Actuator Placement for Optimizing Performance under Controllability Constraints

B. Guo*, Orcun Karaca*, T. Summers†, M. Kamgarpour*

*Institut für Automatik, ETH Zürich
†Dept. of Mechanical Eng., UT Dallas

58th Conference on Decision and Control
Dec 13th, 2019, Nice
Efficient operation of large-scale dynamical networks

- Large-scale complex dynamical networks:
  - Critical infrastructure
  - Multi-robot + transportation
  - Industrial manufacturing
  - Eco, bio, econ, social

- A fundamental problem: **Actuator placement**
Efficient operation of large-scale dynamical networks

- Large-scale complex dynamical networks:
  - critical infrastructure
  - multi-robot + transportation
  - industrial manufacturing
  - eco, bio, econ, social

- A fundamental problem: **Actuator placement**
  
  *Find a subset from a finite set of placements*
Efficient operation of large-scale dynamical networks

- Large-scale complex dynamical networks:
  - Critical infrastructure
  - Multi-robot + transportation
  - Industrial manufacturing
  - Eco, bio, econ, social

- A fundamental problem: **Actuator placement**
  
  *Find a subset from a finite set of placements*
  
  - Minimizing the control energy
Efficient operation of large-scale dynamical networks

- Large-scale complex dynamical networks:
  - Critical infrastructure
  - Multi-robot + transportation
  - Industrial manufacturing
  - Eco, bio, econ, social

- A fundamental problem: **Actuator placement**
  - Find a subset from a finite set of placements
  - Minimizing the control energy
  - Satisfying a cardinality bound & guaranteeing controllability
Outline

Problem formulation

Greedy algorithm and performance guarantees

Feasibility check methods for controllability matroid

Numerical case studies

Conclusion
Outline

Problem formulation

Greedy algorithm and performance guarantees

Feasibility check methods for controllability matrices

Numerical case studies

Conclusion
System model

Continuous linear time-invariant system model with

**State** $x \in \mathbb{R}^n$: $x_i$ associated with a node $v_i \in V = \{v_1, \ldots, v_n\}$

**Control input** $u \in \mathbb{R}^n$: $u_i$ can be exerted at $v_i \in V$

Given the actuator set $S \subset V$

$$\dot{x} = Ax + B(S)u,$$

where $B(S) := \text{diag}(1(S)) \in \mathbb{R}^{n \times n}$

Example: $B(\{v_3\}) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$
System model

- Continuous linear time-invariant system model with
  **State** $x \in \mathbb{R}^n$: $x_i$ associated with a node $v_i \in V = \{v_1, \ldots, v_n\}$
  **Control input** $u \in \mathbb{R}^n$: $u_i$ can be exerted at $v_i \in V$

- Given the actuator set $S \subset V$

\[
\dot{x} = Ax + B(S)u,
\]

where $B(S) := \text{diag}(1(S)) \in \mathbb{R}^{n \times n}$

- Graph representation

\[
G = (V, E), \text{ directed, unweighted}
\]

The edge $(v_j, v_i) \in E$ if $(A)_{ij} \neq 0$

**Assumption:** $G$ is strongly connected
System model

- Continuous linear time-invariant system model with
  - **State** $x \in \mathbb{R}^n$: $x_i$ associated with a node $v_i \in V = \{v_1, \ldots, v_n\}$
  - **Control input** $u \in \mathbb{R}^n$: $u_i$ can be exerted at $v_i \in V$

- Given the actuator set $S \subset V$

\[
\dot{x} = Ax + B(S)u,
\]

where $B(S) := \text{diag}(1(S)) \in \mathbb{R}^{n \times n}$

- Graph representation example:

\[
A = \begin{bmatrix}
0 & -0.5 & -0.8 & -0.6 \\
1 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 \\
\end{bmatrix}
\]
Controllability requirement

Controllability:
Can be determined through controllability matrix

\[ P = \begin{bmatrix} B(S) & AB(S) & \cdots & A^{n-1}B(S) \end{bmatrix} \in \mathbb{R}^{n \times n^2} \]

Sensitive to parameter perturbations.
Controllability requirement

- Controllability:
  Can be determined through controllability matrix

\[
P = \begin{bmatrix} B(S) & AB(S) & \cdots & A^{n-1}B(S) \end{bmatrix} \in \mathbb{R}^{n \times n^2}
\]

Sensitive to parameter perturbations.

- Structural controllability [Lin 1974]:
Controllability requirement

- Controllability: Can be determined through controllability matrix

\[ P = \begin{bmatrix} B(S) & AB(S) & \cdots & A^{n-1}B(S) \end{bmatrix} \in \mathbb{R}^{n \times n^2} \]

Sensitive to parameter perturbations.

- Structural controllability [Lin 1974]:

The pair \((A, B(S))\) is \textit{structurally controllable} if and only if \textit{almost all} of the pairs with \textit{the same structure} are controllable.
Controllability requirement

- Controllability:
  Can be determined through controllability matrix

\[ P = \begin{bmatrix} B(S) & AB(S) & \cdots & A^{n-1}B(S) \end{bmatrix} \in \mathbb{R}^{n \times n^2} \]

Sensitive to parameter perturbations.

- Structural controllability [Lin 1974]:
  The pair \((A, B(S))\) is \textit{structurally controllable} if and only if \textit{almost all} of the pairs with \textit{the same structure} are controllable

  - \textbf{The same structure} refers to positions of the nonzero entries
Controllability requirement

- **Controllability:**
  Can be determined through controllability matrix

\[
P = \begin{bmatrix} B(S) & AB(S) & \cdots & A^{n-1}B(S) \end{bmatrix} \in \mathbb{R}^{n \times n^2}
\]

Sensitive to parameter perturbations.

- **Structural controllability** [Lin 1974]:
  The pair \((A, B(S))\) is *structurally controllable* if and only if *almost all* of the pairs with *the same structure* are controllable
  - **The same structure** refers to positions of the nonzero entries
  - Can be determined through the graph \(G\)
Network performance metric

▶ Controllability metric

Average minimum energy required to steer the system from $x_0 = 0$ at time $t = 0$ to any state $||x||_2 = 1$ at time $T$:

$$F(S) = \text{tr}(W_T^{-1}(S)),$$

where $W_T$ is the controllability Gramian of $(A, B(S))$.
Network performance metric

- **Controllability metric**
  
  Average minimum energy required to steer the system from $x_0 = 0$ at time $t = 0$ to any state $\|x\|_2 = 1$ at time $T$:

  $$F(S) = \text{tr}(W_{T}^{-1}(S)),$$

  where $W_T$ is the controllability Gramian of $(A, B(S))$

- $W_T(S)$ is invertible if and only if $(A, B(S))$ is controllable
Network performance metric

- **Controllability metric**
  Average minimum energy required to steer the system from $x_0 = 0$ at time $t = 0$ to any state $\|x\|_2 = 1$ at time $T$:

  $$F(S) = \text{tr}(W_T^{-1}(S)),$$

  where $W_T$ is the controllability Gramian of $(A, B(S))$

- $W_T(S)$ is invertible if and only if $(A, B(S))$ is controllable

- To analyze uncontrollable systems, introduce a small $\epsilon \in \mathbb{R}_{>0}$:

  $$F_\epsilon(S) = \text{tr}((W_T(S) + \epsilon I)^{-1})$$
Network performance metric

- **Controllability metric**
  
  Average minimum energy required to steer the system from $x_0 = 0$ at time $t = 0$ to any state $\|x\|_2 = 1$ at time $T$:

  $$F(S) = \text{tr}(W_T^{-1}(S)),$$

  where $W_T$ is the controllability Gramian of $(A, B(S))$

  - $W_T(S)$ is invertible if and only if $(A, B(S))$ is controllable

  - To analyze uncontrollable systems, introduce a small $\epsilon \in \mathbb{R}_{>0}$:

    $$F_\epsilon(S) = \text{tr}((W_T(S) + \epsilon I)^{-1})$$

  **Lemma 1**

  $F_\epsilon(S)$ is strictly decreasing
Problem statement

Given a $K$ number of actuators, our main problem is

$$\min_{S \subset V} F_\epsilon(S)$$

s.t. $S \in C_K$, 

where

$$C_K := \{ S \subset V \mid (A, B(S)) \text{ is structurally controllable}, |S| = K \}$$
Problem statement

Given a $K$ number of actuators, our main problem is

$$\min_{S \subset V} F_\epsilon(S)$$

s.t. $S \in \mathcal{C}_K,$

where

$$\mathcal{C}_K := \{S \subset V \mid (A, B(S)) \text{ is structurally controllable, } |S| = K\}$$

No computationally feasible method to calculate the exact optimum

Variants have been shown to be NP-hard

Problem statement

- Given a $K$ number of actuators, our main problem is

\[
\min_{S \subset V} F_\epsilon(S)
\]

\[\text{s.t. } S \in C_K,\]

where

\[C_K := \{S \subset V | (A, B(S)) \text{ is structurally controllable}, |S| = K\}\]

- No computationally feasible method to calculate the exact optimum

- Variants have been shown to be NP-hard

- **Alternative:** heuristics to derive an approximate solution
Outline

Problem formulation

Greedy algorithm and performance guarantees

Feasibility check methods for controllability matrix

Numerical case studies

Conclusion
Greedy Algorithm

Algorithm description

- Start from $S = \emptyset$
- At each iteration, until $|S| = K$:
  - Look for the node with the largest reduction in the metric
  - Check whether after adding this node to the set $S$, we can still expand the resulting set to a set in $\mathcal{C}_K$?
  - If yes include, otherwise ignore that node

How suboptimal can this solution be?
Greedy Algorithm

▷ Algorithm description
  ▷ Start from $S = \emptyset$
  ▷ At each iteration, until $|S| = K$:
    - Look for the node with the largest reduction in the metric
    - Check whether after adding this node to the set $S$, we can still expand the resulting set to a set in $C_K$?
    - If yes include, otherwise ignore that node

How suboptimal can this solution be?
We need structure; bring in matroids and submodularity!
A reformulation of the feasible region

▶ An **equivalent** problem:

\[
\max_S -F_\epsilon(S)
\]

s.t. \( S \in \tilde{C}_K \).

where \( \tilde{C}_K := \{ \Omega \mid \Omega \subset S \text{ for some } S \in C_K \} \)

▶ **Remark:** Equivalent since \( F_\epsilon \) is strictly decreasing.
A reformulation of the feasible region

▶ An equivalent problem:

$$\max_S -F_\epsilon(S)$$

s.t. \( S \in \tilde{C}_K. \)

where \( \tilde{C}_K := \{ \Omega \mid \Omega \subset S \text{ for some } S \in C_K \} \)

▶ Remark: Equivalent since \( F_\epsilon \) is strictly decreasing.

Theorem 1
\( \mathcal{M} = (V, \tilde{C}_K) \) is a matroid.

▶ Proof idea: Map our problem into a corresponding leader selection problem, invoke a result from [Clark et al. 2012]
A reformulation of the feasible region

▶ An equivalent problem:

$$\max_S -F_\epsilon(S)$$

s.t. $S \in \tilde{C}_K$.

where $\tilde{C}_K := \{\Omega \mid \Omega \subset S \text{ for some } S \in C_K\}$

▶ Remark: Equivalent since $F_\epsilon$ is strictly decreasing.

Theorem 1

$M = (V, \tilde{C}_K)$ is a matroid.

▶ Matroid property we exploit:

If $S_1, S_2 \in \tilde{C}_K$ and $|S_1| < |S_2|$, then there exists $v \in S_2 \setminus S_1$ such that $S_1 \cup v \in \tilde{C}_K$
A reformulation of the feasible region

▶ An equivalent problem:

\[
\max_S -F_\epsilon(S)
\]

s.t. \( S \in \tilde{\mathcal{C}}_K \).

where \( \tilde{\mathcal{C}}_K := \{ \Omega \mid \Omega \subset S \text{ for some } S \in \mathcal{C}_K \} \)

▶ Remark: Equivalent since \( F_\epsilon \) is strictly decreasing.

Theorem 1
\( \mathcal{M} = (V, \tilde{\mathcal{C}}_K) \) is a matroid.

▶ Matroid property we exploit:
If \( S^k, S^* \in \tilde{\mathcal{C}}_K \) and \( |S^k| < |S^*| \), then there exists \( v \in S^* \setminus S^k \) such that \( S^k \cup v \in \tilde{\mathcal{C}}_K \)
Characterizing the objective function

Definition 1

For an increasing set function $f$, the submodularity ratio is the largest $\gamma \in \mathbb{R}_+$ such that for $\forall S, U, \{\omega\} \subset V$

$$\gamma [f(S \cup U \cup \{\omega\}) - f(S \cup U)] \leq f(S \cup \{\omega\}) - f(S)$$

We have $\gamma \in [0, 1]$. Set function $f$ is said to be submodular if $\gamma = 1$ and weakly submodular if $0 < \gamma < 1$. 
Characterizing the objective function

Definition 1

For an increasing set function $f$, submodularity ratio is the largest $\gamma \in \mathbb{R}_+$ such that for $\forall S, U, \{\omega\} \subset V$

$$\gamma [f(S \cup U \cup \{\omega\}) - f(S \cup U)] \leq f(S \cup \{\omega\}) - f(S)$$

We have $\gamma \in [0, 1]$. Set function $f$ is said to be submodular if $\gamma = 1$ and weakly submodular if $0 < \gamma < 1$.

- Is $-F_\epsilon$ submodular?
  
  No [Summers et al. 2018]
Characterizing the objective function

**Definition 1**

*For an increasing set function $f$, submodularity ratio is the largest $\gamma \in \mathbb{R}_+$ such that for $\forall S, U, \{\omega\} \subset V$*

$$\gamma \left[ f(S \cup U \cup \{\omega\}) - f(S \cup U) \right] \leq f(S \cup \{\omega\}) - f(S)$$

*We have $\gamma \in [0, 1]$. Set function $f$ is said to be submodular if $\gamma = 1$ and weakly submodular if $0 < \gamma < 1$.*

- **Is $-F_\epsilon$ submodular?**
  - No [Summers et al. 2018]

- **Is $-F_\epsilon$ weakly submodular?**

**Lemma 2**

$-F_\epsilon$ is weakly submodular with submodularity ratio $\gamma$

- Lower bounds can be derived via eigenvalue inequalities [Summers and Kamgarpour 2019]
A general problem class

- **Generalization of the problem**

  \[
  \max_{S \subseteq V} f(S), \text{ increasing and } \gamma\text{-submodular}
  \]

  s.t. \( S \in \mathcal{F} \), where \( \mathcal{M} = (V, \mathcal{F}) \) is a matroid.

- Many applications in machine learning, e.g., video summarization, splice site detection

- A well-known result by [Fisher et al. 1977]:

  If \( f \) is submodular, then

  \[
  \frac{f(S^G) - f(\emptyset)}{f(S^*) - f(\emptyset)} \geq \frac{1}{2}
  \]

  where \( S^* \) is the optimum, \( S^G \) is the greedy solution
Suboptimality guarantee for greedy algorithm

**Theorem 2**

*If $f$ is $\gamma$-submodular, then*

\[
\frac{f(S^G) - f(\emptyset)}{f(S^*) - f(\emptyset)} \geq \frac{\gamma^3}{\gamma^3 + 1}
\]

- If $\gamma = 1$, we have $1/2$ of [Fisher et al. 1977]
- Holds also for greedy version of the submodularity ratio $\gamma_G$
Suboptimality guarantee for greedy algorithm

Theorem 2
If \( f \) is \( \gamma \)-submodular, then

\[
\frac{f(S^G) - f(\emptyset)}{f(S^*) - f(\emptyset)} \geq \frac{\gamma^3}{\gamma^3 + 1}
\]

- If \( \gamma = 1 \), we have 1/2 of [Fisher et al. 1977]
- Holds also for greedy version of the submodularity ratio \( \gamma_G \)
- Revisiting the actuator placement problem

\[
\frac{F_\epsilon(\emptyset) - F_\epsilon(S^G)}{F_\epsilon(\emptyset) - F_\epsilon(S^*)} \geq \frac{\gamma^3}{\gamma^3 + 1}, \text{ or equivalently}
\]
Suboptimality guarantee for greedy algorithm

**Theorem 2**

*If \( f \) is \( \gamma \)-submodular, then*

\[
\frac{f(S^G) - f(\emptyset)}{f(S^*) - f(\emptyset)} \geq \frac{\gamma^3}{\gamma^3 + 1}
\]

- If \( \gamma = 1 \), we have 1/2 of [Fisher et al. 1977]
- Holds also for greedy version of the submodularity ratio \( \gamma_G \)
- Revisiting the actuator placement problem

\[
\frac{F_\epsilon(\emptyset) - F_\epsilon(S^G)}{F_\epsilon(\emptyset) - F_\epsilon(S^*)} \geq \frac{\gamma^3}{\gamma^3 + 1}, \text{ or equivalently}
\]

\[
F_\epsilon(S^G) \leq \frac{1}{\gamma^3 + 1} F_\epsilon(\emptyset) + \frac{\gamma^3}{\gamma^3 + 1} F_\epsilon(S^*)
\]

where \( F_\epsilon(\emptyset) = n\epsilon^{-1} \)
Outline

Problem formulation

Greedy algorithm and performance guarantees

Feasibility check methods for controllability matroid

Numerical case studies

Conclusion
Feasibility check for structural controllability

- We need **computationally tractable** methods to check whether a set belongs to $\tilde{C}_K$
- Adapting a result from leader selection problems

**Theorem 3 (Liu et al. 2011, Theorem 2)**

$S \in C_K$ if and only if $|S| = K$ and there exists a perfect matching in $H_b(S)$.
Feasibility check for structural controllability

- We need **computationally tractable** methods to check whether a set belongs to $\tilde{C}_K$

- Adapting a result from leader selection problems

**Theorem 3** (Liu et al. 2011, Theorem 2)

$S \in C_K$ if and only if $|S| = K$ and there exists a perfect matching in $\mathcal{H}_b(S)$.

\[
\{v_3, v_4\} \in C_2
\]
Our method of feasibility check

▶ This result does not directly answer whether $S \in \tilde{C}_K$

**Theorem 4**

$S \in \tilde{C}_K$ if and only if $|m(S)| \geq n - (K - k)$, where $m(S)$ is a maximum matching in $\mathcal{H}_b(S)$, and $|S| = k$.

▶ Can efficiently be done by solving a max-flow problem (e.g., polynomial-time Edmonds-Karp algorithm)
Our method of feasibility check

- This result does not directly answer whether $S \in \tilde{C}_K$

**Theorem 4**

$S \in \tilde{C}_K$ if and only if $|m(S)| \geq n - (K - k)$, where $m(S)$ is a maximum matching in $\mathcal{H}_b(S)$, and $|S| = k$.

- Can efficiently be done by solving a max-flow problem (e.g., polynomial-time Edmonds-Karp algorithm)

**Corollary 1**

$C_K$ is nonempty if and only if $|m(\emptyset)| \geq n - K$, where $m(\emptyset)$ is a maximum matching in $\mathcal{H}_b(\emptyset)$

- Coincides with [Pequito et al. 2013, Theorem 3]
Our method of feasibility check

- This result does not directly answer whether $S \in \tilde{C}_K$

**Theorem 4**
$S \in \tilde{C}_K$ if and only if $|m(S)| \geq n - (K - k)$, where $m(S)$ is a maximum matching in $\mathcal{H}_b(S)$, and $|S| = k$.

- Can efficiently be done by solving a max-flow problem (e.g., polynomial-time Edmonds-Karp algorithm)

**Corollary 1**
$C_K$ is nonempty if and only if $|m(\emptyset)| \geq n - K$, where $m(\emptyset)$ is a maximum matching in $\mathcal{H}_b(\emptyset)$

- Coincides with [Pequito et al. 2013, Theorem 3]
- Can be used to set a feasible $K$ by observing $K \geq n - |m(\emptyset)|$
Outline

Problem formulation

Greedy algorithm and performance guarantees

Feasibility check methods for controllability matrices

Numerical case studies

Conclusion
Numerical Results

- A 23-node network
  Undirected with $\epsilon = 1.9 \times 10^{-4}$, $T = 2$, and $K = 8$

<table>
<thead>
<tr>
<th>Node</th>
<th>1</th>
<th>2</th>
<th>⋯</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>⋯</th>
<th>23</th>
</tr>
</thead>
<tbody>
<tr>
<td>Degree</td>
<td>1</td>
<td>2</td>
<td>⋯</td>
<td>11</td>
<td>12</td>
<td>11</td>
<td>⋯</td>
<td>1</td>
</tr>
</tbody>
</table>

Ordered greedy choices: 16, 13, 5, 8, 6, 20, 10, 21
Guarantee and insights on total degrees of actuator sets

- Can efficiently compute $\gamma_G = 1$
- The guarantee is given by

$$9226.5 = F_\epsilon(S^G) \leq 0.5F_\epsilon(\emptyset) + 0.5F_\epsilon(S^*)$$

where $F_\epsilon(\emptyset) = 1.2 \times 10^5$, and $F_\epsilon(S^*) = 6052.7$

- The guarantee is affected by the large $F_\epsilon(\emptyset)$!
Guarantee and insights on total degrees of actuator sets

- Can efficiently compute $\gamma_G = 1$
- The guarantee is given by

$$9226.5 = F_\epsilon(S^G) \leq 0.5F_\epsilon(\emptyset) + 0.5F_\epsilon(S^*)$$

where $F_\epsilon(\emptyset) = 1.2 \times 10^5$, and $F_\epsilon(S^*) = 6052.7$

- The guarantee is affected by the large $F_\epsilon(\emptyset)$!
- Greedy algorithm has a tendency to pick high degree nodes

<table>
<thead>
<tr>
<th>Rand. Examples</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Greedy</td>
<td>60</td>
<td>50</td>
<td>49</td>
<td>53</td>
<td>56</td>
<td>57</td>
<td>41</td>
<td>76</td>
<td>57</td>
</tr>
<tr>
<td>Optimal</td>
<td>53</td>
<td>47</td>
<td>41</td>
<td>43</td>
<td>46</td>
<td>54</td>
<td>42</td>
<td>50</td>
<td>44</td>
</tr>
</tbody>
</table>

Table: Total degrees of different solutions in randomized instances
Guarantee and insights on total degrees of actuator sets

- Can efficiently compute $\gamma_G = 1$
- The guarantee is given by

$$9226.5 = F_\epsilon(S^G) \leq 0.5F_\epsilon(\emptyset) + 0.5F_\epsilon(S^*)$$

where $F_\epsilon(\emptyset) = 1.2 \times 10^5$, and $F_\epsilon(S^*) = 6052.7$

- The guarantee is affected by the large $F_\epsilon(\emptyset)$!
- Greedy algorithm has a tendency to pick high degree nodes

<table>
<thead>
<tr>
<th>Rand. Examples</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
</tr>
</thead>
<tbody>
<tr>
<td>Greedy</td>
<td>60</td>
<td>50</td>
<td>49</td>
<td>53</td>
<td>56</td>
<td>57</td>
<td>41</td>
<td>76</td>
<td>57</td>
</tr>
<tr>
<td>Optimal</td>
<td>53</td>
<td>47</td>
<td>41</td>
<td>43</td>
<td>46</td>
<td>54</td>
<td>42</td>
<td>50</td>
<td>44</td>
</tr>
</tbody>
</table>

Table: Total degrees of different solutions in randomized instances

- Originates from the earlier stages of greedy algorithm
Outline

- Problem formulation
- Greedy algorithm and performance guarantees
- Feasibility check methods for controllability matrices
- Numerical case studies
- Conclusion
Conclusion

▶ Summary
  ▶ Formulated the structural controllability constraints in actuator placement problem as a matroid
  ▶ Obtained performance guarantees for the greedy algorithm
  ▶ Derived methods for feasibility check
  ▶ Illustrated degree-dependence of different solutions

▶ Outlook
  ▶ Tightening the performance guarantee of the greedy algorithm with the curvature
  ▶ Is there a class of networks for which $\gamma_G = 1$ for our metric?
Outlook

- Highlights of our current work
  1. Proposed **reverse greedy** implementation with its corresponding **feasibility check** methods
  2. Provided **an empty-set independent guarantee**:
     \[ \mathcal{O}(N^{\frac{\alpha}{(1-\alpha)\gamma}}), \]
     where \( \alpha \) is the curvature
  3. Proposed **algorithms to pick** \( \varepsilon \) in an optimal manner

- These results appear in
  - Guo, Karaca, Summers, and Kamgarpour, IEEE CDC, 2019
  - Karaca, Guo, and Kamgarpour, ArXiv:1912.04638, 2019
Thank you for your attention

You may contact me: okaraca@ethz.ch