The common-information approach to multi-agent teams

Aditya Mahajan
McGill University

ACC Workshop on Learning and Control of Multi-Agent Systems
9 July 2024

- email: aditya.mahajan@mcgill.ca
- web: https://adityam.github.io

Acknowledgements

Demos Teneketzis

Ashutosh Nayyar

Acknowledgements

Demos Teneketzis

Ashutosh Nayyar

Jhelum Chakravorty

Jayakumar Subramanian

Mohammad Afshari

Acknowledgements


Demos Teneketzis


Ashutosh Nayyar


Jhelum Chakravorty


Jayakumar Subramanian

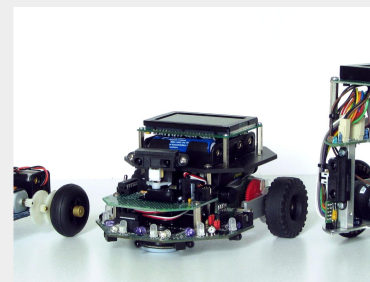

Mohammad Afshari

NSERC
CRSNG

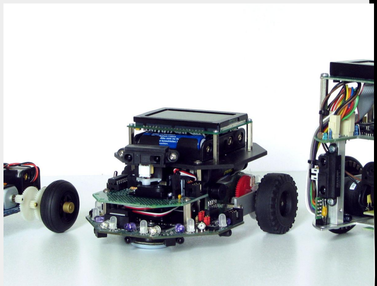
IDEaS
INNOVATION FOR DEFENCE
EXCELLENCE AND SECURITY

Common theme: multi-stage multi-agent decision making under uncertainty

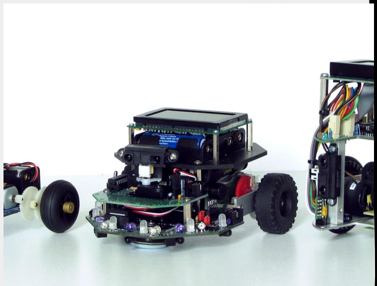
Networked control systems



Networked control systems



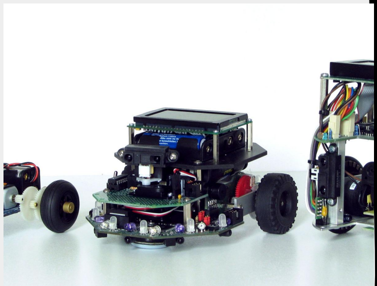
Networked control systems



Challenges

- Signals sent over wireless channels (packet drops)

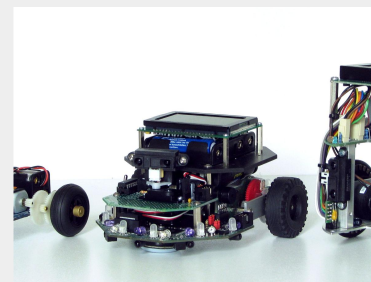
Networked control systems



Challenges

- Signals sent over wireless channels (packet drops)
- Different vehicles have different information

Networked control systems



Challenges

- Signals sent over wireless channels (packet drops)
- Different vehicles have different information
  - Decentralized control
  - Decentralized estimation
  - Decentralized learning

Salient Features of Modern Engineering Systems

Salient Features of Modern Engineering Systems

Multiple agents

Agents have different partial information about the environment and each other

Salient Features of Modern Engineering Systems

Multiple agents

Agents have different partial information about the environment and each other

Decentralized Coordination

All agents must coordinate to achieve a system-wide objective

Salient Features of Modern Engineering Systems

Multiple agents

Agents have different partial information about the environment and each other

Decentralized Coordination

All agents must coordinate to achieve a system-wide objective

Communication and Signaling

Possible to explicitly or implicitly communicate information

Salient Features of Modern Engineering Systems

Multiple agents

Agents have different partial information about the environment and each other

Decentralized Coordination

All agents must coordinate to achieve a system-wide objective

Communication and Signaling

Possible to explicitly or implicitly communicate information

Learning

Dynamics may not be completely known or may change over time

Salient Features of Modern Engineering Systems

Multiple agents

Agents have different partial information about the environment and each other

Decentralized Coordination

All agents must coordinate to achieve a system-wide objective

Communication and Signaling

Possible to explicitly or implicitly communicate information

Learning

Dynamics may not be completely known or may change over time

Teams versus Games

Teams vs Games

Teams

- All agents have **common objective**
- Agents **cooperate** to min team cost
- Agents are **not strategic**
- Solution concepts: person-by-person optimality, global optimality . . .

Games

- Each agent has **individual objective**
- Agents **compete** to minimize individual cost
- Agents are **strategic**
- Solution concepts: Nash equil, Bayesian Nash, Subgame perfect, Markov perfect, Bayesian perfect, . . .

Teams vs Games

Teams

- All agents have **common objective**

Games

- Each agent has **individual objective**

In many engineering problems, game theory is used as an algorithmic toolbox to provide distributed solutions to **static** problems.

We are interested in finding **globally optimal** solution to problems where **agents have decentralized information**.

Teams have a reputation of being notoriously difficult . . .

Some historical context

Some historical context

S&C until the 1960s

- About 300 years of knowledge in designing **LTI systems**
- Good "intuitive" understanding of **frequency domain methods**
  - Root locus
  - Bode plots
  - Nyquist plots
  - Loop shaping

Some historical context

S&C until the 1960s

- About 300 years of knowledge in designing **LTI systems**
- Good "intuitive" understanding of **frequency domain methods**
  - Root locus
  - Bode plots
  - Nyquist plots
  - Loop shaping

Advances in 1960s

- Emergence of **state space methods** for filtering and control
- Could be implemented in digital computers (of that time!)

Some historical context

S&C until the 1960s

- About 300 years of knowledge in designing **LTI systems**
- Good "intuitive" understanding of **frequency domain methods**
  - Root locus
  - Bode plots
  - Nyquist plots
  - Loop shaping

Advances in 1960s

- Emergence of **state space methods** for filtering and control
- Could be implemented in digital computers (of that time!)

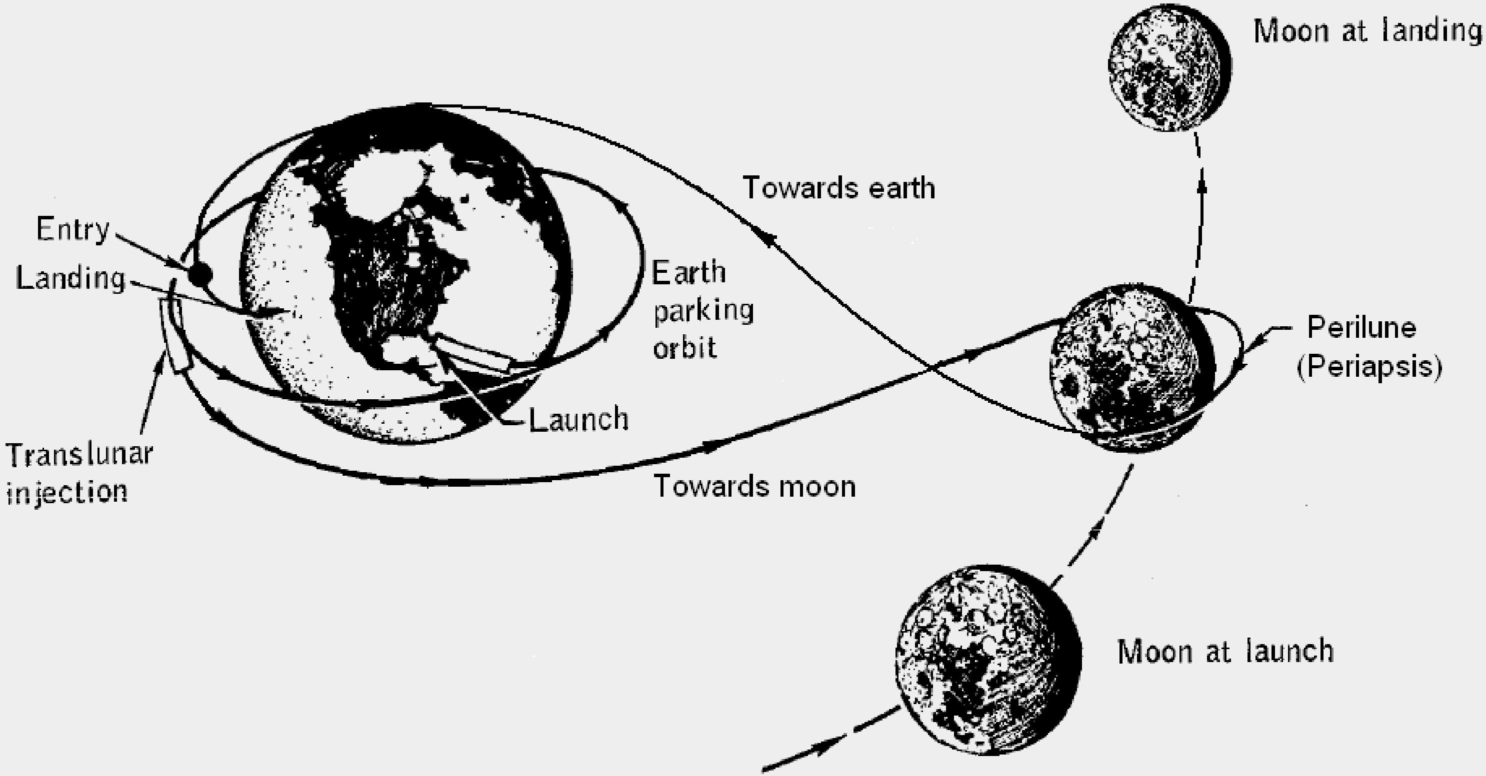
State Space Deisgn

- Linearize the system dynamics
- Design **optimal control** assuming full state feedback (LQR)

  control action(t) = −gain(t) · state(t)
- Estimate the state using noisy measurements (Kalman filtering)

  state estimate(t) = Function(estimate(t-1), measurement(t).
- **Optimal controller**:

  control action(t) = −gain(t) · state estimate(t)



Moon at landing
Towards earth
Entry
Landing
Earth
parking
orbit
Perilune
(Periapsis)
Launch
Translunar
injection
Towards moon
Moon at launch

Conceptual difficulties in team problems

Witsenhausen Counterexample (1968)

- A two step dynamical system with two controllers
- Linear dynamics, quadratic cost, and Gaussian disturbance
- Non-linear controllers outperform linear control strategies . . .

  . . . cannot use Kalman filtering + Riccati equations
- Later papers: Non-linear can perform arbitrarily well compared to linear.

Witsenhausen, "A counterexample in stochastic optimum control," SICON 1968.

Whittle and Rudge, "The optimal linear solution of a symmetric team control problem," App. Prob. 1974.

Bernstein, et al, "The complexity of decentralized control of Markov decision processes," MOR 2002.

Conceptual difficulties in team problems

Witsenhausen Counterexample (1968)

- A two step dynamical system with two controllers
- Linear dynamics, quadratic cost, and Gaussian disturbance
- Non-linear controllers outperform linear control strategies . . .

  . . . cannot use Kalman filtering + Riccati equations
- Later papers: Non-linear can perform arbitrarily well compared to linear.

Whittle and Rudge Example (1974)

- Infinite horizon dynamical system with two symmetric controllers
- Linear dynamics, quadratic cost, and Gaussian disturbance
- *A priori* restrict attention to linear controllers
- Best linear controllers **don't** have finite dimensional representation

---

Witsenhausen, "A counterexample in stochastic optimum control," SICON 1968.

Whittle and Rudge, "The optimal linear solution of a symmetric team control problem," App. Prob. 1974.

Bernstein, et al, "The complexity of decentralized control of Markov decision processes," MOR 2002.

Conceptual difficulties in team problems

Witsenhausen Counterexample (1968)	▷ A two step dynamical system with two controllers ▷ Linear dynamics, quadratic cost, and Gaussian disturbance ▷ Non-linear controllers outperform linear control strategies . . . . . . cannot use Kalman filtering + Riccati equations ▷ Later papers: Non-linear can perform arbitrarily well compared to linear.
Whittle and Rudge Example (1974)	▷ Infinite horizon dynamical system with two symmetric controllers ▷ Linear dynamics, quadratic cost, and Gaussian disturbance ▷ A priori restrict attention to linear controllers ▷ Best linear controllers don't have finite dimensional representation
Complexity analysis	▷ All random variables are finite valued ▷ Finite horizon setup ▷ The problem of finding the best control strategy is in NEXP

Witsenhausen, "A counterexample in stochastic optimum control," SICON 1968.

Whittle and Rudge, "The optimal linear solution of a symmetric team control problem," App. Prob. 1974.

Bernstein, et al, "The complexity of decentralized control of Markov decision processes," MOR 2002.

Why are team problems hard?

Why are team problems hard?

Why are single agent problems easy?

Static stochastic optimization problems

$$\min_{g:\mathcal{Y}\to\mathcal{U}} \mathbb{E}[c(X, g(Y))]$$

	$X = 0$	$X = 1$	$X = 2$	$X = 3$
$U = 0$	0.5	0.2	1.2	0.5
$U = 1$	1.2	0.5	0.2	0.3

$Y = 0$	$Y = 1$

Static stochastic optimization problems

$$\min_{g : \mathcal{Y} \to \mathcal{U}} \mathbb{E}[c(X, g(Y))]$$

	$X = 0$	$X = 1$	$X = 2$	$X = 3$
$u = 0$	0.5	0.2	1.2	0.5
$u = 1$	1.2	0.5	0.2	0.3
	$Y = 0$		$Y = 1$	

Static stochastic optimization problems

$$\min_{g:\mathcal{Y}\to\mathcal{U}} \mathbb{E}[c(X, g(Y))]$$

	$X=0$	$X=1$	$X=2$	$X=3$
$U=0$	0.5	0.2	1.2	0.5
$U=1$	1.2	0.5	0.2	0.3

$Y=0$	$Y=1$

- This is a functional optimization problem.
- Search complexity $|\mathcal{U}|^{|\mathcal{Y}|}$.

Static stochastic optimization problems

$$\min_{g:\mathcal{Y}\to\mathcal{U}} \mathbb{E}[c(X, g(Y))]$$

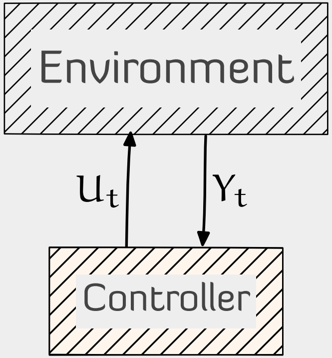
	$X=0$	$X=1$	$X=2$	$X=3$
$u=0$	0.5	0.2	1.2	0.5
$u=1$	1.2	0.5	0.2	0.3
	$Y=0$		$Y=1$	

- This is a functional optimization problem.
- Search complexity $|\mathcal{U}|^{|\mathcal{Y}|}$.

$$\text{for each } y, \quad \min_{u\in\mathcal{U}} \mathbb{E}[c(X, u) \mid Y = y]$$

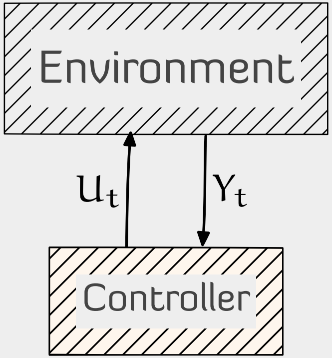
- Each sub-problem is a parameter optimization problem.
- Search complexity $|\mathcal{U}| \cdot |\mathcal{Y}|$.

Dynamic stochastic optimization problems



Dyanmics	$X_{t+1} = f_t(X_t, U_t, W_t)$
Observations	$Y_t = h^i_t(X_t, N_t)$
Control law	$U_t = g_t(Y_{1:t}, U_{1:t-1})$

Dynamic stochastic optimization problems



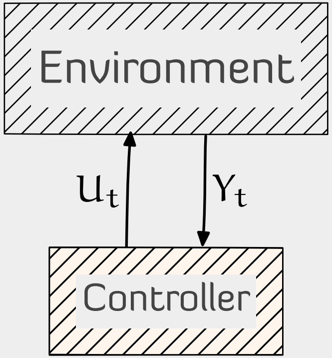
Dyanmics	$X_{t+1} = f_t(X_t, U_t, W_t)$
Observations	$Y_t = h^i_t(X_t, N_t)$
Control law	$U_t = g_t(Y_{1:t}, U_{1:t-1})$

Objective

Choose control strategy $g = (g_1, \dots, g_T)$ to minimize

$$J(g) = \mathbb{E}\Big[\sum_{t=1}^{T} c_t(X_t, U_t)\Big]$$

Dynamic stochastic optimization problems



Dyanmics	$X_{t+1} = f_t(X_t, U_t, W_t)$
Observations	$Y_t = h^i_t(X_t, N_t)$
Control law	$U_t = g_t(Y_{1:t}, U_{1:t-1})$

Objective: Choose control strategy $g = (g_1, \dots, g_T)$ to minimize

Dynamic programming solution

- Define **belief state** $b_t = P(X_t | Y_{1:t}, U_{1:t-1})$.
- Write a DP in terms of the belief state b_t.
- Solution complexity: $T \cdot |\mathcal{U}| \cdot |\mathcal{Z}|$.

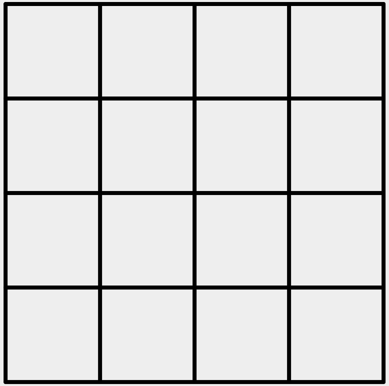
Why don't these simplifications work for teams?

Static team problem

$$\min_{g^1, g^2} \mathbb{E}[c(X, g^1(Y^1), g^2(Y^2))]$$

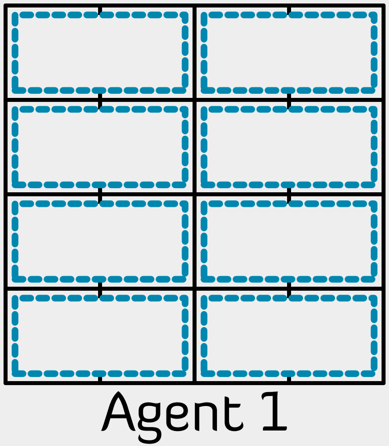
Static team problem

$$\min_{g^1, g^2} \mathbb{E}[c(X, g^1(Y^1), g^2(Y^2))]$$



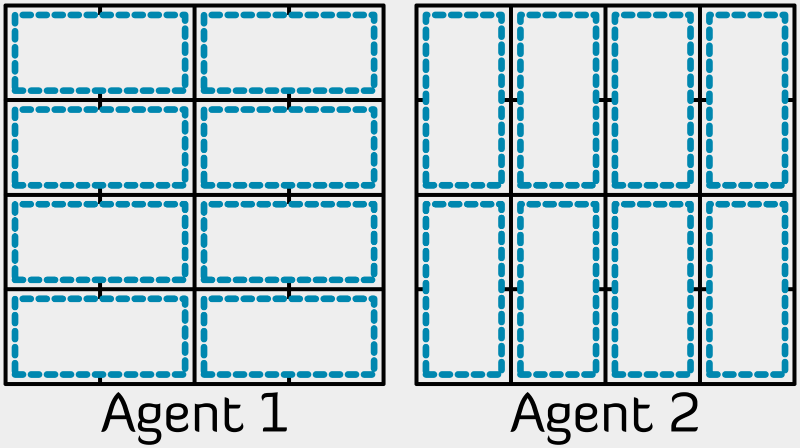
Static team problem

$$\min_{g^1, g^2} \mathbb{E}[c(X, g^1(Y^1), g^2(Y^2))]$$



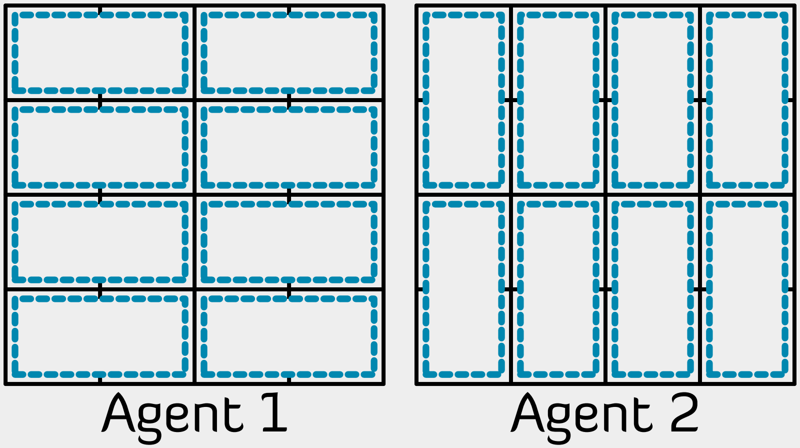
Static team problem

$$\min_{g^1, g^2} \mathbb{E}[c(X, g^1(Y^1), g^2(Y^2))]$$



Static team problem

$$\min_{g^1, g^2} \mathbb{E}[c(X, g^1(Y^1), g^2(Y^2))]$$

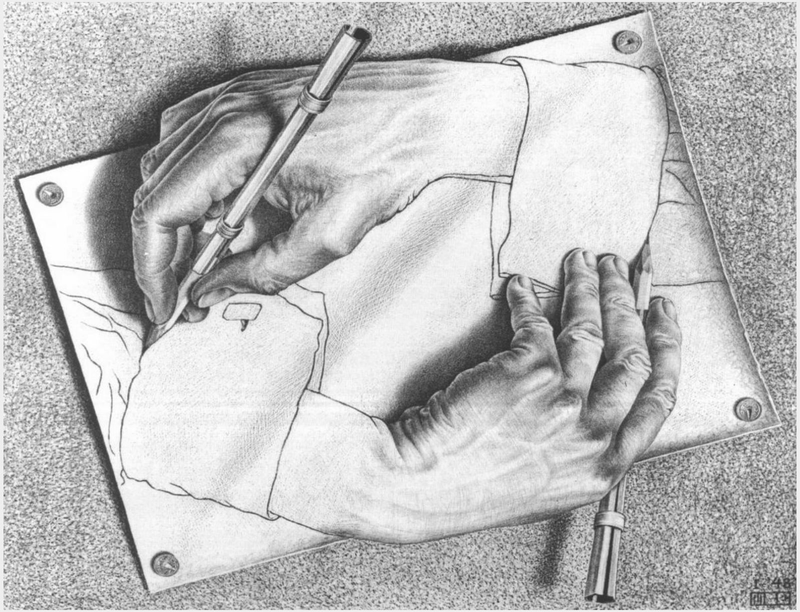


Previous idea of

$$\text{for all } y^1, \quad \min_{u^1} \mathbb{E}[c(X, u^1, g^2(Y^2)) \mid Y^1 = y^1]$$

leads to person-by-person optimal solution (not globally opt.)

Static team problem



$$\min_{g^1, g^2} \mathbb{E}[c(X, g^1(Y^1), g^2(Y^2))]$$

Previous idea of

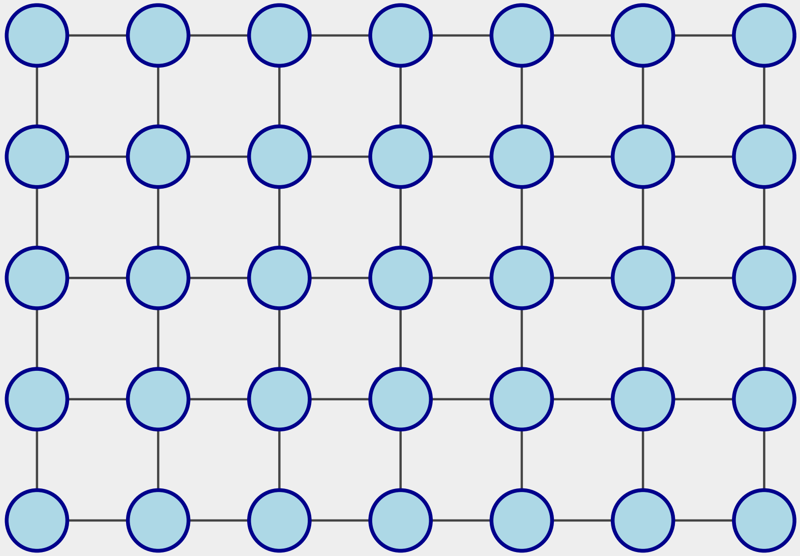
for all y^1, $\min_{u^1} \mathbb{E}[c(X, u^1, g^2(Y^2)) \mid Y^1 = y^1]$

leads to person-by-person optimal solution (not globally opt.)

There are additional challenges in dynamic problems

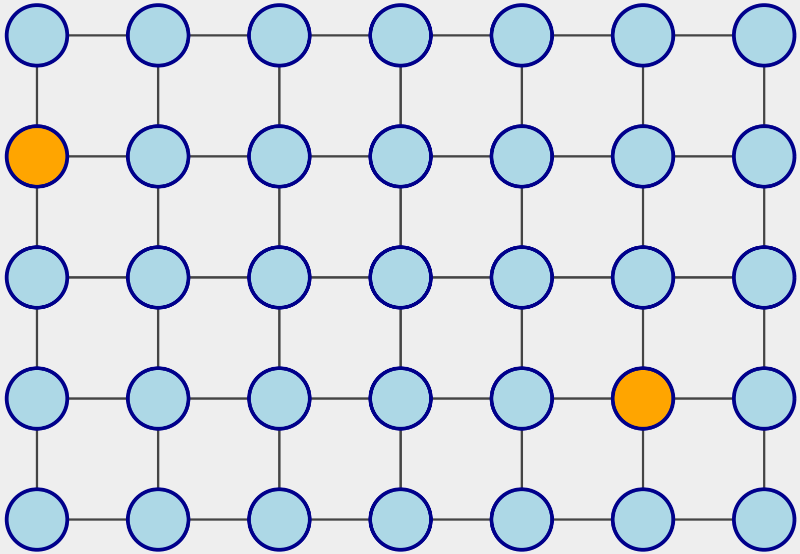
k-step delayed sharing information structure

- Consider a network with coupled dynamics.
- Information exchange between nodes with unit delay.



k-step delayed sharing information structure

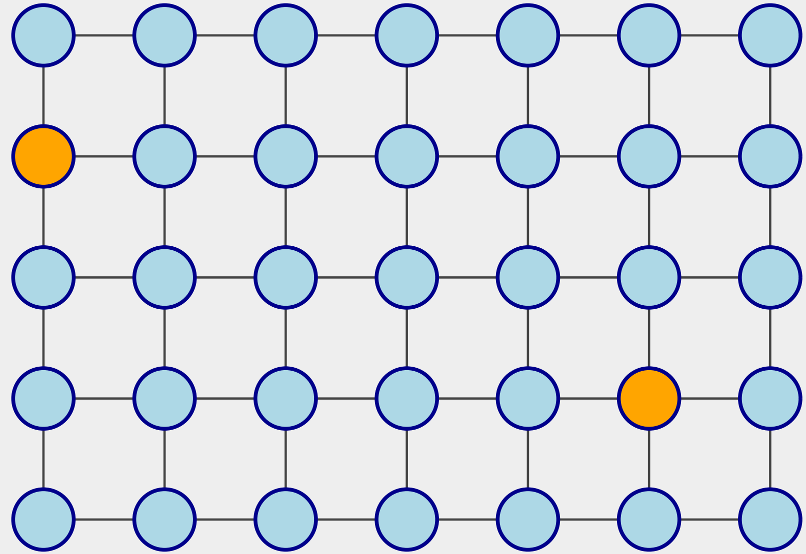
- Consider a network with coupled dynamics.
- Information exchange between nodes with unit delay.



- Fix the strategy of all but two subsystems which are k-hop apart. What is the best response strategy at these two nodes?

Witsenhausen, "Separation of Esitmation and Control for Discrete-Time Systems," Proc. IEEE, 1971.

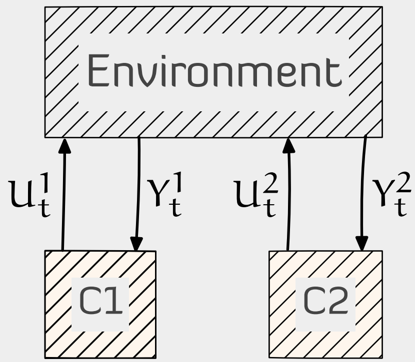
k-step delayed sharing information structure



- Consider a network with coupled dynamics.
- Information exchange between nodes with unit delay.
- Fix the strategy of all but two subsystems which are k-hop apart. What is the best response strategy at these two nodes?
- Proposed by Witsenhausen in a seminal paper.
- Allows to smoothly transition between centralized ($k = 0$) and completely decentralized ($k = \infty$).

Witsenhausen, "Separation of Esitmation and Control for Discrete-Time Systems," Proc. IEEE, 1971.

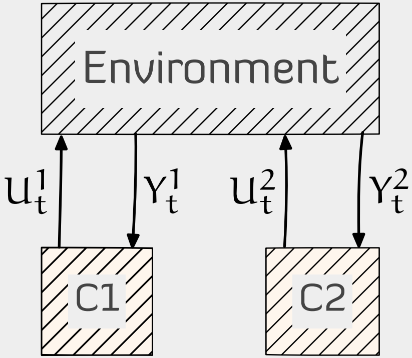
System Model



Dyanmics $X_{t+1} = f_t(X_t, U_t^1, U_t^2, W_t)$

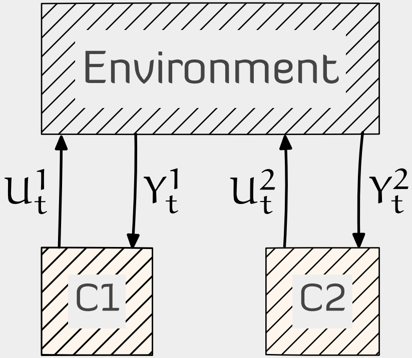
Observations $Y_t^i = h_t^i(X_t, N_t^i)$

System Model



Dyanmics	$X_{t+1} = f_t(X_t, U_t^1, U_t^2, W_t)$
Observations	$Y_t^i = h_t^i(X_t, N_t^i)$
Information Structure	$I_t^i = \{Y^i_{1:t}, U^i_{1:t-1} Y^{-i}_{1:t-k}, U^{-i}_{1:t-k}\}$
Control law	$U_t^i = g_t^i(I_t^i)$

System Model



Dyanmics: $X_{t+1} = f_t(X_t, U_t^1, U_t^2, W_t)$

Observations: $Y_t^i = h_t^i(X_t, N_t^i)$

Information Structure: $I_t^i = \{Y_{1:t}^i, U_{1:t-1}^i Y_{1:t-k}^{-i}, U_{1:t-k}^{-i}\}$

Control law: $U_t^i = g_t^i(I_t^i)$

Objective: Choose control strategies (g_1, g_2) to minimize

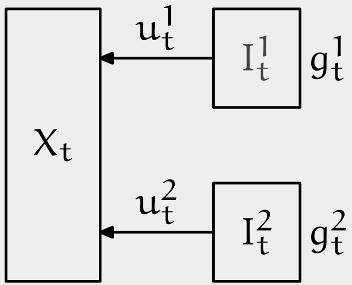
$$J(g_1, g_2) = \mathbb{E}\left[\sum_{t=1}^{T} c_t(X_t, U_t^1, U_t^2)\right]$$

Conceptual difficulty

The data I_t^i available at each controller is increasing with time.
How to find a sufficient statistic or an information state?

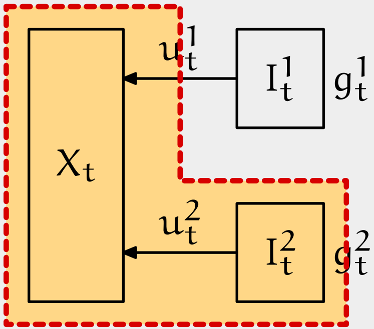
Conceptual difficulty

The data I_t^i available at each controller is increasing with time.
How to find a sufficient statistic or an information state?



Conceptual difficulty

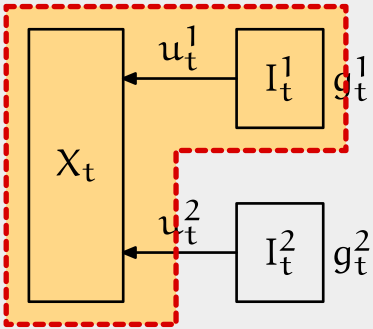
The data I_t^i available at each controller is increasing with time. How to find a sufficient statistic or an information state?



- Unobserved state from the p.o.v. of ctrl 1: X_t, I_t^2.
  Information state $\pi_t^1 = \mathbb{P}(X_t, I_t^2 | I_t^1)$.

Conceptual difficulty

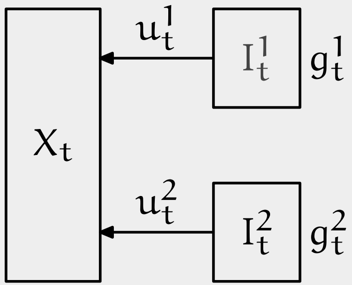
The data I_t^i available at each controller is increasing with time.
How to find a sufficient statistic or an information state?



- Unobserved state from the p.o.v. of ctrl 1: X_t, I_t^2.
  Information state $\pi_t^1 = \mathbb{P}(X_t, I_t^2 | I_t^1)$.
- Unobserved state from the p.o.v. of ctrl 2: X_t, π_t^1.
  Information state $\pi_t^2 = \mathbb{P}(X_t, \pi_t^1 | I_t^2)$.

Conceptual difficulty

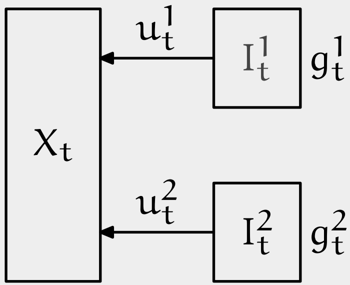
The data I_t^i available at each controller is increasing with time. How to find a sufficient statistic or an information state?

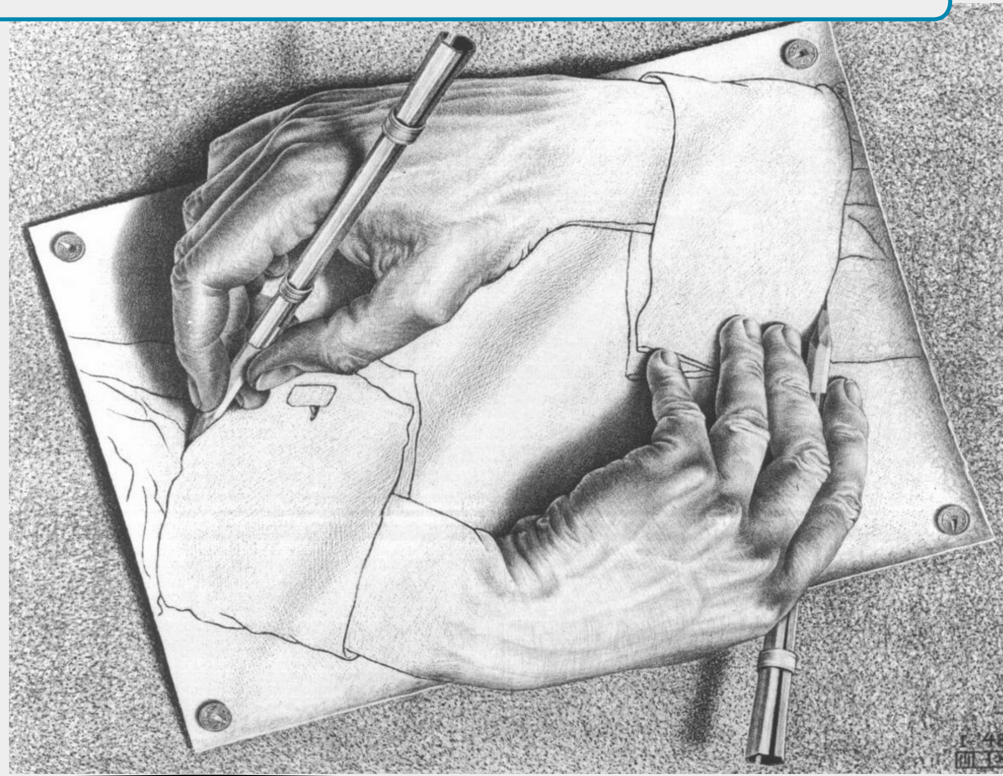


- Unobserved state from the p.o.v. of ctrl 1: X_t, I_t^2.
  Information state $\pi_t^1 = \mathbb{P}(X_t, I_t^2 | I_t^1)$.
- Unobserved state from the p.o.v. of ctrl 2: X_t, π_t^1.
  Information state $\pi_t^2 = \mathbb{P}(X_t, \pi_t^1 | I_t^2)$.
- Unobserved state from the p.o.v. of ctrl 1: X_t, π_t^2.
  Information state $\pi_t^{1,2} = \mathbb{P}(X_t, \pi_t^2 | I_t^1)$.
- . . . infinite regress . . .

Conceptual difficulty

The data I_t^i available at each controller is increasing with time.
How to find a sufficient statistic or an information state?





History of the problem

Witsenhausen's Assertion

Let $C_t = \{Y_{1:t-k}, U_{1:t-k}\}$ and $L^i_t = \{Y^i_{t-k+1:t}, u^i_{t-k+1:t-1}\}$.
Then $\mathbb{P}(X_{t-k} \mid C_t)$ is a sufficient statistic for C_t.

Rationale: $\mathbb{P}(X_{t-k} | Y_{1:t-k}, U_{1:t-k})$ is policy independent.

Witsenhausen, "Separation of Esitmation and Control for Discrete-Time Systems," Proc. IEEE, 1971.

History of the problem

Witsenhausen's Assertion

Let $C_t = \{Y_{1:t-k}, U_{1:t-k}\}$ and $L^i_t = \{Y^i_{t-k+1:t}, u^i_{t-k+1:t-1}\}$.
Then $\mathbb{P}(X_{t-k} \mid C_t)$ is a sufficient statistic for C_t.

Rationale: $\mathbb{P}(X_{t-k} | Y_{1:t-k}, U_{1:t-k})$ is policy independent.

Follow-up Literature

- Assertion true for $k=1$
  [Sandell, Athans, 1974], [Kurtaran, 1976]
- Assertion false for $k>1$
  [Varaiya, Walrand 1979], [Yoshikawa, Kobayashi, 1978]
- No subsequent positive result!

Witsenhausen, "Separation of Esitmation and Control for Discrete-Time Systems," Proc. IEEE, 1971.

History of the problem

Witsenhausen's Assertion

Let $C_t = \{Y_{1:t-k}, U_{1:t-k}\}$ and $L_t^i = \{Y_{t-k+1:t}^i, u_{t-k+1:t-1}^i\}$.
Then $\mathbb{P}(X_{t-k} \mid C_t)$ is a sufficient statistic for C_t.

Rationale: $\mathbb{P}(X_{t-k} | Y_{1:t-k}, U_{1:t-k})$ is policy independent.

Follow-up Literature

- Assertion true for $k=1$
  [Sandell, Athans, 1974], [Kurtaran, 1976]
- Assertion false for $k>1$
  [Varaiya, Walrand 1979], [Yoshikawa, Kobayashi, 1978]
- No subsequent positive result!

Are there sufficient statistics or information states for C_t?

Witsenhausen, "Separation of Esitmation and Control for Discrete-Time Systems," Proc. IEEE, 1971.

Importance of the problem

Applications (of one-step delay sharing)

- **Power systems**: Altman et al, 2009
- **Queueing theory**: Kuri and Kumar, 1995
- **Communication networks**: Grizzle et al, 1982
- **Stochastic games**: Papavassilopoulos, 1982; Chang and Cruz, '83
- **Economics**: Li and Wu, 1991.

Importance of the problem

Applications (of one-step delay sharing)

- Power systems: Altman et al, 2009
- Queueing theory: Kuri and Kumar, 1995
- Communication networks: Grizzle et al, 1982
- Stochastic games: Papavassilopoulos, 1982; Chang and Cruz, '83
- Economics: Li and Wu, 1991.

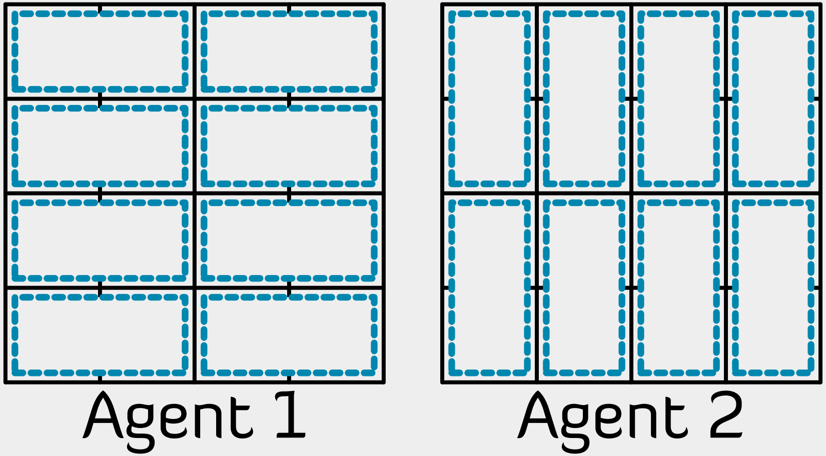
Conceptual Significance

- Understanding the design of networked control systems
- Bridge between centralized and decentralized systems
- Insights for the design of general decentralized systems.

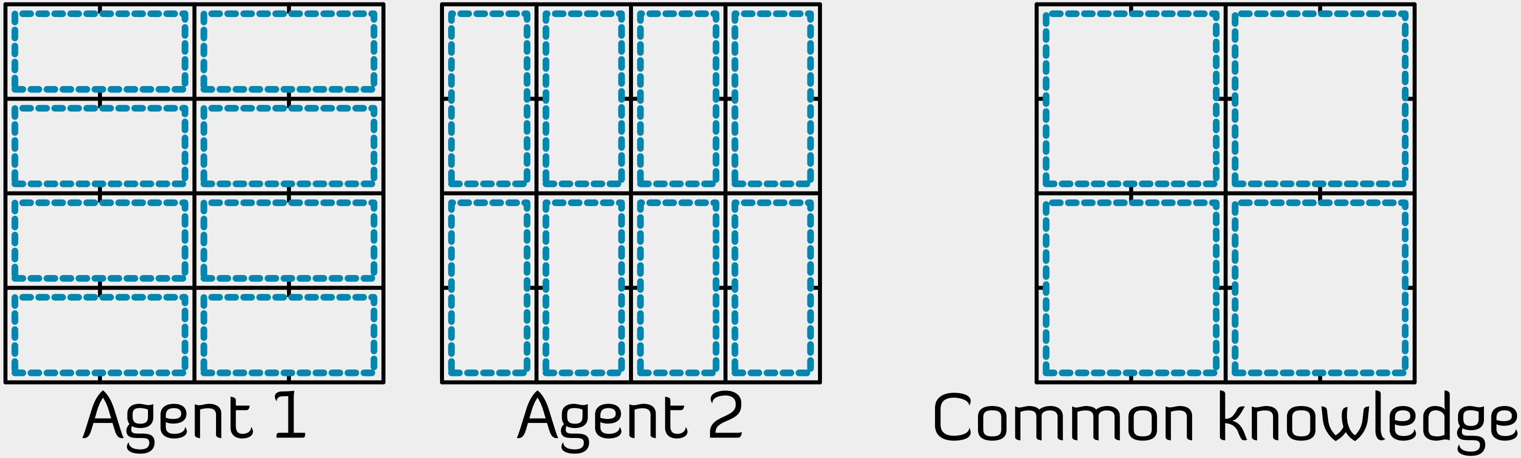
Common information approach for teams

[Nayyar, Mahajan, Teneketzis (TAC 2011, 2013)]

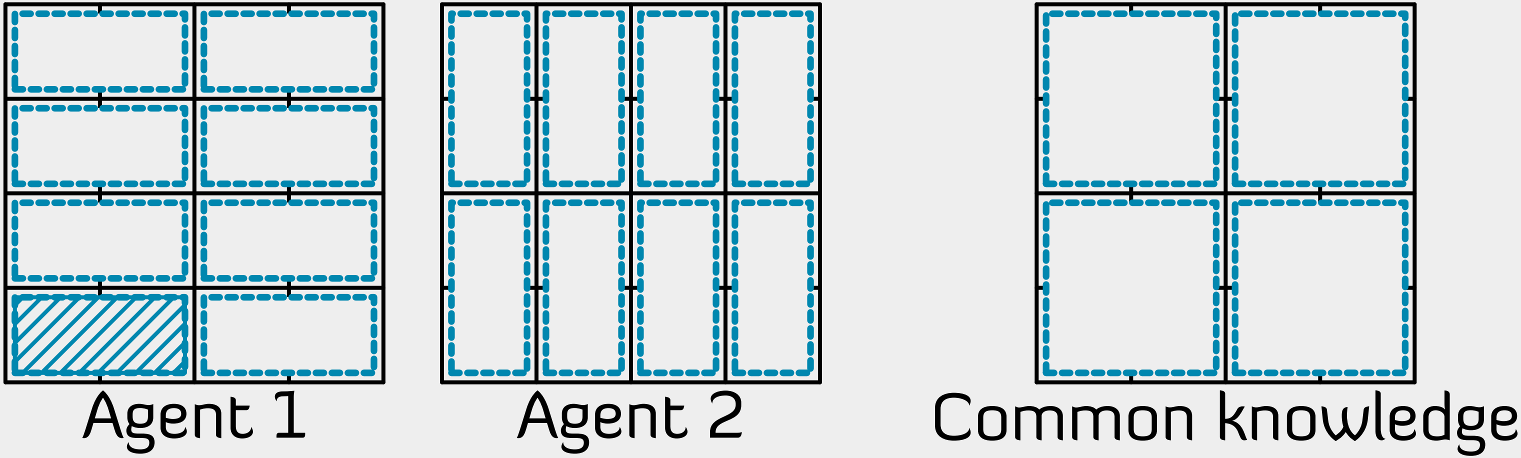
Key idea: exploit common knowledge



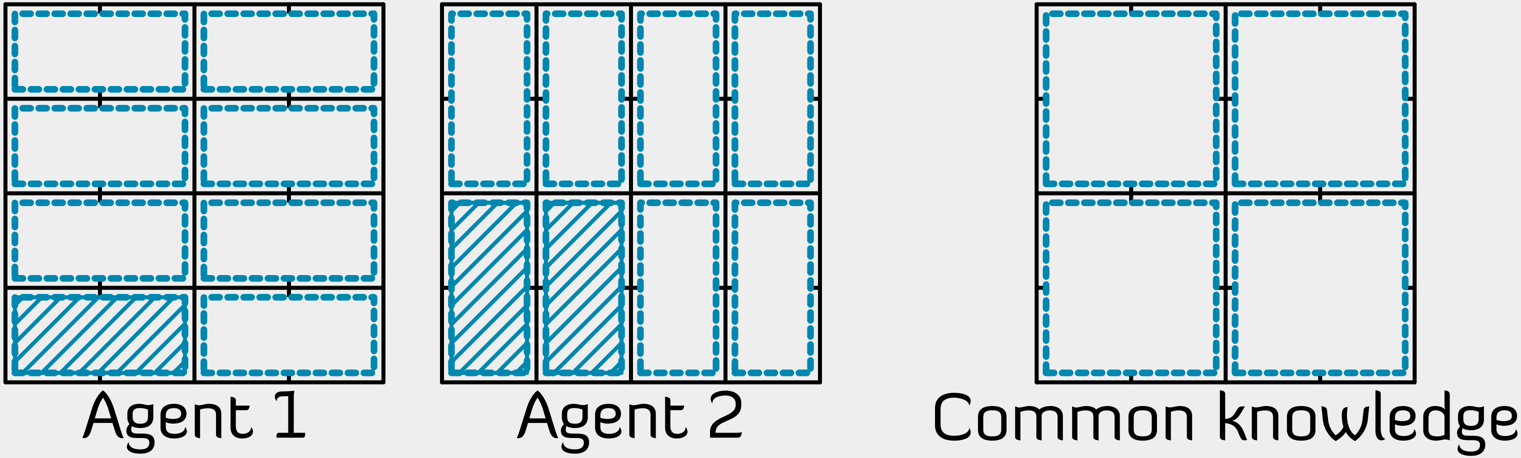
Key idea: exploit common knowledge



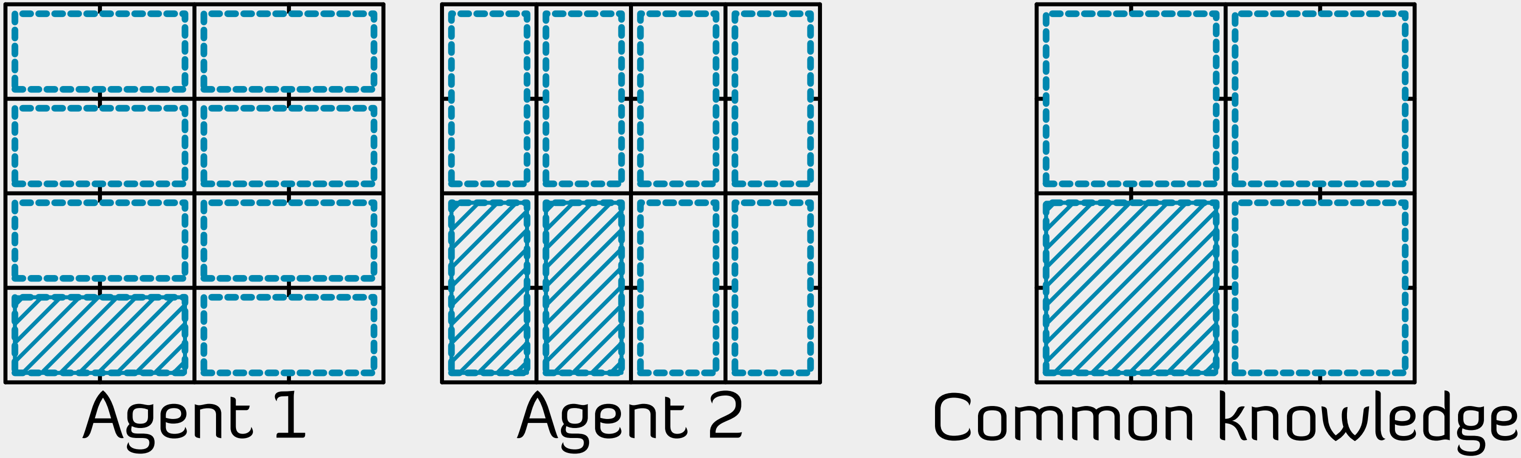
Key idea: exploit common knowledge



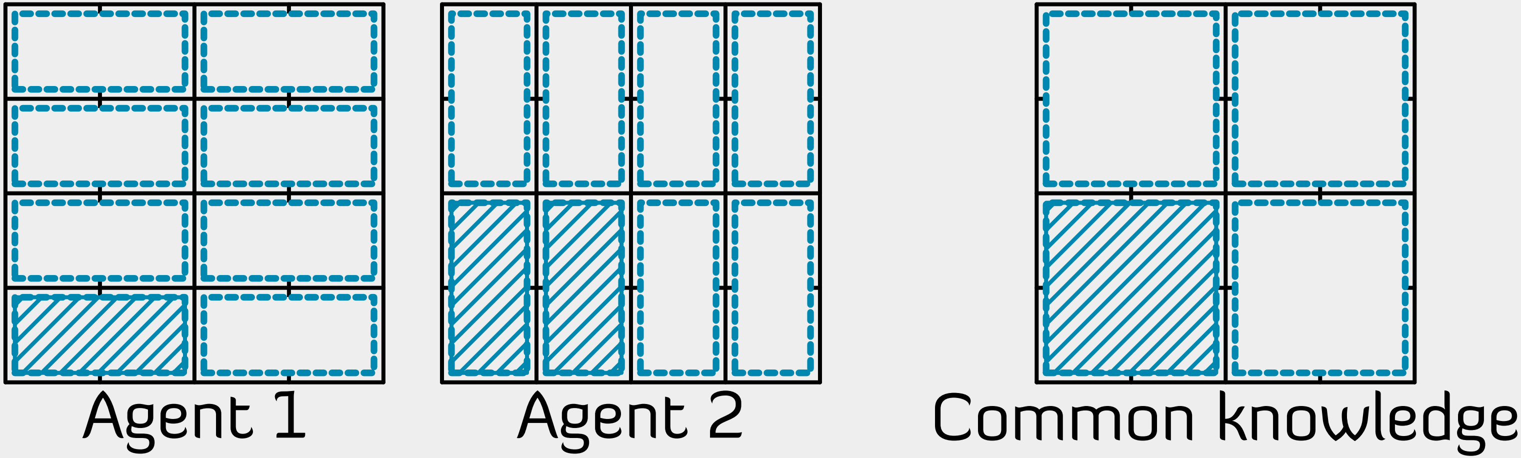
Key idea: exploit common knowledge



Key idea: exploit common knowledge



Key idea: exploit common knowledge



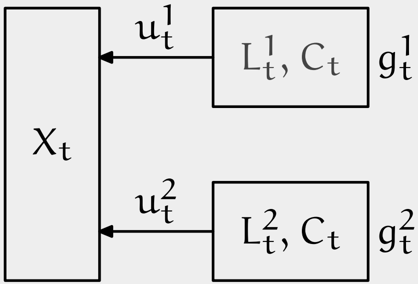
Split $Y^1 = (L^1, C)$ and $Y^2 = (L^2, C)$.

for all c, $\min_{\gamma^1, \gamma^2} \mathbb{E}[c(X, \gamma^1(L^1), \gamma^2(L^2))) \mid C = c]$

Reduction in complexity: $|\mathcal{U}|^8 \cdot |\mathcal{U}|^8$ to $4|\mathcal{U}|^2 \cdot |\mathcal{U}|^2$

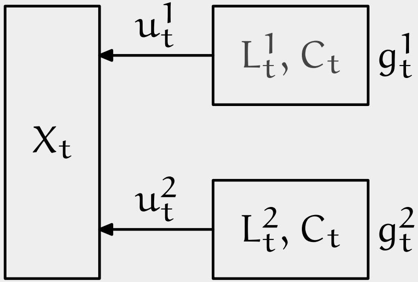
Common-info approach for k-step delay sharing

Original System

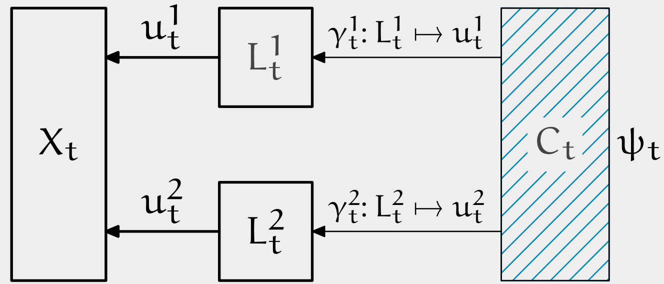


Common-info approach for k-step delay sharing

Original System

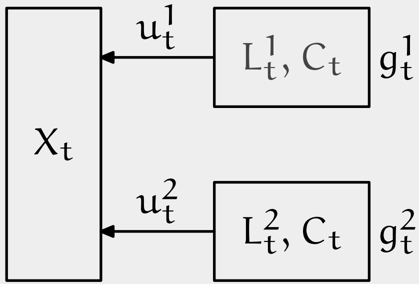


Virtual Coordinated System

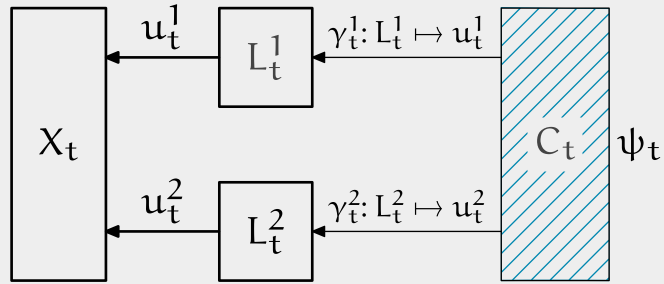


Common-info approach for k-step delay sharing

Original System



Virtual Coordinated System



Information split

- Common information: $C_t = I_t^1 \cap I_t^2 = \{Y_{1:t-k}, U_{1:t-k}\}$
- Local information: $L_t^i = I_t^i \setminus C_t = \{Y^i_{t-k+1:t}, u^i_{t-k+1:t-1}\}$.
- Prescription: $\gamma_t^i \colon L_t^i \mapsto u_t^i$.

Common-info approach for k-step delay sharing

Main Result

- The virtual coordinator is a single agent stochastic ctrl problem.
- Information state: for C_t: $b_t = \mathbb{P}(X_t, L_t^1, L_t^2 \mid C_t, \gamma^1_{1:t-1}, \gamma^2_{1:t-1})$.
- Dynamic program: $V_{T+1}(b) = 0$ and

$$V_t(b_t) = \min_{\gamma_t^1, \gamma_t^2} \left\{ \mathbb{E}[c_t(X_t, u_t^1, u_t^2) + V_{t+1}(B_+) \mid b_t, \gamma_t^1, \backslash\gamma_t^2] \right\}.$$

- Each step of the DP is a functional optimization problem.

Common-info approach for k-step delay sharing

Main Result

- The virtual coordinator is a single agent stochastic ctrl problem.
- **Information state**: for C_t: $b_t = \mathbb{P}(X_t, L^1_t, L^2_t \mid C_t, \gamma^1_{1:t-1}, \gamma^2_{1:t-1})$
- **Dynamic program**: $V_{T+1}(b) = 0$ and

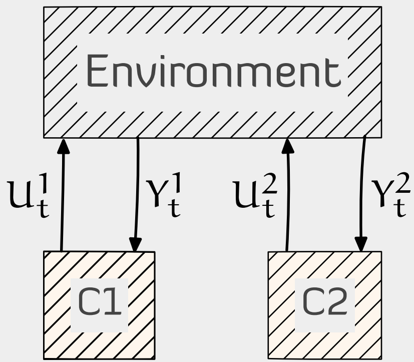
$$V_t(b_t) = \min_{\gamma^1_t, \gamma^2_t} \left\{ \mathbb{E}[c_t(X_t, u^1_t, u^2_t) + V_{t+1}(B_+) \mid b_t, \gamma^1_t, \backslash\gamma^2_t] \right\}.$$

- Each step of the DP is a **functional** optimization problem.

Salient Features

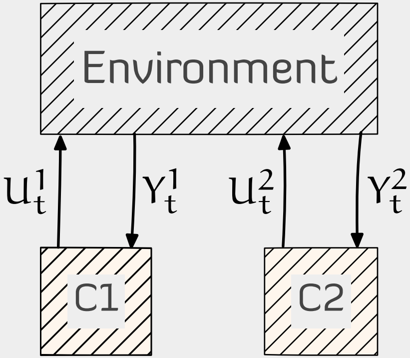
- The virtual coordinator is purely for conceptual clarity as it allows us to view the original problem from the p.o.v. of a “higher authority”. The presence of the coordinator is not necessary.
- The common information is known to both controllers and therefore both of them can carry out the calculations to solve the DP on their own.

The general common-info approach



- n controllers with general info structure $\{I_t^i\}_{i=1}^n$.

The general common-info approach



- n controllers with general info structure $\{I_t^i\}_{i=1}^n$.

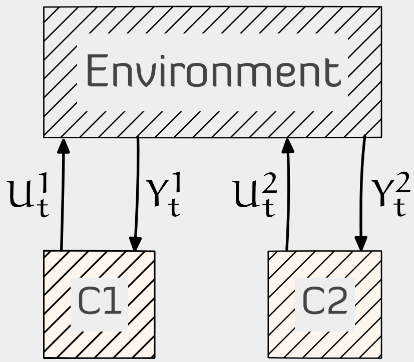
Information Split

- **Common information**:

$$C_t = \bigcap_{s \geqslant t} \bigcap_{i=1}^{n} I_s^i.$$

- **Local information**: $L_t^i = I_t^i \setminus C_t$.

The general common-info approach



- n controllers with general info structure $\{I_t^i\}_{i=1}^n$.

Information Split

- **Common information**:

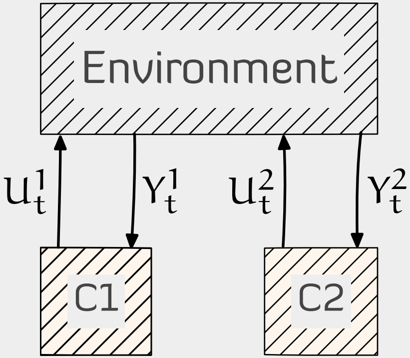
$$C_t = \bigcap_{s \geqslant t} \bigcap_{i=1}^{n} I_s^i.$$

- **Local information**: $L_t^i = I_t^i \setminus C_t$.

Partial history sharing

- $|L_t^i|$ is uniformly bounded.
- $\mathbb{P}^\psi(C_{t+1} \setminus C_t \mid C_t, \gamma_t^1, \gamma_t^2)$ doesn't depend on ψ.

The general common-info approach



- n controllers with general info structure $\{I_t^i\}_{i=1}^n$.

Information Split

- **Common information**:
$$C_t = \bigcap_{s \geqslant t} \bigcap_{i=1}^{n} I_s^i.$$
- **Local information**: $L_t^i = I_t^i \setminus C_t$.

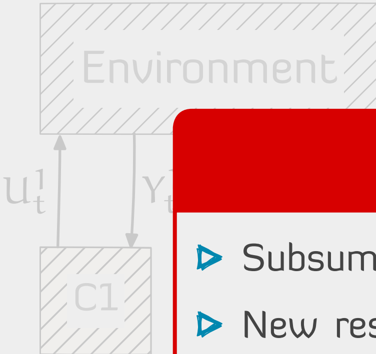
Partial history sharing

- $|L_t^i|$ is uniformly bounded.
- $\mathbb{P}^{\psi}(C_{t+1} \setminus C_t \mid C_t, \gamma_t^1, \gamma_t^2)$ doesn't depend on ψ.

Main Result

- **Information state**: for C_t: $b_t = \mathbb{P}(X_t, L_t^1, L_t^2 \mid C_t, \gamma_{1:t-1}^1, \gamma_{1:t-1}^2)$.
- **Dynamic program**: $V_{T+1}(g) = 0$ and
$$V_t(b) = \min_{\gamma_t^1, \gamma_t^2} \{\mathbb{E}[c_t(X_t, u_t^1, u_t^2) + V_{t+1}(B_+) \mid b_t, \gamma_t^1, \gamma_t^2]\}.$$

The general common-info approach



- n controllers with general info structure $\{I_t^i\}_{i=1}^n$.

... C_t.

... doesn't depend on ψ.

Implications and impact

- Subsumes many existing results (...)
- New results on sufficient statistics and DP for specific models (control sharing, mean-field sharing, NCS, and others)
- Common-information based refinements of Nash equilibrium in dynamic games with asymmetric information

Main Result

- **Information state**: for C_t: $b_t = \mathbb{P}(X_t, L_t^1, L_t^2 \mid C_t, \gamma_{1:t-1}^1, \gamma_{1:t-1}^2)$
- **Dynamic program**: $V_{T+1}(g) = 0$ and

$$V_t(b) = \min_{\gamma_t^1, \gamma_t^2} \left\{ \mathbb{E}[c_t(X_t, u_t^1, u_t^2) + V_{t+1}(B_+) \mid b_t, \gamma_t^1, \gamma_t^2] \right\}.$$

Some examples

Control sharing information structure

Dynamics $X^i_{t+1} = f^i(X^i_t, \mathbf{U}_t, W^i_t)$

Info structure $I^i_t = \{X^i_{1:t}, \mathbf{U}_{1:t}\}$

Sandell and Athans, "Solution of some non-classical LQG decision problems," TAC 1974.
Mahajan, "Optimal decentralized control of coupled subsystems with control sharing," TAC 2013.

Control sharing information structure

Dynamics	$X^i_{t+1} = f^i(X^i_t, \mathbf{U}_t, W^i_t)$

Info structure	$I^i_t = \{X^i_{1:t}, \mathbf{U}_{1:t}\}$

Step 1: Using person-by-person approach

- Show that: $X^1_{1:t} \perp X^2_{1:t} \perp \cdots \perp X^n_{1:t} \mid \mathbf{U}_{1:t}$
- Implies no loss of optimality in shedding $X^i_{1:t-1}$ at agent i.

Sandell and Athans, "Solution of some non-classical LQG decision problems," TAC 1974.

Mahajan, "Optimal decentralized control of coupled subsystems with control sharing," TAC 2013.

Control sharing information structure

Dynamics	$X^i_{t+1} = f^i(X^i_t, U_t, W^i_t)$

Info structure	$I^i_t = \{X^i_{1:t}, U_{1:t}\}$

Step 1: Using person-by-person approach

- Show that: $X^1_{1:t} \perp X^2_{1:t} \perp \cdots \perp X^n_{1:t} \mid U_{1:t}$
- Implies no loss of optimality in shedding $X^i_{1:t-1}$ at agent i.

Step 2: Use common information approach

- Common-info based belief simplies due to the conditional independence (see step 1)

Suff statistic for $U_{1:t} = (\mathbb{P}(X^1_t \mid U_{1:t}), \dots, \mathbb{P}(X^n_t \mid U_{1:t}))$

Sandell and Athans, "Solution of some non-classical LQG decision problems," TAC 1974.
Mahajan, "Optimal decentralized control of coupled subsystems with control sharing," TAC 2013.

Mean-field teams

Dynamics	$X^i_{t+1} = f^i(X^i_t, U^i_t, Z_t, W^i_t)$
Mean-field	$Z_t = \dfrac{1}{n} \sum_{i=1}^{n} \delta_{X^i_t}$

Info structure	$I^i_t = \{X^i_t, Z_{1:t}\}$

Arabneydi and Mahajan, "Team optimal control of coupled subsystems with mean field sharing," CDC 2013.

Mean-field teams

Dynamics	$X^i_{t+1} = f^i(X^i_t, U^i_t, Z_t, W^i_t)$
Mean-field	$Z_t = \frac{1}{n}\sum_{i=1}^{n} \delta_{X^i_t}$

Info structure	$I^i_t = \{X^i_t, Z_{1:t}\}$

- Interesting model for applications with large population of a few types of agents
- Smart grids, IoT, . . .

Arabneydi and Mahajan, "Team optimal control of coupled subsystems with mean field sharing," CDC 2013.

Mean-field teams

Dynamics	$X^i_{t+1} = f^i(X^i_t, U^i_t, Z_t, W^i_t)$
Mean-field	$Z_t = \frac{1}{n}\sum_{i=1}^{n} \delta_{X^i_t}$

Info structure	$I^i_t = \{X^i_t, Z_{1:t}\}$

- Interesting model for applications with large population of a few types of agents
- Smart grids, IoT, . . .

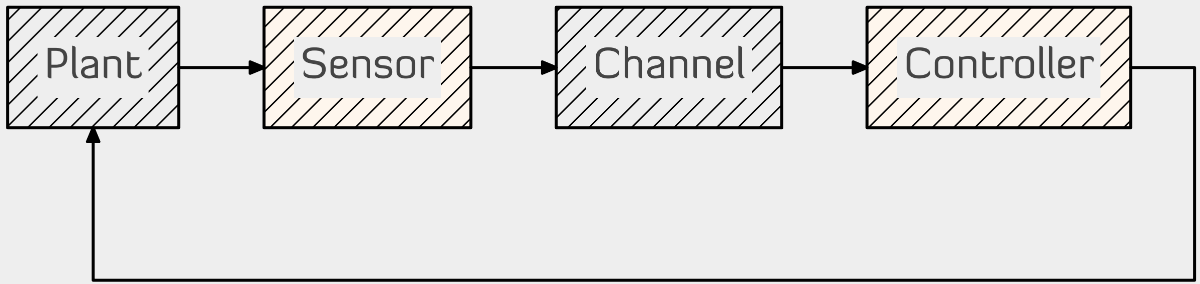
Use common information approach

- Using ideas from exchangeable Markov chains show that

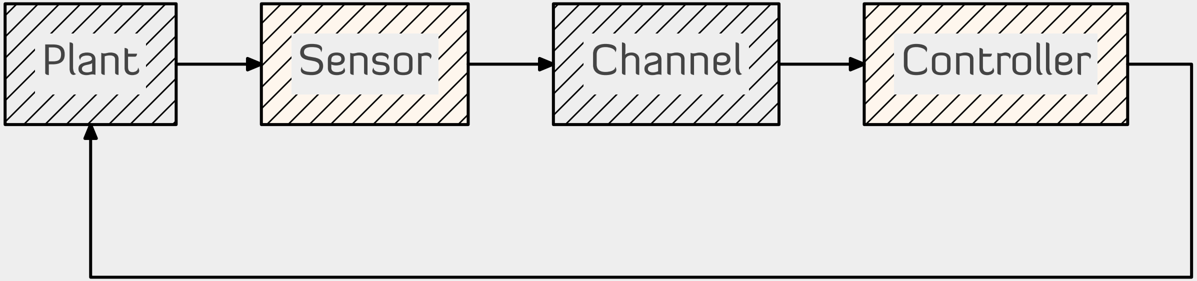
  Suff statistic for $Z_{1:t} = Z_t$

Arabneydi and Mahajan, "Team optimal control of coupled subsystems with mean field sharing," CDC 2013.

Networked control over wireless channels

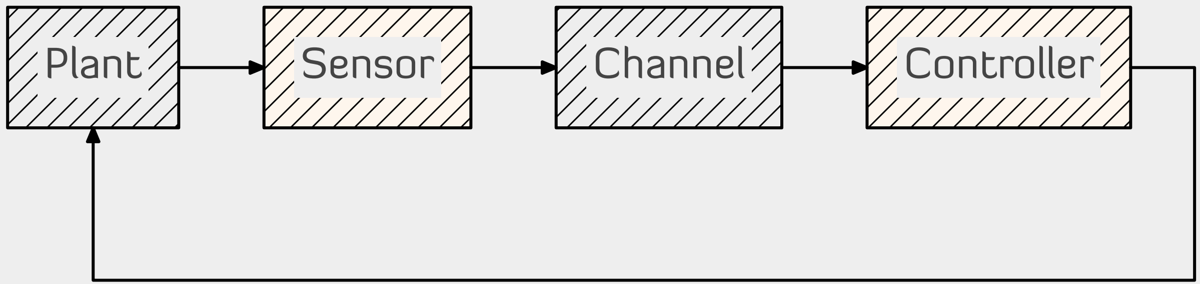


Networked control over wireless channels



Dyanmics $x_{t+1} = Ax_t + Bu_t + w_t$

Networked control over wireless channels

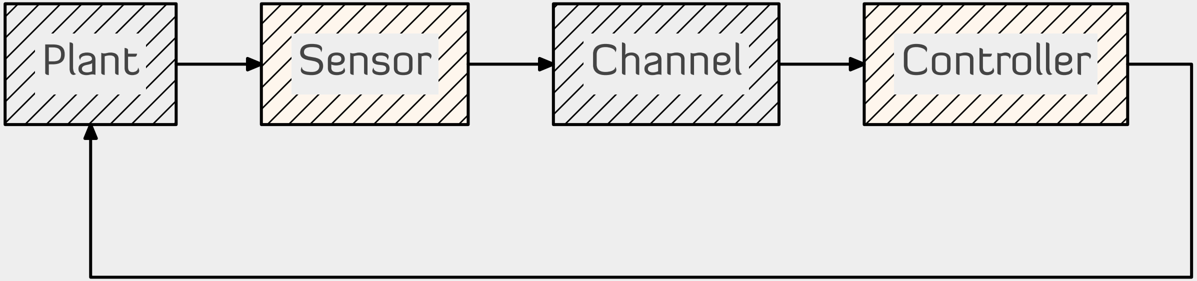


Dyanmics $x_{t+1} = Ax_t + Bu_t + w_t$

Wireless Channel

- Sensor sends a packet to the controller using power level $p_t \in \mathcal{P}$.
- Packet is dropped with probability $q(p_t)$, which is decreasing in p_t.
- TCP-like transport layer protocol, so sensor knows when packet is dropped.

Networked control over wireless channels



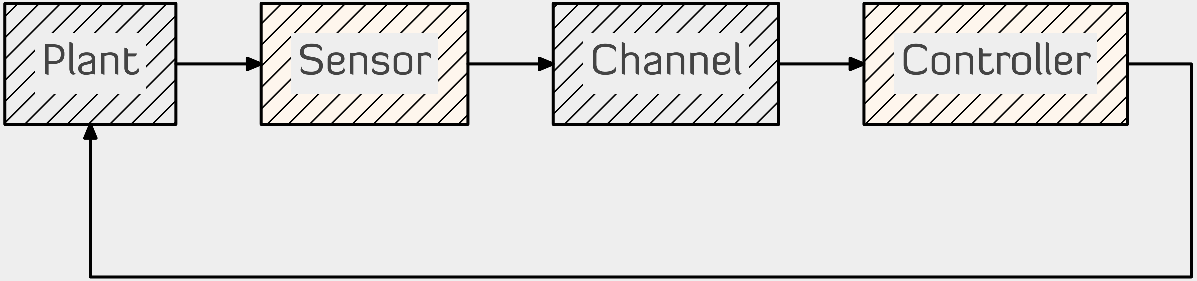
Dyanmics: $x_{t+1} = Ax_t + Bu_t + w_t$

Wireless Channel: $$y_t = \begin{cases} x_t & \text{w.p. } 1 - q(p_t) \\ \mathfrak{E} & \text{w.p. } q(p_t) \end{cases}$$

Wireless Channel

- Sensor sends a packet to the controller using power level $p_t \in \mathcal{P}$.
- Packet is dropped with probability $q(p_t)$, which is decreasing in p_t.
- TCP-like transport layer protocol, so sensor knows when packet is dropped.

Networked control over wireless channels

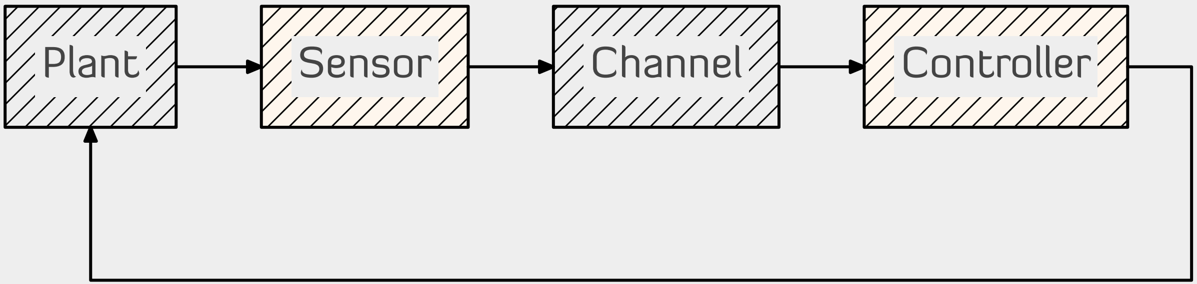


Dyanmics	$x_{t+1} = Ax_t + Bu_t + w_t$
Wireless Channel	$y_t = \begin{cases} x_t & \text{w.p. } 1 - q(p_t) \\ \mathfrak{E} & \text{w.p. } q(p_t) \end{cases}$
Information Structure	$I^s_t = \{x_{1:t}, y_{1:t-1}, u_{1:t-1}\}$ $I^c_t = \{y_{1:t}, u_{1:t-1}\}$

Wireless Channel

- Sensor sends a packet to the controller using power level $p_t \in \mathcal{P}$.
- Packet is dropped with probability $q(p_t)$, which is decreasing in p_t.
- TCP-like transport layer protocol, so sensor knows when packet is dropped.

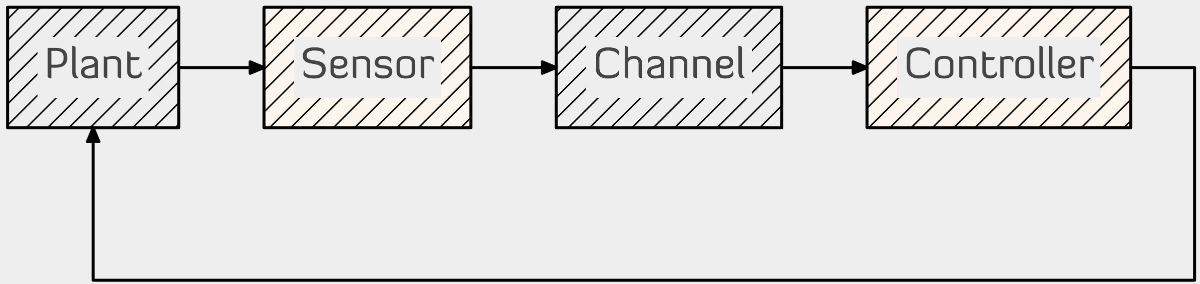
Networked control over wireless channels



Dyanmics	$x_{t+1} = Ax_t + Bu_t + w_t$
Wireless Channel	$y_t = \begin{cases} x_t & \text{w.p. } 1 - q(p_t) \\ \mathfrak{E} & \text{w.p. } q(p_t) \end{cases}$
Information Structure	$I^s_t = \{x_{1:t}, y_{1:t-1}, u_{1:t-1}\}$ $I^c_t = \{y_{1:t}, u_{1:t-1}\}$

Decision strategies	$p_t = f_t(I^s_t), \quad u_t = g_t(I^c_t).$

Networked control over wireless channels



Dyanmics: $x_{t+1} = Ax_t + Bu_t + w_t$

Wireless Channel: $y_t = \begin{cases} x_t & \text{w.p. } 1 - q(p_t) \\ \mathfrak{E} & \text{w.p. } q(p_t) \end{cases}$

Information Structure: $I^s_t = \{x_{1:t}, y_{1:t-1}, u_{1:t-1}\}$

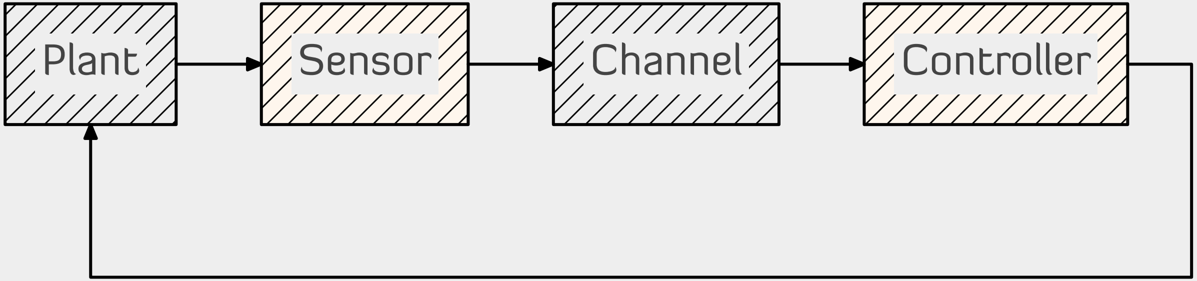
$I^c_t = \{y_{1:t}, u_{1:t-1}\}$

Decision strategies: $p_t = f_t(I^s_t), \quad u_t = g_t(I^c_t).$

Per-step cost: $x_t^\top Q x_t + u_t^\top R u_t + \lambda(p_t)$

Control cost + comm. cost

Networked control over wireless channels



Dyanmics	$x_{t+1} = Ax_t + Bu_t + w_t$
Wireless Channel	$y_t = \begin{cases} x_t & \text{w.p. } 1 - q(p_t) \\ \mathfrak{E} & \text{w.p. } q(p_t) \end{cases}$
Information Structure	$I^s_t = \{x_{1:t}, y_{1:t-1}, u_{1:t-1}\}$ $I^c_t = \{y_{1:t}, u_{1:t-1}\}$

Decision strategies	$p_t = f_t(I^s_t), \quad u_t = g_t(I^c_t).$
Per-step cost	$x_t^\intercal Q x_t + u_t^\intercal R u_t + \lambda(p_t)$ Control cost + comm. cost
Objective	$J(f, g) = \mathbb{E}\Big[\sum_{t=1}^{T} c(x_t, u_t, p_t)\Big]$

Conceptual difficulties

Packet-drop is a non-linearlity

- The closed loop system is non-linear. Choice of optimal control strategy is not obvious.

Conceptual difficulties

Packet-drop is a non-linearlity

- The closed loop system is non-linear. Choice of optimal control strategy is not obvious.

Dual effect of control

- For a fixed transmission strategy, the innovation at the controller depends on controller's strategy.
- Not obvious if there is separation of estimation and control.

Conceptual difficulties

Packet-drop is a non-linearlity

- The closed loop system is non-linear. Choice of optimal control strategy is not obvious.

Dual effect of control

- For a fixed transmission strategy, the innovation at the controller depends on controller's strategy.
- Not obvious if there is separation of estimation and control.

Sensor can use power-levels to signal information

- As an example, suppose $\mathcal{P} = \{0, 1\}$, with $q(0) = 1$ and $q(1) = 0$. If the controller doesn't receive a packet, it knows that the state lied in the set where the transmitter chooses $p = 0$.
- Related to real-time communication (a notoriously difficult problem).

Common-info based solutions to NCS

Large literature on these models

- Using the common-info based dynamic program, prove that there are optimal transmission strategies that don't depend on the control strategy.
- Highly non-trivial because the state space of the DP is belief valued; the action space is function valued.
- Implication: there is **no dual effect** and there is separation of estimation and control.
- Note that there is no contradiction. Under an arbitrary policy, control has a dual effect; under the optimal policy it doesn't.

Rabi, Moustakides, and Baras, "Adaptive sampling for linear state estimation," SICON 2012.
Lipsa and Martins, "Remote state estimation with communication costs for first order LTI systems," TAC 2011.
Molin and Hirsche, "Event triggered state estimation: An iterative algorithm and optimality properties," TAC 2017.
Chakravorty and Mahajan, "Remote estimation over a packet-drop channel with Markovian state" TAC 2020.

Common information based approach to linear systems

LQ system	Linear dynamics and quadratic cost

PHS Info structure	$I^i_t = \{C_t, L^i_t\}$

Mahajan and Nayyar, "Sufficient statistics for linear control strategies in decentralized systems with partial history sharing", TAC 2015

Afshari and Mahajan, "Decentralized linear quadratic systems with major and minor agents and non-Gaussian noise", TAC 2023

Common information based approach to linear systems

LQ system	Linear dynamics and quadratic cost

PHS Info structure	$I^i_t = \{C_t, L^i_t\}$

Structure of optimal strategies	▷ $U^i_t = K^c_t \hat{S}_{t\|c} + K^i_t L^i_t$ where $\hat{S}_{t\|c} = \mathbb{E}[X_t, L^1_t, L^2_t \mid C_t, \gamma^1_{1:t-1}, \gamma^2_{1:t-1}]$.

Mahajan and Nayyar, "Sufficient statistics for linear control strategies in decentralized systems with partial history sharing", TAC 2015

Afshari and Mahajan, "Decentralized linear quadratic systems with major and minor agents and non-Gaussian noise", TAC 2023

Common information based approach to linear systems

LQ system	Linear dynamics and quadratic cost

PHS Info structure	$I^i_t = \{C_t, L^i_t\}$

Structure of optimal strategies

- $U^i_t = K^c_t \hat{S}_{t|c} + K^i_t L^i_t$

  where $\hat{S}_{t|c} = \mathbb{E}[X_t, L^1_t, L^2_t \mid C_t, \gamma^1_{1:t-1}, \gamma^2_{1:t-1}]$.

How to compute optimal gains?

- Framework based on fundamental ideas of linear systems:
  - State splitting • Completion of squares
  - Orthogonal projection • Conditional independence
- Noise need not be Gaussian. Identify optimal (possibly non-linear) controllers or best linear controllers.

Mahajan and Nayyar, "Sufficient statistics for linear control strategies in decentralized systems with partial history sharing", TAC 2015

Afshari and Mahajan, "Decentralized linear quadratic systems with major and minor agents and non-Gaussian noise", TAC 2023

Common information resolves conceptual difficulties in decentralized control

Common information resolves conceptual difficulties in decentralized control

But, this is a workshop on learning and control

Learning in dynamic teams

Implications of common-info approach

- Converts planning in multi-agent teams to a POMDP
- In the learning setting, use your favorite RL algo for POMDP at the coordinator (offline traning) or each agent's local copy of the coordinator (online training)
- Beautiful theory ... doesn't work in practice.
- Too complicated. The action space is too large.

Learning in dynamic teams

Implications of common-info approach

- Converts planning in multi-agent teams to a POMDP
- In the learning setting, use your favorite RL algo for POMDP at the coordinator (offline traning) or each agent's local copy of the coordinator (online training)
- Beautiful theory ... doesn't work in practice.
- Too complicated. The action space is too large.

Practical MARL algorithms

- Many SOTA MARL algos build on the common-info approach

  BAD (Bayesian action decoder), SOTA on Hannabi

  CAPI (cooperative approximate policy iteration), SOTA on Tiny-Bridge

  . . .

Learning in dynamic teams

Implications of common-info approach

- Converts planning in multi-agent teams to a POMDP
- In the learning setting, use your favorite RL algo for POMDP at the coordinator (offline traning) or each agent's local copy of the coordinator (online training)
- Beautiful theory … doesn't work in practice.
- Too complicated. The action space is too large.

Practical MARL algorithms

- Many SOTA MARL algos build on the common-info approach

  BAD (Bayesian action decoder), SOTA on Hannabi

  CAPI (cooperative approximate policy iteration), SOTA on Tiny-Bridge

  . . .

But no theory! How do we develop RL theory MARL?

Tentative Roadmap for MARL Theory

Step 1 RL for POMDPs	
	▷ Simplest "MARL" environment. Theory still lacking. ▷ Our recent results (AIS theory) that resolve key conceptual challenges ▷ Generalizes to MARL seeting using common-info approach but . . .

Tentative Roadmap for MARL Theory

Step 1
RL for POMDPs

- Simplest "MARL" environment. Theory still lacking.
- Our recent results (AIS theory) that resolve key conceptual challenges
- Generalizes to MARL seeting using common-info approach but . . .

Step 2
Centralized vs decentralized training

- Most MARL algos use centralized training.
- Some recent preliminary results for analysis of centralized training.
- Some empirical results on decentralized training.

Tentative Roadmap for MARL Theory

Step 1
RL for POMDPs

- Simplest "MARL" environment. Theory still lacking.
- Our recent results (AIS theory) that resolve key conceptual challenges
- Generalizes to MARL seeting using common-info approach but . . .

Step 2
Centralized vs decentralized training

- Most MARL algos use centralized training.
- Some recent preliminary results for analysis of centralized training.
- Some empirical results on decentralized training.

Next Steps

- Credit assignment (among agents)
- Agents helping each other to learning

- email: aditya.mahajan@mcgill.ca
- web: https://adityam.github.io

Thank you