimum?

# Distilling the properties of convexification

$$\begin{aligned}\min_{Z,L,G} \quad & f(L,G,Z) := \mathbf{Tr}(QG+ZR) \\ \text{s.t.,} \quad & \mathcal{A}(G)+\mathcal{B}(L)+\Omega=0, \\ & G \succ 0, \\ & \begin{bmatrix} Z & L \\ L^\top & G \end{bmatrix} \succeq 0 \end{aligned} \quad \Longrightarrow \quad \begin{aligned}\min_{Z,L,G} \quad & f(L,G,Z), \\ \text{s.t.,} \quad & (L,G,Z)\in\mathcal{S}\end{aligned}$$

The following assumptions are required:

1. $\mathcal{S}$ is convex.
2. $f(L,G,Z)$ is convex (special: linear, strongly convex) on $\mathcal{S}$.

# Distilling the properties of convexification

$$\begin{aligned}\min_{Z,L,G}\ & f(L,G,Z) := \mathbf{Tr}(QG + ZR)\\ \text{s.t., } & \mathcal{A}(G) + \mathcal{B}(L) + \Omega = 0,\\ & G \succ 0,\\ & \begin{bmatrix} Z & L \\ L^\top & G \end{bmatrix} \succeq 0 \end{aligned} \quad \Longrightarrow \quad \begin{aligned}\min_{Z,L,G}\ & f(L,G,Z),\\ \text{s.t., } & (L,G,Z) \in \mathcal{S}\end{aligned}$$

Either of the following holds.

3. 3.1 Let $G$ be invertible, $K = LG^{-1}$ defines a bijection $K \leftrightarrow (L, G)$. For any such bijection $K \leftrightarrow (L, G)$, $\exists Z$, such that $(L, G, Z) \in \mathcal{S}$.
   3.2 $loss(K) = \min_Z\ f(L, G, Z)$ subject to $(L, G, Z) \in \mathcal{S}$.
4. We can express $loss(K)$ as:

$$\begin{aligned} loss(K) = \min_{L,G,Z}\ & f(L,G,Z)\\ \text{s.t., } & (L,G,Z) \in \mathcal{S},\ K = LG^{-1}. \end{aligned}$$

More generally, $K = LG^{-1}$ can be replaced by a general map $K = F(L, G)$ with well-defined first order derivative (handles structured LQR [Furieri, Zheng, and Kamgarpour 2020]).

# Theorem

$$\begin{aligned} \min_{Z,L,G} \quad & f(L, G, Z), \\ \text{s.t.,} \quad & (L, G, Z) \in \mathcal{S} \end{aligned}$$

**Theorem (simplified)**

Under assumptions 1,2,3 or 1,2,4, $\nabla loss(K) = 0 \iff K = K^*$. More specifically,

1. If $f$ is linear, $\|\nabla loss(K)\| \gtrsim loss(K) - loss(K^*)$.
2. If $f$ is $\mu$ strongly convex, $\|\nabla loss(K)\| \gtrsim (\mu(loss(K) - loss(K^*)))^{1/2}$.

# Minimizing $\mathcal{L}_2$ gain

- Dynamics

$$\dot{x}(t) = Ax(t) + Bu(t) + B_w w(t), \;\; y = Cx(t) + Du(t)$$

$x$ is state, $u$ is input, $w$ is a perturbation. We hope to find the optimal state feedback controller $u(t) = K^* x(t)$ that minimizes the following loss.

- Loss/Objective:

$$loss(K) := \sup_{\|w\|_2 = 1} \|y\|_2.$$

# Minimizing $\mathcal{L}_2$ gain

- Dynamics

$$\dot{x}(t) = Ax(t) + Bu(t) + B_w w(t), \;\; y = Cx(t) + Du(t)$$

  $x$ is state, $u$ is input, $w$ is a perturbation. We hope to find the optimal state feedback controller $u(t) = K^* x(t)$ that minimizes the following loss.

- Loss/Objective:

$$loss(K) := \sup_{\|w\|_2 = 1} \|y\|_2.$$

- Convexification

$$\min_{L,G,\gamma} \; f(L, G, \gamma) := \gamma$$

$$\text{s.t.,} \; \begin{bmatrix} AG + GA^\top + BL + L^\top B^\top + B_w B_w^\top & (CG + DL)^\top \\ CG + DL & -\gamma^2 I \end{bmatrix} \preceq 0,$$

$$\gamma \geq 0.$$

And $K^* = L^* {G^*}^{-1}$.

# Maximizing dissipativity

- Dynamics

$$\dot{x}(t) = Ax(t) + Bu(t) + B_w w(t),\ y = Cx(t) + Du(t)$$

$x$ is state, $u$ is input, $w$ is a perturbation.
Dissipativity $\eta$:

$$\int_0^T w^\top y - \eta w^\top w dt \geq 0,\ \forall T > 0.$$

# Maximizing dissipativity

- Dynamics

$$\dot{x}(t) = Ax(t) + Bu(t) + B_w w(t), \;\; y = Cx(t) + Du(t)$$

$x$ is state, $u$ is input, $w$ is a perturbation.
Dissipativity $\eta$:

$$\int_0^T w^\top y - \eta w^\top w dt \geq 0, \;\; \forall T > 0.$$

- Convexification

$$\max_{L,G,\eta} \; f(L,G,\eta) := \eta$$
$$\text{s.t.,} \; \begin{bmatrix} AG + GA^\top + BL + L^\top B^\top & B_w - GC^\top \\ B_w^\top - CG & 2\eta I - (D + D^\top) \end{bmatrix} \preceq 0.$$

And $K^* = L^* G^{*-1}$.

# System level synthesis (SLS)

(Anderson et al. 2019) Dynamics $x(t+1) = A(t)x(t) + B(t)u(t) + w(t)$.

# System level synthesis (SLS)

(Anderson et al. 2019) Dynamics $x(t+1) = A(t)x(t) + B(t)u(t) + w(t)$.

Finite horizon, time varying, discrete LQR
Search for $u(t) = \sum_{i=0}^{t} K(t, t-i)x(i)$.

# System level synthesis (SLS)

(Anderson et al. 2019) Dynamics $x(t+1) = A(t)x(t) + B(t)u(t) + w(t)$.

Finite horizon, time varying, discrete LQR
Search for $u(t) = \sum_{i=0}^{t} K(t, t-i)x(i)$.

$$X = \begin{bmatrix} x(0) \\ ... \\ x(T) \end{bmatrix}, \ U = \begin{bmatrix} u(0) \\ ... \\ u(T) \end{bmatrix}, \ W = \begin{bmatrix} x(0) \\ w(0) \\ ... \\ w(T-1) \end{bmatrix},$$

$$Z = \begin{bmatrix} 0 & 0 & ... & 0 & 0 \\ I & 0 & ... & 0 & 0 \\ 0 & I & ... & 0 & 0 \\ ... & & & & \\ 0 & 0 & ... & I & 0 \end{bmatrix}, \ \mathcal{K} = \begin{bmatrix} K(0,0) & 0 & ... & 0 \\ K(1,1) & K(1,0) & ... & 0 \\ ... & & & \\ K(T,T) & K(T,T-1) & ... & K(T,0) \end{bmatrix},$$

$$\mathcal{A} = \mathrm{diag}(A(0), ..., A(T-1), 0), \ \mathcal{B} = \mathrm{diag}(B(0), ..., B(T-1), 0)$$

# System level synthesis (SLS)

(Anderson et al. 2019) Dynamics $x(t+1) = A(t)x(t) + B(t)u(t) + w(t)$.

Finite horizon, time varying, discrete LQR
Search for $u(t) = \sum_{i=0}^{t} K(t, t-i)x(i)$.

$$X = \begin{bmatrix} x(0) \\ \dots \\ x(T) \end{bmatrix}, \; U = \begin{bmatrix} u(0) \\ \dots \\ u(T) \end{bmatrix}, \; W = \begin{bmatrix} x(0) \\ w(0) \\ \dots \\ w(T-1) \end{bmatrix},$$

$$Z = \begin{bmatrix} 0 & 0 & \dots & 0 & 0 \\ I & 0 & \dots & 0 & 0 \\ 0 & I & \dots & 0 & 0 \\ \dots & & & & \\ 0 & 0 & \dots & I & 0 \end{bmatrix}, \; \mathcal{K} = \begin{bmatrix} K(0,0) & 0 & \dots & 0 \\ K(1,1) & K(1,0) & \dots & 0 \\ \dots & & & \\ K(T,T) & K(T,T-1) & \dots & K(T,0) \end{bmatrix},$$

$$\mathcal{A} = \mathrm{diag}(A(0), \dots, A(T-1), 0), \; \mathcal{B} = \mathrm{diag}(B(0), \dots, B(T-1), 0)$$

Constraint:

$$X = Z\mathcal{A}X + Z\mathcal{B}U + W = Z(\mathcal{A} + \mathcal{B}\mathcal{K})X + W$$

# System level synthesis (SLS)

(Anderson et al. 2019) Dynamics $x(t+1) = A(t)x(t) + B(t)u(t) + w(t)$.

Finite horizon, time varying, discrete LQR
Search for $u(t) = \sum_{i=0}^{t} K(t, t-i)x(i)$.

Constraint: $X = Z\mathcal{A}X + Z\mathcal{B}U + W = Z(\mathcal{A} + \mathcal{B}\mathcal{K})X + W$

# System level synthesis (SLS)

(Anderson et al. 2019) Dynamics $x(t+1) = A(t)x(t) + B(t)u(t) + w(t)$.

Finite horizon, time varying, discrete LQR
Search for $u(t) = \sum_{i=0}^{t} K(t, t-i)x(i)$.

Constraint: $X = Z\mathcal{A}X + Z\mathcal{B}U + W = Z(\mathcal{A} + \mathcal{B}\mathcal{K})X + W$

We define the mapping from $W$ to $X, U$ by

$$\begin{bmatrix} X \\ U \end{bmatrix} = \begin{bmatrix} \Phi_X \\ \Phi_U \end{bmatrix} W.$$

where $\Phi_X, \Phi_U$ are block lower triangular, $\mathcal{K} = \Phi_U \Phi_X^{-1}$. There is a constraint on $\Phi_X, \Phi_U$:

$$\begin{bmatrix} I - Z\mathcal{A} & -Z\mathcal{B} \end{bmatrix} \begin{bmatrix} \Phi_X \\ \Phi_U \end{bmatrix} = I.$$

# System level synthesis (SLS)

(Anderson et al. 2019) Dynamics $x(t+1) = A(t)x(t) + B(t)u(t) + w(t)$.

Finite horizon, time varying, discrete LQR
Search for $u(t) = \sum_{i=0}^{t} K(t, t-i)x(i)$.

Constraint: $\begin{bmatrix} I - Z\mathcal{A} & -Z\mathcal{B} \end{bmatrix} \begin{bmatrix} \Phi_X \\ \Phi_U \end{bmatrix} = I$.

The LQR loss $\mathcal{L}(\mathcal{K}) := \sum_{t=0}^{T} x(t)^\top Q(t)x(t) + u(t)^\top R(t)u(t)$
with $x(0), w(t)$ being i.i.d from $\mathcal{N}(0, \Sigma)$ is

$$f(\Phi_X, \Phi_U) = \left\| \mathrm{diag}(\mathcal{Q}^{1/2}, \mathcal{R}^{1/2}) \begin{bmatrix} \Phi_X \\ \Phi_U \end{bmatrix} \Sigma^{1/2} \right\|_F^2.$$

# System level synthesis (SLS)

(Anderson et al. 2019) Dynamics $x(t+1) = A(t)x(t) + B(t)u(t) + w(t)$.

Finite horizon, time varying, discrete LQR
Search for $u(t) = \sum_{i=0}^{t} K(t, t-i)x(i)$.

Constraint: $\begin{bmatrix} I - Z\mathcal{A} & -Z\mathcal{B} \end{bmatrix} \begin{bmatrix} \Phi_X \\ \Phi_U \end{bmatrix} = I$.

The LQR loss $\mathcal{L}(\mathcal{K}) := \sum_{t=0}^{T} x(t)^\top Q(t)x(t) + u(t)^\top R(t)u(t)$
with $x(0), w(t)$ being i.i.d from $\mathcal{N}(0, \Sigma)$ is

$$f(\Phi_X, \Phi_U) = \left\| \mathrm{diag}(\mathcal{Q}^{1/2}, \mathcal{R}^{1/2}) \begin{bmatrix} \Phi_X \\ \Phi_U \end{bmatrix} \Sigma^{1/2} \right\|_F^2 .$$

**Convex:** Minimize $f(\Phi_X, \Phi_U)$ subject to $\begin{bmatrix} I - Z\mathcal{A} & -Z\mathcal{B} \end{bmatrix} \begin{bmatrix} \Phi_X \\ \Phi_U \end{bmatrix} = I$.

Recover $\mathcal{K}$ by $\mathcal{K} = \Phi_U \Phi_X^{-1}$.

# Distilling the properties of convexification

$$\begin{aligned}\min_{Z,L,G} \ & f(L,G,Z) := \mathbf{Tr}(QG + ZR) \\ \text{s.t.,} \ & \mathcal{A}(G) + \mathcal{B}(L) + \Omega = 0, \\ & G \succ 0, \\ & \begin{bmatrix} Z & L \\ L^\top & G \end{bmatrix} \succeq 0 \end{aligned} \quad \Longrightarrow \quad \begin{aligned}\min_{Z,L,G} \quad & f(L,G,Z), \\ \text{s.t.,} \quad & (L,G,Z) \in \mathcal{S}\end{aligned}$$

3. 3.1 Let $G$ be invertible, $K = LG^{-1}$ defines a bijection $K \leftrightarrow (L, G)$. For any such bijection $K \leftrightarrow (L, G)$, $\exists Z$, such that $(L, G, Z) \in \mathcal{S}$.
   3.2 $loss(K) = \min_Z \ f(L, G, Z)$ subject to $(L, G, Z) \in \mathcal{S}$.

# Proof sketch

$$\min_{K} \quad loss(K), \qquad \text{s.t.,} \quad K \text{ stabilizes}$$

$$\Rightarrow$$

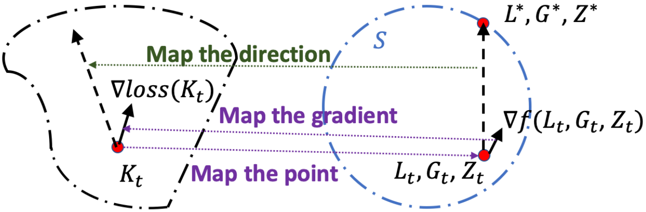
$$\min_{Z,L,G} \quad f(L,G,Z), \qquad \text{s.t.,} \quad (L,G,Z) \in \mathcal{S}$$

$$\Downarrow$$

$$\|\mathcal{P}_{\mathcal{S}} \nabla f(L,G,Z)\| \gtrsim \Phi(f(L,G,Z) - f(L^*,G^*,Z^*))$$

$$\Leftarrow$$

$$\|\nabla loss(K)\| \gtrsim \Phi(loss(K) - loss(K^*))$$



# Conclusion

- For a family of optimal control problems that can be convexified, policy gradient descent converges to the global optimum despite the nonconvexity.
- We propose a concise proof that bridges the nonconvex landscape with the convexified problems.

**Future:** Understanding the discrete optimal control problem. Beat the convergence rate with related papers on specific problems.

# References I


- Anderson, James et al. (2019). “System level synthesis”. In: *Annual Reviews in Control* 47, pp. 364–393.
- Bu, Jingjing, Afshin Mesbahi, Maryam Fazel, et al. (2019). “LQR through the lens of first order methods: Discrete-time case”. In: *arXiv preprint arXiv:1907.08921*.
- Bu, Jingjing, Afshin Mesbahi, and Mehran Mesbahi (2020). “Policy gradient-based algorithms for continuous-time linear quadratic control”. In: *arXiv preprint arXiv:2006.09178*.
- Fazel, Maryam et al. (2018). “Global Convergence of Policy Gradient Methods for Linearized Control Problems”. In: *arXiv preprint arXiv:1801.05039*.
- Furieri, Luca, Yang Zheng, and Maryam Kamgarpour (2020). “Learning the globally optimal distributed LQ regulator”. In: *Learning for Dynamics and Control*. PMLR, pp. 287–297.

# References II

Mohammadi, Hesameddin et al. (2019). "Convergence and sample complexity of gradient methods for the model-free linear quadratic regulator problem". In: *arXiv preprint arXiv:1912.11899*.

Zhang, Kaiqing, Bin Hu, and Tamer Basar (2020). "Policy Optimization for $\mathcal{H}_2$ Linear Control with $\mathcal{H}_\infty$ Robustness Guarantee: Implicit Regularization and Global Convergence". In: *Learning for Dynamics and Control*, pp. 179–190.