Optimal decentralized stochastic control: A common information approach

Aditya Mahajan
McGill University

Joint work: Ashutosh Nayyar (UIUC) and Demosthenis Teneketzis (Univ of Michigan)

GERAD Seminar, April 23, 2012
Common theme: multi-stage multi-agent decision making under uncertainty
Interconnected Power Systems

Aditya Mahajan

Optimal decentralized stochastic control
Interconnected Power Systems

Region 1  Region 2
Interconnected Power Systems

Region 1

Controller 1

Region 2

Controller 2
Interconnected Power Systems

Region 1

Controller 1

Interconnect

Region 2

Controller 2

Aditya Mahajan

Optimal decentralized stochastic control
Interconnected Power Systems

Region 1

Controller 1

Region 2

Controller 2

Interconnect

Communication

Aditya Mahajan
Optimal decentralized stochastic control
Interconnected Power Systems

Challenges

- How to coordinate?
- When, what, and how to communicate?
Sensor and Surveillance Networks
Sensor and Surveillance Networks

Limited resources
Sensor and Surveillance Networks

Limited resources    Noisy observations
Sensor and Surveillance Networks

Limited resources  Noisy observations
Communication
Sensor and Surveillance Networks

Limited resources  Noisy observations
Communication

Challenges
- Real-time communication
- Scheduling measurements and communication
- Detect node failures

Aditya Mahajan  Optimal decentralized stochastic control
Networked Control Systems

Aditya Mahajan

Optimal decentralized stochastic control
Networked Control Systems
Networked Control Systems

Aditya Mahajan
Optimal decentralized stochastic control
Networked Control Systems

Challenges

Control and communication over networks
(internet ⇒ delay, wireless ⇒ losses)
Networked Control Systems

Challenges

- Control and communication over networks
  (internet $\Rightarrow$ delay, wireless $\Rightarrow$ losses)

- Distributed estimation
Networked Control Systems

Challenges

- Control and communication over networks
  (internet $\Rightarrow$ delay, wireless $\Rightarrow$ losses)

- Distributed estimation

- Distributed learning
Salient features in decentralized decision making

Multiple decision makers

Decisions made by multiple controllers in a stochastic environment
Salient features in decentralized decision making

Multiple decision makers
Decisions made by multiple controllers in a stochastic environment

Coordination issues
All controllers must coordinate to achieve a system-wide objective
Salient features in decentralized decision making

Multiple decision makers
Decisions made by multiple controllers in a stochastic environment

Coordination issues
All controllers must coordinate to achieve a system-wide objective

Communication issues
Controllers can communicate either directly or indirectly
Salient features in decentralized decision making

Multiple decision makers
Decisions made by multiple controllers in a stochastic environment

Coordination issues
All controllers must coordinate to achieve a system-wide objective

Communication issues
Controllers can communicate either directly or indirectly

Robustness
System model may not be completely known
Outline of this talk

Decentralized stochastic control
  Classification and examples

Solution approaches
  A common information based approach

Delayed sharing information structure
  Structure of optimal strategies and dynamic programming decomposition

Concluding remarks
  Generalizations and Connection to other results
Outline of this talk

Decentralized stochastic control
   Classification and examples

Solution approaches
   A common information based approach

Delayed sharing information structure
   Structure of optimal strategies and dynamic programming decomposition

Concluding remarks
   Generalizations and Connection to other results
Classification of decentralized systems

Controllers/agents are coupled in two ways:

1. Coupling due to cost/utility
2. Coupling due to dynamics
Classification of decentralized systems

Controllers/agents are coupled in two ways:

1. Coupling due to cost/utility
2. Coupling due to dynamics

Decentralized systems may be classified according to:
Classification of decentralized systems

Controllers/agents are coupled in two ways:

1. Coupling due to cost/utility
2. Coupling due to dynamics

Decentralized systems may be classified according to:

1. Objective
   Team vs Games
Classification of decentralized systems

Controllers/agents are coupled in two ways:

1. Coupling due to cost/utility
2. Coupling due to dynamics

Decentralized systems may be classified according to:

1. Objective
   - Team vs Games
2. Dynamics
   - Static vs Dynamic
Classification of decentralized systems

Controllers/agents are coupled in two ways:

1. Coupling due to cost/utility
2. Coupling due to dynamics

Decentralized systems may be classified according to:

1. Objective
   - Team vs Games
2. Dynamics
   - Static vs Dynamic

This talk will focus on Dynamic Teams
Classification of decentralized systems

Controllers/agents are coupled in two ways:

1. Coupling due to cost/utility
2. Coupling due to dynamics

Decentralized systems may be classified according to:

1. Objective
   Team vs Games
2. Dynamics
   Static vs Dynamic

This talk will focus on Dynamic Teams

- Studied in economics and systems and control since the mid 50s.
- Unlike games, agents have no incentive to cheat.
- Instead of equilibrium, we seek globally optimal strategies.
Why is decentralized stochastic control difficult?
An example of centralized static optimization

\[ P = [ \bullet \bullet \bullet \bullet ] \]

\[ \begin{array}{cccc}
\omega_1 & \omega_2 & \omega_3 & \omega_4 \\
\end{array} \]
An example of centralized static optimization

\[ P = \begin{bmatrix} \bullet & \bullet & \bullet & \bullet & \bullet \end{bmatrix} \]

\[ x = \begin{array}{ccccc} \omega_1 & \omega_2 & \omega_3 & \omega_4 \\ 1 & 1 & 2 & 2 \end{array} \]
An example of centralized static optimization

\[
P = \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix}
\]

\[
  \begin{array}{c|c|c|c}
  \omega_1 & \omega_2 & \omega_3 & \omega_4 \\
  \hline
  1 & 1 & 2 & 2 \\
  \end{array}
\]

\[
x = (x_1, x_2, x_3, x_4) \in \{1, 2, 3\}
\]

\[
u = g(x) \in \{1, 2, 3\}
\]
An example of centralized static optimization

\[ P = \begin{bmatrix} \bullet & \bullet & \bullet & \bullet & \bullet \end{bmatrix} \]

\[ x = \begin{array}{cccc} \omega_1 & \omega_2 & \omega_3 & \omega_4 \\ 1 & 1 & 2 & 2 \end{array} \]

\[ u = g(x) \in \{1, 2, 3\} \]

\[ c(\omega, u) \]

\[ c(\omega, u) = \begin{array}{cccc} \omega_1 & \omega_2 & \omega_3 & \omega_4 \\ u = 1 & \bullet & \bullet & \bullet & \bullet \\ u = 2 & \bullet & \bullet & \bullet & \bullet \\ u = 3 & \bullet & \bullet & \bullet & \bullet \end{array} \]

\[ J(g) = \mathbb{E}^g[c(\omega, u)] \]
An example of centralized static optimization

\[ P = \begin{bmatrix} \bullet & \bullet & \bullet & \bullet & \bullet \end{bmatrix} \]

\[ x = \begin{array}{cccc}
\omega_1 & \omega_2 & \omega_3 & \omega_4 \\
1 & 1 & 2 & 2 
\end{array} \]

\[ u = g(x) \in \{1, 2, 3\} \]

\[ c(\omega, u) \]

\[ J(g) = \mathbb{E}^g [c(\omega, u)] \]

Brute force search \[ \min_g J(g), \quad |g| = |\mathcal{U}|^{|\mathcal{X}|} = 9 \text{ possibilities.} \]
An example of centralized static optimization

\[ P = \begin{bmatrix} \bullet & \bullet & \bullet & \bullet & \bullet \end{bmatrix} \]

\[ \begin{array}{cccc}
\omega_1 & \omega_2 & \omega_3 & \omega_4 \\
1 & 1 & 2 & 2 \\
\end{array} \]

\[ x = \begin{bmatrix} \omega_1 & \omega_2 & \omega_3 & \omega_4 \end{bmatrix} \]

\[ u = g(x) \in \{1, 2, 3\} \]

\[ c(\omega, u) \]

\[ \begin{array}{cccc}
\omega_1 & \omega_2 & \omega_3 & \omega_4 \\
\bullet & \bullet & \bullet & \bullet \\
\bullet & \bullet & \bullet & \bullet \\
\bullet & \bullet & \bullet & \bullet \\
\end{array} \]

\[ J(g) = \mathbb{E}^g [c(\omega, u)] \]

**Brute force search**

\[ \min_{g} J(g), \quad |g| = |U|^{|X|} = 9 \text{ possibilities.} \]

**Systematic search**

\[ u_1 = g(1) \quad u_2 = g(2) \]

\[ \min_{u_1} \mathbb{E}[c(\omega, u_1) | x = 1] \quad \min_{u_2} \mathbb{E}[c(\omega, u_2) | x = 2] \]

Aditya Mahajan

Optimal decentralized stochastic control
An example of centralized static optimization

\[ P = \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix} \]

\[
\begin{array}{ccc}
\omega_1 & \omega_2 & \omega_3 & \omega_4 \\
1 & 1 & 2 & 2 \\
\end{array}
\]

\[ c(\omega, u) \]

\[
\begin{array}{cccc}
\omega_1 & \omega_2 & \omega_3 & \omega_4 \\
u = 1 & \cdot & \cdot & \cdot & \cdot \\
u = 2 & \cdot & \cdot & \cdot & \cdot \\
u = 3 & \cdot & \cdot & \cdot & \cdot \\
\end{array}
\]

\[ J(g) = \mathbb{E}^g[c(\omega, u)] \text{ (functional opt.)} \]

**Brute force search** \[ \min_{u} J(g), \quad |g| = |\mathcal{U}|^{|\mathcal{X}|} = 9 \text{ possibilities.} \]

**Systematic search** \[ 3 + 3 = 6 \text{ possibilities} \text{ (parametric opt.)} \]

\[ u_1 = g(1) \]

\[ u_2 = g(2) \]

\[ \min_{u_1} \mathbb{E}[c(\omega, u_1) \mid x = 1] \]

\[ \min_{u_2} \mathbb{E}[c(\omega, u_2) \mid x = 2] \]
An example of decentralized static optimization

\[ P = \begin{bmatrix} \cdot & \cdot & \cdot & \cdot \end{bmatrix} \]

| \( \omega_1 \) | \( \omega_2 \) | \( \omega_3 \) | \( \omega_4 \) |
An example of decentralized static optimization

\[ P = [ \bullet \bullet \bullet \bullet ] \]

\[
\begin{array}{cccc}
\omega_1 & \omega_2 & \omega_3 & \omega_4 \\
x = & 1 & 1 & 2 & 2 \\
y = & 2 & 1 & 1 & 2 \\
\end{array}
\]
An example of decentralized static optimization

\[ P = \begin{bmatrix} \bullet & \bullet & \bullet & \bullet \end{bmatrix} \]

\[
\begin{array}{c|cccc}
\omega_1 & \omega_2 & \omega_3 & \omega_4 \\
\hline
x = & 1 & 1 & 2 & 2 \\
y = & 2 & 1 & 1 & 2 \\
\end{array}
\]

\( u = g(x) \in \{1, 2, 3\} \quad v = h(y) \in \{1, 2\} \)
An example of decentralized static optimization

\[ P = [ \begin{array}{ccccc} \cdot & \cdot & \cdot & \cdot & \cdot \end{array} ] \]

<table>
<thead>
<tr>
<th></th>
<th>( \omega_1 )</th>
<th>( \omega_2 )</th>
<th>( \omega_3 )</th>
<th>( \omega_4 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( x = )</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>( y = )</td>
<td>2</td>
<td>1</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

\[ u = g(x) \in \{1, 2, 3\} \quad v = h(y) \in \{1, 2\} \]

\[ c(\omega, u, v) \]

\[ J(g, h) = \mathbb{E}^{g,h}[c(\omega, u, v)] \]
An example of decentralized static optimization

\[ P = [ \bullet \bullet \bullet \bullet \bullet ] \]

\[
\begin{array}{c|c|c|c|c}
& \omega_1 & \omega_2 & \omega_3 & \omega_4 \\
\hline
x &= 1 & 1 & 2 & 2 \\
y &= 2 & 1 & 1 & 2 \\
u &= g(x) \in \{1, 2, 3\} & v &= h(y) \in \{1, 2\} \\
\end{array}
\]

\[ c(\omega, u, v) \]

\[
\begin{array}{c|c|c|c|c}
\omega_1 & \omega_2 & \omega_3 & \omega_4 \\
\hline
u &= 1 & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\
u &= 2 & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\
u &= 3 & \bullet & \bullet & \bullet & \bullet & \bullet & \bullet \\
v &= 1 & 2 & 1 & 2 & 1 & 2 & 2 \\
\end{array}
\]

\[ J(g, h) = \mathbb{E}^{g,h}[c(\omega, u, v)] \]

\[ \min_{g,h} J(g, h), \quad |g| = |U|^{\lvert x \rvert}, \quad |h| = |V|^{\lvert y \rvert}, \]

\[ 9 \times 4 = 36 \text{ possibilities.} \]
An example of decentralized static optimization

\[ P = [ \cdot \cdot \cdot \cdot ] \]

\[
\begin{array}{cccc}
\omega_1 & \omega_2 & \omega_3 & \omega_4 \\
x = & 1 & 1 & 2 & 2 \\
y = & 2 & 1 & 1 & 2 \\
\end{array}
\]

\[ u = g(x) \in \{1, 2, 3\} \quad v = h(y) \in \{1, 2\} \]

\[ c(\omega, u, v) \]

\[ J(g, h) = \mathbb{E}^{g,h}[c(\omega, u, v)] \]

Brute force search \[ \min_{g, h} J(g), \quad |g| = |\mathcal{U}|^{\|x\|}, \quad |h| = |\mathcal{V}|^{\|y\|}, \]

\[ 9 \times 4 = 36 \text{ possibilities.} \]

For one controller/agent to choose an optimal action, it must second guess the other controller’s/agent’s policy.

Aditya Mahajan
Optimal decentralized stochastic control
An example of decentralized static optimization

\[ P = \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix} \]

\[
\begin{array}{c|cccc}
\omega & \omega_1 & \omega_2 & \omega_3 & \omega_4 \\
\hline
x & 1 & 1 & 2 & 2 \\
y & 2 & 1 & 1 & 2 \\
\end{array}
\]

\[ u = g(x) \in \{1, 2, 3\} \quad v = h(y) \in \{1, 2\} \]

\[ c(\omega, u, v) \]

\[ J(g, h) = \mathbb{E}^{g,h}[c(\omega, u, v)] \]

Orthogonal search

1. Suppose \( h \) is fixed: \( \min_{u_i} \mathbb{E}^h[c(\omega, u_i, v) \mid x = i], \quad i = 1, 2, 3. \)

2. Suppose \( g \) is fixed: \( \min_{v_j} \mathbb{E}^g[c(\omega, u, v_j) \mid y = j], \quad j = 1, 2. \)
An example of decentralized static optimization

\[ P = \begin{bmatrix} \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix} \]

\[ x = \begin{array}{ccccc} \omega_1 & \omega_2 & \omega_3 & \omega_4 \\ 1 & 1 & 2 & 2 \end{array} \]

\[ y = \begin{array}{ccccc} 2 & 1 & 1 & 2 \end{array} \]

\[ u = g(x) \in \{1, 2, 3\} \quad v = h(y) \in \{1, 2\} \]

\[ c(\omega, u, v) \]

\[ J(g, h) = \mathbb{E}^{g,h}[c(\omega, u, v)] \]

**Orthogonal search yields person-by-person opt strategy**

1. Suppose \( h \) is fixed: \( \min_{u_i} \mathbb{E}^h[c(\omega, u_i, v) \mid x = i], \quad i = 1, 2, 3. \)

2. Suppose \( g \) is fixed: \( \min_{v_j} \mathbb{E}^g[c(\omega, u, v_j) \mid y = j], \quad j = 1, 2. \)
To find globally optimal strategies, in general, we cannot do better than brute force search.
An example of centralized multi-stage optimization

<table>
<thead>
<tr>
<th>$\omega_1$</th>
<th>$\omega_2$</th>
<th>$\omega_3$</th>
<th>$\omega_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\omega_5$</td>
<td>$\omega_6$</td>
<td>$\omega_7$</td>
<td>$\omega_8$</td>
</tr>
</tbody>
</table>
An example of centralized multi-stage optimization

<table>
<thead>
<tr>
<th>$\omega_1$</th>
<th>$\omega_2$</th>
<th>$\omega_3$</th>
<th>$\omega_4$</th>
<th>$y_1 = 1$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\omega_5$</td>
<td>$\omega_6$</td>
<td>$\omega_7$</td>
<td>$\omega_8$</td>
<td>$y_1 = 2$</td>
</tr>
</tbody>
</table>
An example of centralized multi-stage optimization

<table>
<thead>
<tr>
<th>$\omega_1$</th>
<th>$\omega_2$</th>
<th>$\omega_3$</th>
<th>$\omega_4$</th>
<th>$y_1 = 1$</th>
<th>$u_1 = g_1(y_1) \in {1, 2}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\omega_5$</td>
<td>$\omega_6$</td>
<td>$\omega_7$</td>
<td>$\omega_8$</td>
<td>$y_1 = 2$</td>
<td></td>
</tr>
</tbody>
</table>
An example of centralized multi-stage optimization

\[ y_1 = 1 \quad u_1 = g_1(y_1) \in \{1, 2\} \]

\[ y_1 = 2 \]

<table>
<thead>
<tr>
<th>\omega_1</th>
<th>\omega_2</th>
<th>\omega_3</th>
<th>\omega_4</th>
</tr>
</thead>
<tbody>
<tr>
<td>\omega_5</td>
<td>\omega_6</td>
<td>\omega_7</td>
<td>\omega_8</td>
</tr>
</tbody>
</table>

| \begin{align*}
u_1 &= 1 \quad \Rightarrow \quad y_2 &= 1 & 1 & 2 & 2 \\
u_1 &= 2 \quad \Rightarrow \quad y_2 &= 1 & 2 & 2 & 1 \\
u_1 &= 1 \quad \Rightarrow \quad y_2 &= 1 & 1 & 2 & 2 \\
u_1 &= 2 \quad \Rightarrow \quad y_2 &= 1 & 2 & 2 & 1 \\
\end{align*} |
An example of centralized multi-stage optimization

<table>
<thead>
<tr>
<th>ω₁</th>
<th>ω₂</th>
<th>ω₃</th>
<th>ω₄</th>
<th>y₁ = 1</th>
<th>u₁ = g₁(y₁) ∈ {1, 2}</th>
</tr>
</thead>
<tbody>
<tr>
<td>ω₅</td>
<td>ω₆</td>
<td>ω₇</td>
<td>ω₈</td>
<td>y₁ = 2</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>u₁ = 1 ⇒ y₂ =</th>
<th>1</th>
<th>1</th>
<th>2</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>u₁ = 2 ⇒ y₂ =</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>u₁ = 1 ⇒ y₂ =</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>u₁ = 2 ⇒ y₂ =</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

u₂ = g₂(y₁, y₂, u₁) ∈ {1, 2}
An example of centralized multi-stage optimization

\[ \begin{array}{cccc|c|c}
\omega_1 & \omega_2 & \omega_3 & \omega_4 & y_1 = 1 \\
\omega_5 & \omega_6 & \omega_7 & \omega_8 & y_1 = 2 \\
\hline
u_1=1 & y_2= & 1 & 1 & 2 & 2 \\
u_1=1 & y_2= & 1 & 1 & 2 & 2 \\
u_1=2 & y_2= & 1 & 2 & 2 & 1 \\
u_1=2 & y_2= & 1 & 2 & 2 & 1 \\
\end{array} \]

\[ u_1 = g_1(y_1) \in \{1, 2\} \]
\[ u_2 = g_2(y_1, y_2, u_1) \in \{1, 2\} \]
\[ c_1(\omega, u_1) + c_2(\omega, u_2) \]

\[ J(g_1, g_2) = \mathbb{E}^{g_1,g_2}[c_1(\omega, u_1) + c_2(\omega, u_2)] \]
An example of centralized multi-stage optimization

\[
\begin{array}{cccc}
\omega_1 & \omega_2 & \omega_3 & \omega_4 \\
\omega_5 & \omega_6 & \omega_7 & \omega_8 \\
\end{array}
\]

\(y_1 = 1 \quad u_1 = g_1(y_1) \in \{1, 2\}\)

\(d_1 = \{y_1\}\)

\(y_1 = 2 \quad d_2 = \{y_1\}\)

\[
\begin{array}{cccc}
1 & 1 & 2 & 2 \\
1 & 1 & 2 & 2 \\
1 & 2 & 2 & 1 \\
1 & 2 & 2 & 1 \\
\end{array}
\]

\(u_2 = g_2(y_1, y_2, u_1) \in \{1, 2\}\)

\(d_2 = \{y_1, y_2, u_1\}\)

\[c_1(\omega, u_1) + c_2(\omega, u_2)\]

\[
J(g_1, g_2) = \mathbb{E}^{g_1, g_2}[c_1(\omega, u_1) + c_2(\omega, u_2)]
\]
An example of centralized multi-stage optimization

<table>
<thead>
<tr>
<th>$\omega_1$</th>
<th>$\omega_2$</th>
<th>$\omega_3$</th>
<th>$\omega_4$</th>
<th>$y_1 = 1$</th>
<th>$u_1 = g_1(y_1) \in {1, 2}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\omega_5$</td>
<td>$\omega_6$</td>
<td>$\omega_7$</td>
<td>$\omega_8$</td>
<td>$y_1 = 2$</td>
<td>$d_1 = {y_1}$</td>
</tr>
</tbody>
</table>

$u_1=1 \Rightarrow y_2=\begin{array}{cccc}1 & 1 & 2 & 2 \\1 & 1 & 2 & 2 \end{array}$

$u_2 = g_2(y_1, y_2, u_1) \in \{1, 2\}$

$u_1=1 \Rightarrow y_2=\begin{array}{cccc}1 & 2 & 2 & 1 \\1 & 2 & 2 & 1 \end{array}$

$u_1=2 \Rightarrow y_2=\begin{array}{cccc}1 & 2 & 2 & 1 \\1 & 2 & 2 & 1 \end{array}$

$u_2 = g_2(y_1, y_2, u_1) \in \{1, 2\}$

$c_1(\omega, u_1) + c_2(\omega, u_2)$

$J(g_1, g_2) = \mathbb{E}^{g_1,g_2}[c_1(\omega, u_1) + c_2(\omega, u_2)]$

Critical Assumption: Centralized information

$d_1 \subseteq d_2$
Solution approach for centralized multi-stage optimization

Brute force search: \[ \min_{g_1, g_2} J(g_1, g_2). \]

\[ |g_1| = |U_1| |y_1|, \quad |g_2| = |U_2| |y_1| \times |y_2| \times |u_1|. \]

\[ 2^2 \times 2^8 = 1024 \text{ possibilities.} \]
Solution approach for centralized multi-stage optimization

Brute force search

\[
\min_{g_1, g_2} J(g_1, g_2).
\]

\[|g_1| = |U_1| |y_1|, \quad |g_2| = |U_2| |y_1| \times |y_2| \times |u_1|. \quad 2^2 \times 2^8 = 1024 possibilities.
\]

Dynamic programming decomposition

\[
V_2(d_2) = \min_{u_2} \mathbb{E}[c_2(\omega, u_2) \mid d_2, u_2]
\]

\[
V_1(d_1) = \min_{u_1} \mathbb{E}[c_1(\omega, u_1) + V_2(d_2) \mid d_1, u_1]
\]
Solution approach for centralized multi-stage optimization

Brute force search

\[ \min_{g_1, g_2} J(g_1, g_2). \]  (functional opt.)

\[ |g_1| = |U| |y_1|, \quad |g_2| = |U| |y_1| \times |y_2| \times |u| . \quad 2^2 \times 2^8 = 1024 possibilities. \]

Dynamic programming decomposition  (parametric opt.)

\[ V_2(d_2) = \min_{u_2} E[c_2(\omega, u_2) \mid d_2, u_2] \]

\[ V_1(d_1) = \min_{u_1} E[c_1(\omega, u_1) + V_2(d_2) \mid d_1, u_1] \]
Solution approach for centralized multi-stage optimization

Brute force search

\[ \min_{g_1, g_2} J(g_1, g_2). \]  
\[ |g_1| = |\mathcal{U}_1||y_1|, \quad |g_2| = |\mathcal{U}_2||y_1||y_2||u_1|. \]  
\[ 2^2 \times 2^8 = 1024 \text{ possibilities.} \]

Dynamic programming decomposition

\[ V_2(d_2) = \min_{u_2} \mathbb{E}[c_2(\omega, u_2) | d_2, u_2] \]
\[ V_1(d_1) = \min_{u_1} \mathbb{E}[c_1(\omega, u_1) + V_2(d_2) | d_1, u_1] \]

- Step 1 works because \( \mathbb{P}(\omega | d_2) \) does not depend on \( g_1 \).
- Step 2 works because \( \mathbb{P}(d_2 | d_1, u_1) \) does not depend on \( g_1 \).
Solution approach for centralized multi-stage optimization

Brute force search

\[
\min_{g_1,g_2} J(g_1, g_2).
\]

\[
|g_1| = |U_1| |y_1|, \quad |g_2| = |U_2| |y_1| \times |y_2| \times |U_1|.
\]

\[2^2 \times 2^8 = 1024 \text{ possibilities.}\]

Dynamic programming decomposition

\[
V_2(d_2) = \min_{u_2} \mathbb{E}[c_2(\omega, u_2) \mid d_2, u_2]
\]

\[
V_1(d_1) = \min_{u_1} \mathbb{E}[c_1(\omega, u_1) + V_2(d_2) \mid d_1, u_1]
\]

- Step 1 works because \(\mathbb{P}(\omega \mid d_2)\) does not depend on \(g_1\).
- Step 2 works because \(\mathbb{P}(d_2 \mid d_1, u_1)\) does not depend on \(g_1\).
- Both steps work because \(d_1 \subseteq d_2\)
An example of decentralized multi-stage optimization

<table>
<thead>
<tr>
<th>$\omega_1$</th>
<th>$\omega_2$</th>
<th>$\omega_3$</th>
<th>$\omega_4$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\omega_5$</td>
<td>$\omega_6$</td>
<td>$\omega_7$</td>
<td>$\omega_8$</td>
</tr>
</tbody>
</table>

$y_1 = 1 \quad u_1 = g_1(y_1) \in \{1, 2\}$

$y_1 = 2 \quad d_1 = \{y_1\}$

$u_1 = 1 \Rightarrow y_2 = \begin{bmatrix} 1 & 1 & 2 & 2 \end{bmatrix}$

$u_2 = g_2(y_2) \in \{1, 2\}$

$\quad d_2 = \{y_2\}$

$u_1 = 1 \Rightarrow y_2 = \begin{bmatrix} 1 & 1 & 2 & 2 \end{bmatrix}$

$u_1 = 2 \Rightarrow y_2 = \begin{bmatrix} 1 & 2 & 2 & 1 \end{bmatrix}$

$u_1 = 2 \Rightarrow y_2 = \begin{bmatrix} 1 & 2 & 2 & 1 \end{bmatrix}$

$J(g_1, g_2) = \mathbb{E}^{g_1, g_2}[c_1(\omega, u_1) + c_2(\omega, u_2)]$
An example of decentralized multi-stage optimization

<table>
<thead>
<tr>
<th>$\omega_1$</th>
<th>$\omega_2$</th>
<th>$\omega_3$</th>
<th>$\omega_4$</th>
<th>$\omega_5$</th>
<th>$\omega_6$</th>
<th>$\omega_7$</th>
<th>$\omega_8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$y_1 = 1$</td>
<td></td>
<td></td>
<td></td>
<td>$y_1 = 2$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>$u_1 = g_1(y_1) \in {1, 2}$</td>
<td></td>
<td></td>
<td>$d_1 = {y_1}$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$u_1 = 1$</td>
<td>$u_2 = g_2(y_2) \in {1, 2}$</td>
<td></td>
<td></td>
<td>$d_2 = {y_2}$</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$u_1 = 2$</td>
<td>$c_1(\omega, u_1) + c_2(\omega, u_2)$</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Critical Assumption: Decentralized information

Can we do better than brute force search?
Usual Dynamic programming does not work?

\[ V_2(d_2) \overset{?}{=} \min_{u_2} \mathbb{E}^{g_1} [c_2(\omega, u_2) \mid d_2, u_2] \]

\[ V_1(d_1) \overset{?}{=} \min_{u_1} \mathbb{E}^{g_1} [c_1(\omega, u_1) + V_2(d_2) \mid d_1, u_1] \]
Usual Dynamic programming does not work?

\[ V_2(d_2) \overset{?}{=} \min_{u_2} \mathbb{E}^{g_1}[c_2(\omega, u_2) \mid d_2, u_2] \]

\[ V_1(d_1) \overset{?}{=} \min_{u_1} \mathbb{E}^{g_1}[c_1(\omega, u_1) + V_2(d_2) \mid d_1, u_1] \]

A sequential decomposition is possible (Witsenhausen, 1973)

Define \( \pi_t = \mathbb{P}(\omega \mid g_{1:t-1}) \).

\[ V_t(\pi_t) = \min_{g_t} \mathbb{E}^{g_t}[c_t(\omega, u_t) + V_{t+1}(\pi_{t+1}) \mid \pi_t] \]

But, the worst case complexity remains the same.
Can we obtain a systematic approach to find optimal strategies that does better than brute force search?
Outline of this talk

Decentralized stochastic control
   Classification and examples

Solution approaches
   A common information based approach

Delayed sharing information structure
   Structure of optimal strategies and dynamic programming decomposition

Concluding remarks
   Generalizations and Connection to other results
The intrinsic model for controlled dynamical systems

\((\Omega, \mathcal{F}, P)\)

Dynamical Model
The intrinsic model for controlled dynamical systems

\((\Omega, \mathcal{F}, P)\)

Dynamical system

Controller

Dynamical Model
The intrinsic model for controlled dynamical systems

Dynamical Model

$(\Omega, \mathcal{F}, P)$

$\omega$

Dynamical system

$u_t$

Controller

$y_t$
The intrinsic model for controlled dynamical systems

Dynamical Model
The intrinsic model for controlled dynamical systems

\((\Omega, \mathcal{F}, P)\)

Dynamical Model

Intrinsic Model

\((\Omega, \mathcal{F}, P)\)

Dynamical system

Controller

\(\omega\)

\(c_t\)

\(u_t\)

\(y_t\)
The intrinsic model for controlled dynamical systems

\((\Omega, \mathcal{F}, P)\)

\(\omega\)

\(c_t\)

Dynamical system

\(u_t\)

\(y_t\)

Controller

\((\Omega, \mathcal{F}, P)\)

\(C_1\)

\(C_2\)

\(C_{t-1}\)

\(C_t\)

\(C_{t+1}\)

\(C_{t+2}\)

Dynamical Model

Intrinsic Model
The intrinsic model for controlled dynamical systems

\[(\Omega, \mathcal{F}, P)\]

Dynamical model

\[
(\Omega, \mathcal{F}, P)
\]

Intrinsic model

\[
C_1 \quad C_2 \quad C_{t-1}
\]

\[
C_t \quad C_{t+1} \quad C_{t+2}
\]

Dynamical Model

Intrinsic Model

Aditya Mahajan

Optimal decentralized stochastic control
The intrinsic model for controlled dynamical systems

\[(\Omega, \mathcal{F}, P)\]

Dynamical Model

Intrinsic Model

\(C_1\)
\(C_2\)
\(C_{t-1}\)

\(C_{t+1}\)
\(C_{t+2}\)
Information state and a general solution approach for centralized stochastic systems

In a centralized system, i.e., \( d_t \subseteq d_{t+1} \), a function \( \pi_t = \pi_t(d_t) \) is an information state if it satisfies:

1. The controller Markov property

\[
\mathbb{E}^g[\pi_{t+1} \mid d_t, u_t] = \mathbb{E}[\pi_{t+1} \mid \pi_t, u_t]
\]

2. The expected cost property

\[
\mathbb{E}^g[c_t \mid d_t, u_t] = \mathbb{E}[c_t \mid \pi_t, u_t]
\]
Information state and a general solution approach for centralized stochastic systems

In a centralized system, i.e., \( d_t \subseteq d_{t+1} \), a function \( \pi_t = \pi_t(d_t) \) is an **information state** if it satisfies:

1. The controller Markov property

\[
\mathbb{E}^g[\pi_{t+1} | d_t, u_t] = \mathbb{E}[\pi_{t+1} | \pi_t, u_t]
\]

2. The expected cost property

\[
\mathbb{E}^g[c_t | d_t, u_t] = \mathbb{E}[c_t | \pi_t, u_t]
\]

- Info-state in MDPs: current state
- Info-state in POMDPs:
  - posterior belief on current state
Information state and a general solution approach for centralized stochastic systems

In a centralized system, i.e., \( d_t \subseteq d_{t+1} \), a function \( \pi_t = \pi_t(d_t) \) is an **information state** if it satisfies:

1. The controller Markov property

\[
E^g[\pi_{t+1} \mid d_t, u_t] = E[\pi_{t+1} \mid \pi_t, u_t]
\]

2. The expected cost property

\[
E^g[c_t \mid d_t, u_t] = E[c_t \mid \pi_t, u_t]
\]

**Structure of optimal strategy**

Restricting attention to control strategies of the form

\[
u_t = g_t(\pi_t)
\]

is without any loss.

- Info-state in MDPs: current state
- Info-state in POMDPs: posterior belief on current state
Information state and a general solution approach for centralized stochastic systems

In a centralized system, i.e., $d_t \subseteq d_{t+1}$, a function $\pi_t = \pi_t(d_t)$ is an information state if it satisfies:

1. The controller Markov property

$$\mathbb{E}^g[\pi_{t+1} \mid d_t, u_t] = \mathbb{E}[\pi_{t+1} \mid \pi_t, u_t]$$

2. The expected cost property

$$\mathbb{E}^g[c_t \mid d_t, u_t] = \mathbb{E}[c_t \mid \pi_t, u_t]$$

- Info-state in MDPs: current state
- Info-state in POMDPs: posterior belief on current state

Structure of optimal strategy

Restricting attention to control strategies of the form

$$u_t = g_t(\pi_t)$$

is without any loss.

Search of optimal strategy

An optimal strategy of the form above is given by the solution of the following dynamic program:

$$V_t(\pi_t) = \min_{u_t} \mathbb{E}[c_t + V_{t+1}(\pi_{t+1}) \mid \pi_t, u_t]$$
How do we define an information state for a decentralized system?
Common Knowledge (Aumann, 1976)

\((\Omega, \mathcal{F}, P)\)

\(X(\omega)\)

\(Y(\omega)\)
Common Knowledge (Aumann, 1976)

\[(\Omega, \mathcal{F}, P)\]

\[
X(\omega) \quad \sigma(X) \cap \sigma(Y) \\
Y(\omega)
\]
Common Knowledge (Aumann, 1976)

\((\Omega, \mathcal{F}, P)\)

\[X(\omega)\]

\[Y(\omega)\]

\(\sigma(X) \cap \sigma(Y)\)

\[
\begin{array}{cccc}
\omega_5 & \omega_6 & \omega_7 & \omega_8 \\
\omega_1 & \omega_2 & \omega_3 & \omega_4 \\
\end{array}
\]

\[
\begin{array}{cccc}
\omega_5 & \omega_6 & \omega_7 & \omega_8 \\
\omega_1 & \omega_2 & \omega_3 & \omega_4 \\
\end{array}
\]
Common Knowledge (Aumann, 1976)

\[ (\Omega, \mathcal{F}, P) \]

\[ X(\omega) \]

\[ Y(\omega) \]

\[ \sigma(X) \cap \sigma(Y) \]

\begin{array}{cccc}
\omega_5 & \omega_6 & \omega_7 & \omega_8 \\
\omega_1 & \omega_2 & \omega_3 & \omega_4 \\
\end{array}
Exploiting common knowledge to simplify decentralized static optimization

\[ u = g(x), \quad v = h(y) \]

\[ J(g, h) = \mathbb{E}^{g,h}[c(\omega, u, v)] \]
Exploiting common knowledge to simplify decentralized static optimization

\[ u = g(x), \quad v = h(y) \]

\[ J(g, h) = \mathbb{E}^{g,h}[c(\omega, u, v)] \]

Let \( k \) denote the common knowledge between \( x \) and \( y \). Write:

\[ x \equiv (k, p), \quad y \equiv (k, q), \]

\[ u = \tilde{g}(k, p). \quad v = \tilde{h}(k, q). \]
Exploiting common knowledge to simplify decentralized static optimization

Let $k$ denote the common knowledge between $x$ and $y$. Write:

\[
\tilde{g} : (k, p) \mapsto u, \quad \tilde{g} : k \mapsto (p \mapsto u) \overset{\gamma}{\mapsto} v
\]

\[
u = g(x), \quad v = h(y)
\]

\[
J(g, h) = \mathbb{E}^{g,h}[c(\omega, u, v)]
\]
Exploiting common knowledge to simplify decentralized static optimization

Let $k$ denote the common knowledge between $x$ and $y$. Write:

$$u = g(x), \quad v = h(y)$$

$$J(g, h) = \mathbb{E}^{g,h}[c(\omega, u, v)]$$

Let $\tilde{g} : (k, p) \mapsto u$, $\tilde{g} : k \mapsto (p \mapsto u)$

Let $\gamma(\cdot) = \tilde{g}(k, \cdot)$ and $\eta(\cdot) = \tilde{h}(k, \cdot)$
Exploiting common knowledge to simplify decentralized static optimization

Let $k$ denote the common knowledge between $x$ and $y$. Write:

$$u = g(x), \quad v = h(y)$$
$$J(g, h) = \mathbb{E}^{g,h}[c(\omega, u, v)]$$

A common knowledge based solution

$$\min \mathbb{E}^{\gamma,\eta}[c(\omega, u, v)|k]$$
Exploiting common knowledge to simplify decentralized static optimization

Let $k$ denote the common knowledge between $x$ and $y$. Write:

$$u = g(x), \quad v = h(y)$$

$$J(g, h) = \mathbb{E}^{g,h}[c(\omega, u, v)]$$

A common knowledge based solution (functional opt. over smaller space)

$$\min_{\gamma, \eta} \mathbb{E}^{\gamma, \eta}[c(\omega, u, v)|k]$$
Exploiting common knowledge to simplify decentralized static optimization

Let \( k \) denote the common knowledge between \( x \) and \( y \). Write:

\[
\tilde{g} : (k, p) \mapsto u, \quad \tilde{g} : k \mapsto (p \mapsto u)
\]

\[
x \equiv (k, p), \quad y \equiv (k, q),
\]

\[
u = \tilde{g}(k, p). \quad v = \tilde{h}(k, q).
\]

A common knowledge based solution (functional opt. over smaller space)

\[
\min_{\gamma, \eta} \mathbb{E}^{\gamma, \eta}[c(\omega, u, v)|k]
\]

Brute force: \( 2^4 \times 2^4 \) possibilities. CK-based soln: \( 2 \cdot (2^2 \times 2^2) \) possibilities.
Main idea: Extend CK-based approach to decentralized multi-stage systems.
Main idea: Extend CK-based approach to decentralized multi-stage systems.
A common information based approach for decentralized multi-stage systems

(Nayyar, 2010; Nayyar, Mahajan, Teneketzis, 2011)

Split data at each controller/agent into two parts:

- **Common information**: \( k_t = \bigcap_{s \geq t} d_s \)
- **Private information**: \( p_t = d_t \setminus k_t \)
A common information based approach for decentralized multi-stage systems
(Nayyar, 2010; Nayyar, Mahajan, Teneketzis, 2011)

Split data at each controller/agent into two parts:

- **Common information**: \( k_t = \bigcap_{s \geq t} d_s \)

- **Private information**: \( p_t = d_t \setminus k_t \)

**Objective** Choose \( u_t = g_t(k_t, p_t) \) to minimize

\[
J(g_{1:T}) = \mathbb{E}^{g_{1:T}}[c(\omega, u_{1:T})]
\]
A common information based approach for decentralized multi-stage systems

(Nayyar, 2010; Nayyar, Mahajan, Teneketzis, 2011)

Split data at each controller/agent into two parts:

- **Common information:** $k_t = \bigcap_{s \geq t} d_s \quad k_t \subseteq k_{t+1}$
- **Private information:** $p_t = d_t \setminus k_t$

**Objective**

Choose $u_t = g_t(k_t, p_t)$ to minimize

$$J(g_{1:T}) = \mathbb{E}^{g_{1:T}}[c(\omega, u_{1:T})]$$
A common information based approach for decentralized multi-stage systems
(Nayyar, 2010; Nayyar, Mahajan, Teneketzis, 2011)

Split data at each controller/agent into two parts:

- **Common information:** \( k_t = \bigcap_{s \geq t} d_s \), \( k_t \subseteq k_{t+1} \)
- **Private information:** \( p_t = d_t \setminus k_t \)

**Objective** Choose \( u_t = g_t(k_t, p_t) \) to minimize

\[
J(g_{1:T}) = \mathbb{E}^{g_{1:T}}[c(\omega, u_{1:T})]
\]

**Solution approach**

1. Construct a **coordinated system** (that has classical info-struct.)
2. Show that coordinated system \( \equiv \) original system.
3. Find a solution to coordinated system using centralized stoc. control.
4. Translate the result back to original system
A common information based approach for decentralized multi-stage systems

\[ u_1 \rightarrow C_1 \rightarrow d_1 \]
\[ g_1 \]

\[ \vdots \]
\[ u_t \rightarrow C_t \rightarrow d_t \]
\[ g_t \]

\[ \vdots \]
\[ u_T \rightarrow C_T \rightarrow d_T \]
\[ g_T \]
A common information based approach for decentralized multi-stage systems

\[
\begin{align*}
&\quad C_1 \quad \cdots \quad C_t \quad \cdots \quad C_T \\
&u_1 \quad \cdots \quad u_t \quad \cdots \quad u_T \\
&d_1 \quad \cdots \quad d_t \quad \cdots \quad d_T \\
&\{\cdots, k_t, \cdots\} \\
&\{\cdots, \gamma_t, \cdots\} \\
&\text{Coordinator}
\end{align*}
\]

**Prescription:** \( \gamma_t : p_t \mapsto u_t \), chosen according to

\[
\begin{align*}
\gamma_t &= \psi_t(k_t, \gamma_{1:t-1}) \\
u_t &= \gamma_t(p_t)
\end{align*}
\]
A common information based approach for decentralized multi-stage systems

![Diagram](image)

**Prescription:** \( \gamma_t : p_t \mapsto u_t \), chosen according to

\[
\gamma_t = \psi_t(k_t, \gamma_{1:t-1}) \\
u_t = \gamma_t(p_t)
\]

The two systems are equivalent

\[
g_t(k_t, p_t) = \gamma_t(p_t) \\
\psi_t(k_t, \gamma_{1:t-1})
\]
A common information based approach for decentralized multi-stage systems

Prescription: \( \gamma_t : p_t \mapsto u_t \), chosen according to

\[
\gamma_t = \psi_t(k_t, \gamma_{1:t-1})
\]

\[
u_t = \gamma_t(p_t)
\]

The two systems are equivalent

\[
g_t(k_t, p_t) = \gamma_t(p_t) = \psi_t(k_t, \gamma_{1:t-1})
\]

Coordinated system is centralized

Find information state \( \pi_t \).

Without loss of optimality, choose \( \gamma_t = \psi_t(\pi_t) \)

Write DP in terms of \( \pi_t \):

\[
V_t(\pi_t) = \min_{\gamma_t} \mathbb{E} \left[ c_t(\cdot) + V_{t+1}(\pi_{t+1}) \mid \pi_t, \gamma_t \right]
\]
A common information based approach for decentralized multi-stage systems

Prescription: $\gamma_t : p_t \mapsto u_t$, chosen according to
\[
\gamma_t = \psi_t(k_t, \gamma_{1:t-1})
\]
\[
u_t = \gamma_t(p_t)
\]
The two systems are equivalent
\[
g_t(k_t, p_t) = \gamma_t(p_t)
\]
\[
\psi_t(k_t, \gamma_{1:t-1})
\]

Coordinated system is centralized

Find information state $\pi_t$.

Without loss of optimality, choose $\gamma_t = \psi_t(\pi_t) \equiv u_t = g_t(\pi_t, p_t)$

Write DP in terms of $\pi_t$: $V_t(\pi_t) = \min_{\gamma_t} \mathbb{E}[c_t(\cdot) + V_{t+1}(\pi_{t+1}) \mid \pi_t, \gamma_t]$
Outline of this talk

Decentralized stochastic control
   Classification and examples

Solution approaches
   A common information based approach

Delayed sharing information structure
   Structure of optimal strategies and dynamic programming decomposition

Concluding remarks
   Generalizations and Connection to other results
Delayed sharing information structure

\[
\begin{align*}
\text{Sys} & \rightarrow X_t \leftarrow \text{Obs channel} \\
& \rightarrow Y_t^1 \rightarrow \text{Controller 1} \rightarrow U_t^1 \rightarrow U_1^2 \\
& \rightarrow Y_t^2 \rightarrow \text{Controller 2} \rightarrow U_t^2 \\
\end{align*}
\]
Delayed sharing information structure

\[ X_{t+1} = f(X_t, U_{t}^{1:2}, W_t) \quad Y_t^i = h^i(X_t, N_t^i) \]
Delayed sharing information structure

\[
X_{t+1} = f(X_t, U_t^{1:2}, W_t) \quad Y_t^i = h^i(X_t, N_t^i)
\]

- $n$-step delayed info sharing
- Perfect recall at controller
Delayed sharing information structure

\[ X_{t+1} = f(X_t, U_t^{1:2}, W_t) \quad Y_t^i = h^i(X_t, N_t^i) \]

- \( n \)-step delayed info sharing
- Perfect recall at controller

\[ J(g_{1:T}^{1,2}) = \mathbb{E}^{g_{1:T}^{1,2}}[c(X_t, U_t^{1,2})] \]
(Witsenhausen, 1971):

- Proposed delayed-sharing information structure.
- **Asserted** a structure of optimal control law (without proof).
Literature Overview

(Witsenhausen, 1971):

- Proposed delayed-sharing information structure.
- **Asserted** a structure of optimal control law (without proof).

(Varaiya and Walrand, 1978):

- Proved Witsenhausen’s assertion for $n = 1$.
- Counter-example to **disproved** the assertion for delay $n > 2$. 

Literature Overview

(Witsenhausen, 1971):
- Proposed delayed-sharing information structure.
- **Asserted** a structure of optimal control law (without proof).

(Varaiya and Walrand, 1978):
- Proved Witsenhausen’s assertion for $n = 1$.
- Counter-example to **disproved** the assertion for delay $n > 2$.

The result of one-step delayed sharing used in various applications:
- **Queueing theory**: Kuri and Kumar, 1995
- **Communication networks**: Altman *et. al*, 2009, Grizzle *et. al*, 1982
- **Stochastic games**: Papavassilopoulos, 1982; Chang and Cruz, 1983
- **Economics**: Li and Wu, 1991
Solution based on common information approach

**Common information**

\[ K_t = (Y_{1:t-n}^{1,2}, U_{1:t-n}^{1,2}) \]

**Private information**

\[ P_t^i = (Y_{t-n+1:t}^i, U_{t-n+1:t-1}^i) \]

**Control actions**

\[ U_t^1 = g^1(K_t, P_t^2), \quad U_t^2 = g^2(K_t, P_t^2) \]
Solution based on common information approach

Common information
\[ K_t = (Y_{1:t-n}^{1,2}, U_{1:t-n}^{1,2}) \]

Private information
\[ P_t^i = (Y_{t-n+1:t}^i, U_{t-n+1:t-1}^i) \]

Control actions
\[ U_t^1 = g^1(K_t, P_t^2), \quad U_t^2 = g^2(K_t, P_t^2) \]

Coordinated System

Data observed
\[ K_t \) (increasing with time)\]

Control actions
\[ (\gamma_t^1, \gamma_t^2), \text{ where } \gamma_t^i : P_t^i \mapsto U_t^i \]
Solution based on common information approach

Common information

\[ K_t = (Y_{1:t-n}^{1,2}, U_{1:t-n}^{1,2}) \]

Private information

\[ P_t^i = (Y_{t-n+1:t}^i, U_{t-n+1:t-1}^i) \]

Control actions

\[ U_t^1 = g^1(K_t, P_t^2), \quad U_t^2 = g^2(K_t, P_t^2) \]

Coordinated System

Data observerd

\[ K_t \text{ (increasing with time)} \]

Control actions

\( (\gamma_t^1, \gamma_t^2), \text{ where } \gamma_t^i : P_t^i \mapsto U_t^i \)

Find a solution to the coordinated system and translate it back to the original system.
The coordinated system: state for I/O mapping

\begin{align*}
\gamma_t^1 & \quad P_t^1 \quad U_t^1 \\
\gamma_t^2 & \quad P_t^2 \quad U_t^2 \\
\psi_t & \quad K_t \quad (\gamma_t^1, \gamma_t^2)
\end{align*}
The coordinated system: state for I/O mapping

State for I/O mapping: \((X_t, P_t^1, P_t^2)\)
Information state for coordinated system

The coordinated system is a **centralized** partially observed system.

\[ \text{Info state} = \mathbb{P}(\text{state for I/O mapping} \mid \text{data at controller}) \]
Information state for coordinated system

The coordinated system is a **centralized** partially observed system.

\[
\text{Info state} = \mathbb{P}(\text{state for I/O mapping} \mid \text{data at controller})
\]

\[
\pi_t = \mathbb{P}(X_t, P_t^1, P_t^2 \mid K_t, \gamma_t^1, \gamma_t^2)
\]

**Structural Result** There is no loss of optimality in restricting prescriptions of the form

\[
\gamma_t = \psi_t(\pi_t) \quad \text{and hence,} \quad U_t^i = g_t^i(\pi_t, P_t^i)
\]
Information state for coordinated system

The coordinated system is a **centralized** partially observed system.

Info state = $\mathbb{P}(\text{state for I/O mapping} | \text{data at controller})$

$$\pi_t = \mathbb{P}(X_t, P^1_t, P^2_t | K_t, \gamma^1_t, \gamma^2_t)$$

**Structural Result** There is no loss of optimality in restricting prescriptions of the form

$$\gamma_t = \psi_t(\pi_t) \quad \text{and hence,} \quad U^i_t = g^i_t(\pi_t, P^i_t)$$

**Dynamic Programming decomposition** An optimal coordination strategy is given by the solution to the following dynamic program

$$V_t(\pi_t) = \min_{\gamma^1_t, \gamma^2_t} \mathbb{E}[c(X_t, \gamma^1_t(P^1_t), \gamma^2_t(P^2_t)) + V_{t+1}(\pi_{t+1} | \pi_t, \gamma^1_t, \gamma^2_t)]$$
Information state for coordinated system

The coordinated system is a centralized partially observed system.

\[
\text{Info state} = \mathbb{P}(\text{state for I/O mapping} \mid \text{data at controller})
\]

\[
\pi_t = \mathbb{P}(X_t, P_t^1, P_t^2 \mid K_t, \gamma_t^1, \gamma_t^2)
\]

Structural Result

There is no loss of optimality in restricting prescriptions of the form

\[
\gamma_t = \psi_t(\pi_t) \quad \text{and hence,} \quad U_t^i = g_t^i(\pi_t, P_t^i)
\]

Dynamic Programming decomposition

An optimal coordination strategy is given by the solution to the following dynamic program

\[
V_t(\pi_t) = \min_{\gamma_t^1, \gamma_t^2} \mathbb{E}[c(X_t, \gamma_t^1(P_t^1), \gamma_t^2(P_t^2)) + V_{t+1}(\pi_{t+1} \mid \pi_t, \gamma_t^1, \gamma_t^2)]
\]

Setting \(g_t^i(\pi_t, P_t^i) = \psi_t^i(\pi_t)(P_t^i)\) gives optimal control strategy.
An easy solution to long standing open problem
Outline of this talk

Decentralized stochastic control
   Classification and examples

Solution approaches
   A common information based approach

Delayed sharing information structure
   Structure of optimal strategies and dynamic programming decomposition

Concluding remarks
   Generalizations and Connection to other results
Connections

Many existing results on decentralized control are special cases

- Delayed state sharing (Aicardi *et al*, 1987)
- Periodic sharing information structures (Ooi *et al*, 1997)
- Control sharing (Bismut, 1972; Sandell and Athans, 1974; Mahajan 2011)
- Finite state memory controllers (Sandell, 1974, Mahajan, 2008)
Connections

Many existing results on decentralized control are special cases

- Delayed state sharing (Aicardi et al, 1987)
- Periodic sharing information structures (Ooi et al, 1997)
- Control sharing (Bismut, 1972; Sandell and Athans, 1974; Mahajan 2011)
- Finite state memory controllers (Sandell, 1974, Mahajan, 2008)

Generalization to other models

- **Infinite horizon** (discounted and average cost) models using standard results for POMDPs
- **Computation algorithms** based on algorithms for POMDPs
- Extend results to systems with **unknown models** based on Q-learning and adaptive control algorithms
Conclusion

Summary of the main idea

- Find **common information** at the controllers
- Look from the point of view of a **coordinator** that observes common information and chooses **prescriptions** to the controllers
- Find **information state** for the coordinated system and use it to set up a **dynamic program**
Conclusion

Summary of the main idea

- Find **common information** at the controllers
- Look from the point of view of a **coordinator** that observes common information and chooses **prescriptions** to the controllers
- Find **information state** for the coordinated system and use it to set up a **dynamic program**

Future Directions

- Computational algorithms
- Connections with sequential games
- Connections with large scale systems/mean field theory
Thank you
References

1. A. Nayyar, A. Mahajan, D. Teneketzis,
   Optimal control strategies for delayed sharing information structures,

2. A. Nayyar, A. Mahajan, D. Teneketzis,
   Dynamic programming for decentralized stochastic control with partial
   information sharing: a common information approach,