Measure and cost dependent properties of information strucutres

Aditya Mahajan
Yale University

Serdar Yüksel
Queen's University

ACC 2010

Why are information structures useful?

Why are information structures useful?

- Info structures capture the design difficulties of decentralized control

Why are information structures useful?

- Info structures capture the design difficulties of decentralized control
- Classical info structures are centralized systems, hence easy to design
- Non-classical info structures are decentralized systems, hence hard to design

Why are information structures useful?

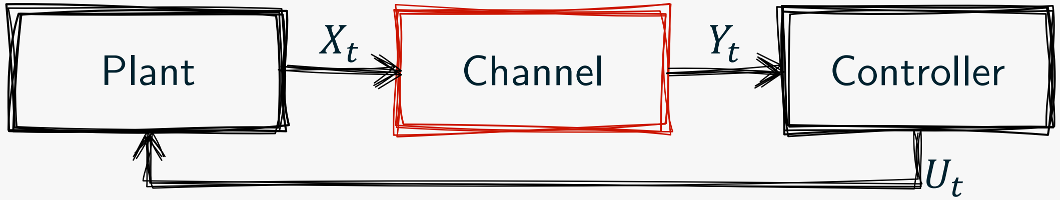
- Info structures capture the design difficulties of decentralized control
- Classical info structures are centralized systems, hence easy to design
- Non-classical info structures are decentralized systems, hence hard to design

Is this really true? Can we have two systems with identical information structures that behave differently?

A controller with no memory

- State Equation: $X_{t+1} = f_t(X_t, U_t, W_t)$
- Observation Equation: $Y_t = h_t(X_t, N_t)$
- Controller with no memory: $U_t = g_t(Y_t)$

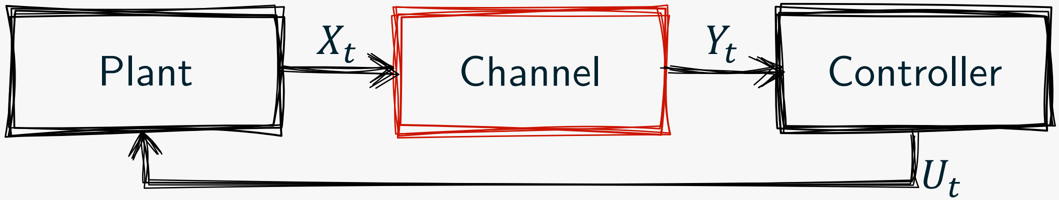
A controller with no memory



- State Equation: $X_{t+1} = f_t(X_t, U_t, W_t)$
- Observation Equation: $Y_t = h_t(X_t, N_t)$
- Controller with no memory: $U_t = g_t(Y_t)$

Non-classical info structure

A controller with no memory

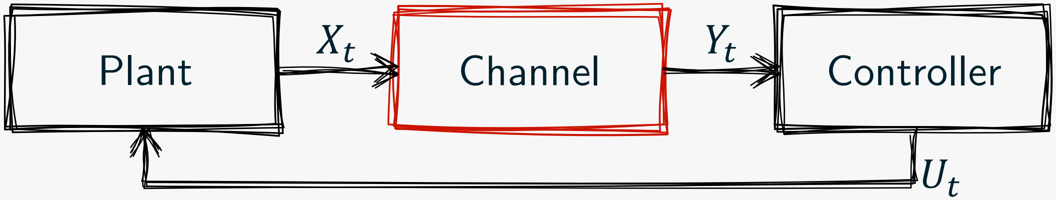


- State Equation: $X_{t+1} = f_t(X_t, U_t, W_t)$
- Observation Equation: $Y_t = h_t(X_t, N_t)$
- Controller with no memory: $U_t = g_t(Y_t)$

Non-classical info structure

- The info structure does not depend on channel h_t

A controller with no memory



- State Equation: $X_{t+1} = f_t(X_t, U_t, W_t)$
- Observation Equation: $Y_t = h_t(X_t, N_t)$
- Controller with no memory: $U_t = g_t(Y_t)$

Non-classical info structure

- The info structure does not depend on channel h_t

When the channel is noiseless, the system is an MDP --- a centralized system

A controller with no memory

Two systems with identical info structures

Perfect observations ⇒ centralized
Imperfect observations ⇒ decentralized

When the channel is noiseless, the system is an MDP --- a centralized system

What is missing?

Information structures do not completely characterize the design difficulties of decentralized systems

What is missing?

Information structures do not completely characterize the design difficulties of decentralized systems

Information structures capture who knows what and when, but do not capture usefulness of available data

What is missing?

Information structures do not completely characterize the design difficulties of decentralized systems

Information structures capture who knows what and when, but do not capture usefulness of available data

We present a generalization of information structures, which we call P-generalization, that captures the usefulness of information.

This generalization depends on the coupling of the cost function and the independence properties of the probability measure

Contributions of the paper

- Defined a P-generalization of an info structure
  The solution technique for any info structure is also applicable to its P-generalization

Contributions of the paper

- Defined a P-generalization of an info structure
  The solution technique for any info structure is also applicable to its P-generalization

- Implications: Follow a two step approach
  - Define info structure in the usual manner (keeps analysis simple)
  - Define the P-generalization of an info structure
    We get the solution technique for P-generalized info structure for free!

Contributions of the paper

- Defined a P-generalization of an info structure
  The solution technique for any info structure is also applicable to its P-generalization
- Implications: Follow a two step approach
  - Define info structure in the usual manner (keeps analysis simple)
  - Define the P-generalization of an info structure
    We get the solution technique for P-generalized info structure for free!
- Present coupled dynamic programs to find pbpo solution of quasiclassical info structures
  - Works for non-linear systems
  - Need to only solve parametric optimization problem

Outline of the paper

- Model
- Information Structures
- P-generalization of info structures
- Coupled dynamic programs for quasiclassical info structure
- Example

The intrinsic model

Originally proposed by Witsenhausen, 1971 and 1975

The intrinsic model

Originally proposed by Witsenhausen, 1971 and 1975

- Intrinsic event: ω taking values in a probability space

The intrinsic model

Originally proposed by Witsenhausen, 1971 and 1975

- Intrinsic event: ω taking values in a probability space
- N agents

The intrinsic model

Originally proposed by Witsenhausen, 1971 and 1975

- Intrinsic event: ω taking values in a probability space
- N agents
- Observations of agent n: Y_n taking value in a measurable space

$$Y_n = f_n(\omega, U_{D_n})$$

  where $D_n \subset [n-1]$

The intrinsic model

Originally proposed by Witsenhausen, 1971 and 1975

- Intrinsic event: ω taking values in a probability space
- N agents
- Observations of agent n: Y_n taking value in a measurable space

$$Y_n = f_n(\omega, U_{D_n})$$

  where $D_n \subset [n-1]$

- Action of agent n: U_n taking values in a measurable space

$$U_n = g_n(Y_n)$$

The intrinsic model

Originally proposed by Witsenhausen, 1971 and 1975

- Intrinsic event: ω taking values in a probability space
- N agents
- Observations of agent n: Y_n taking value in a measurable space

$$Y_n = f_n(\omega, U_{D_n})$$

  where $D_n \subset [n-1]$
- Action of agent n: U_n taking values in a measurable space

$$U_n = g_n(Y_n)$$

- Cost: Additive terms. Agents coupled by k-th cost term: $C_k \subset [N]$

$$\sum_{k=1}^{K} \rho_k(\omega, U_{C_k})$$

The intrinsic model

Originally proposed by Witsenhausen, 1971 and 1975

- Intrinsic event: ω taking values in a probability space
- N agents
- Observations of agent n: Y_n taking value in a measurable space

$$Y_n = f_n(\omega, U_{D_n})$$

  where $D_n \subset [n-1]$
- Action of agent n: U_n taking values in a measurable space

$$U_n = g_n(Y_n)$$

- Cost: Additive terms. Agents coupled by k-th cost term: $C_k \subset [N]$

$$\sum_{k=1}^{K} \rho_k(\omega, U_{C_k})$$

- Objective: Choose $(g_1, \ldots, g_N)$ to minimize expected cost

Salient Features

Agents are coupled in two ways:

- Coupling through dynamics
  - D_n^*: set of agents that can influence the observations of agent n
  - $m \in D_n^* \Rightarrow$ there exist $m = m_0, m_1, \dots, m_\ell = n$ such that

$$m_{i-1} \in D_{m_i}, \quad i = 1, \dots, \ell$$

- Coupling through cost

  C_n^*: agents coupled to agent n through cost

$$C_n^* = \bigcup_{k=1}^{K} C_k \mathbb{1}\{n \in C_k\}$$

Information Structures

- Information Structure

  Collection of information known to each agent

Information Structures

- **Information Structure**

  Collection of information known to each agent

- **Classification of info structures**

  - Classical info structure
    Each agent knows the data available to all agents that act before it

  - Quasiclassical info structure
    Each agent knows the data available to all agents that can influence its observation

Information Structures

- Information Structure

  Collection of information known to each agent

- Classification of info structures

  - Classical info structure
    Each agent knows the data available to all agents that act before it

  - Quasiclassical info structure
    Each agent knows the data available to all agents that can influence its observation

  - Strictly classical info structures
    Each agent . . . data and control actions . . .

  - Strictly quasiclassical info structure
    Each agent . . . data and control actions . . .

Expansion of info structures

- Classical expansion of info structure

A new system obtained by

$$Y_n \mapsto (Y_n, Y_{[n-1]}, U_{[n-1]})$$

Expansion of info structures

- Classical expansion of info structure

  A new system obtained by

$$Y_n \mapsto (Y_n, Y_{[n-1]}, U_{[n-1]})$$

- Quasiclassical expansion of info structure

  A new system obtained by

$$Y_n \mapsto (Y_n, Y_{D_n^*}, U_{D_n^*})$$

The main idea (1)

Dynamic programming works only for strictly classical info structure.

The main idea (1)

Dynamic programming works only for strictly classical info structure.
Nevertheless, we can design for classical info structure (not strict) as follows:

The main idea (1)

Dynamic programming works only for strictly classical info structure.
Nevertheless, we can design for classical info structure (not strict) as follows:

- Denote the classical system by C.

The main idea (1)

Dynamic programming works only for strictly classical info structure. Nevertheless, we can design for classical info structure (not strict) as follows:

- Denote the classical system by C.
- Let SC be the classical expansion of C. SC is strictly classical.
- Find optimal policy g for SC (using dynamic programing)

The main idea (1)

Dynamic programming works only for strictly classical info structure.
Nevertheless, we can design for classical info structure (not strict) as follows:

- Denote the classical system by C.
- Let SC be the classical expansion of C. SC is strictly classical.
- Find optimal policy g for SC (using dynamic programing)
- The difficulty is that g may not be implementable in C

The main idea (1)

Dynamic programming works only for strictly classical info structure.
Nevertheless, we can design for classical info structure (not strict) as follows:

- Denote the classical system by C.
- Let SC be the classical expansion of C. SC is strictly classical.
- Find optimal policy g for SC (using dynamic programing)
- The difficulty is that g may not be implementable in C
- By successive substitution, we can find a corresponding policy g^* such that
  - g and g^* have the same performance in SC
  - g^* is implementable in C

The main idea (1)

Dynamic programming works only for strictly classical info structure.
Nevertheless, we can design for classical info structure (not strict) as follows:

- Denote the classical system by C.
- Let SC be the classical expansion of C. SC is strictly classical.
- Find optimal policy g for SC (using dynamic programing)
- The difficulty is that g may not be implementable in C
- By successive substitution, we can find a corresponding policy g^* such that
  - g and g^* have the same performance in SC
  - g^* is implementable in C

Question: Instead of a classical system, can we start with a more relaxed system such that this procedure still works?

The main idea (1)

Dynamic programming works only for strictly classical info structure.
Nevertheless, we can design for classical info structure (not strict) as follows:

P-classical info structure:

Let $Q_n := \sum_{k=1}^{K} \rho_k(\omega, U_{C_k}) \mathbb{1}\{\{n \in C_k\} \cup \{\exists m \in C_k : n \in D_m^*\}\}$.

Then, an info structure is P-classical if

$$\mathbb{E}\{Q_n \mid Y_n, U_n\} = \mathbb{E}\{Q_n \mid Y_{[n]}, U_{[n]}\}$$

Question: Instead of a classical system, can we start with a more relaxed system such that this procedure still works?

The main idea (2)

We ask a similar question for quasiclassical info structures.

- What is the most relaxed info structure that we can start with such that
  - if we take its quasiclassical expansion
  - find the optimal policy for the quasiclassical expansion
  - then, can find a corresponding optimal policy that is implementable in the original system

The main idea (2)

We ask a similar question for quasiclassical info structures.

- What is the most relaxed info structure that we can start with such that
  - if we take its quasiclassical expansion
  - find the optimal policy for the quasiclassical expansion
  - then, can find a corresponding optimal policy that is implementable in the original system
- Difficulty: No appropriate solution technique for quasiclassical systems
  - Solutions for LQG quasiclassical systems rely convexity of static LQG teams. These results do not extend to non-LQG systems.
  - Sequential decomposition for optimal design gives a functional optimization problem. This makes it extremely hard to find a corresponding policy
    (revisit later)

The main idea (2)

We ask a similar question for quasiclassical info structures.

- What is the most relaxed info structure that we can start with such that
  - if we take its quasiclassical expansion
  - find the optimal policy for the quasiclassical expansion
  - then, can find a corresponding optimal policy that is implementable in the original system
- Difficulty: No appropriate solution technique for quasiclassical systems
  - Solutions for LQG quasiclassical systems rely convexity of static LQG teams. These results do not extend to non-LQG systems.
  - Sequential decomposition for optimal design gives a functional optimization problem. This makes it extremely hard to find a corresponding policy
  - Find pbpo solutions using coupled dynamic programs (revisit later)

The main idea (2)

We ask a similar question for quasiclassical info structures.

- What is the most relaxed info structure that we can start with such that

P-quasiclassical info structure:

Let $Q_n := \sum_{k=1}^{K} \rho_k(\omega, U_{C_k}) \mathbb{1}\{\{n \in C_k\} \cup \{\exists m \in C_k : n \in D^*_m\}\}$.

Then, an info structure is P-quasiclassical if

$$\mathbb{E}\{Q_n \mid Y_n, U_n\} = \mathbb{E}\{Q_n \mid Y_n, U_n, Y_{D^*_n}, U_{D^*_n}\}$$

  - Sequential decomposition for optimal design gives a functional optimization problem. This makes it extremely hard to find a corresponding policy
  - Find pbpo solutions using coupled dynamic programs (revisit later)

Proof outline

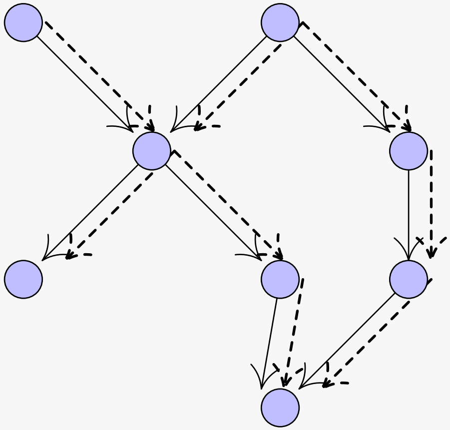
- The proof for both cases is constructive
  - Take expanded info structure
  - Find an optimal (or pbpo) policy
  - Construct a corresponding policy that is implementable in original system
- The details of each step conceptually simple, but notationally cumbersome due to generality of the model

Coupled Dynamic programs for quasiclassical info structure

Any quasiclassical system can be broken into a collection of coupled systems where each subsystem has a classical info structure

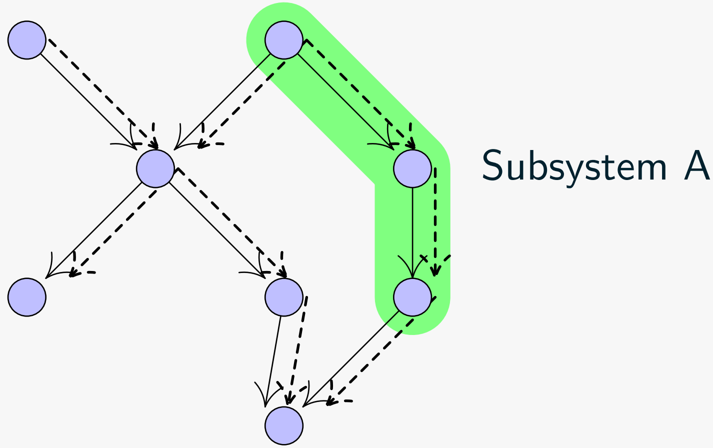
Coupled Dynamic programs for quasiclassical info structure

Any quasiclassical system can be broken into a collection of coupled systems where each subsystem has a classical info structure



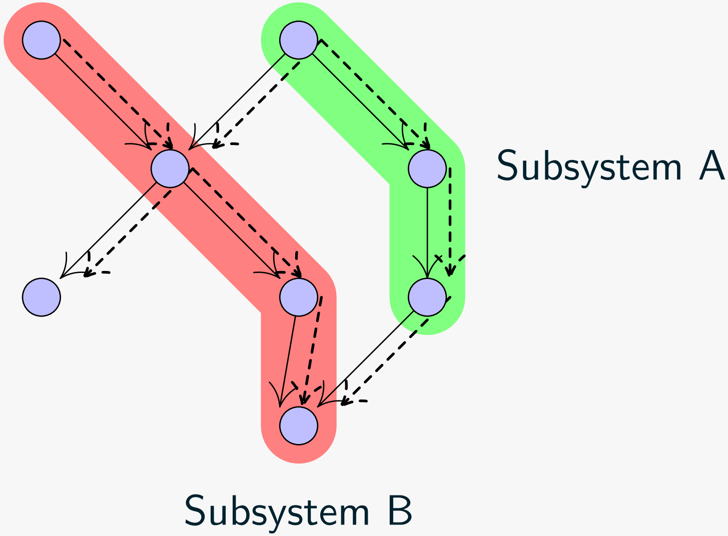
Coupled Dynamic programs for quasiclassical info structure

Any quasiclassical system can be broken into a collection of coupled systems where each subsystem has a classical info structure



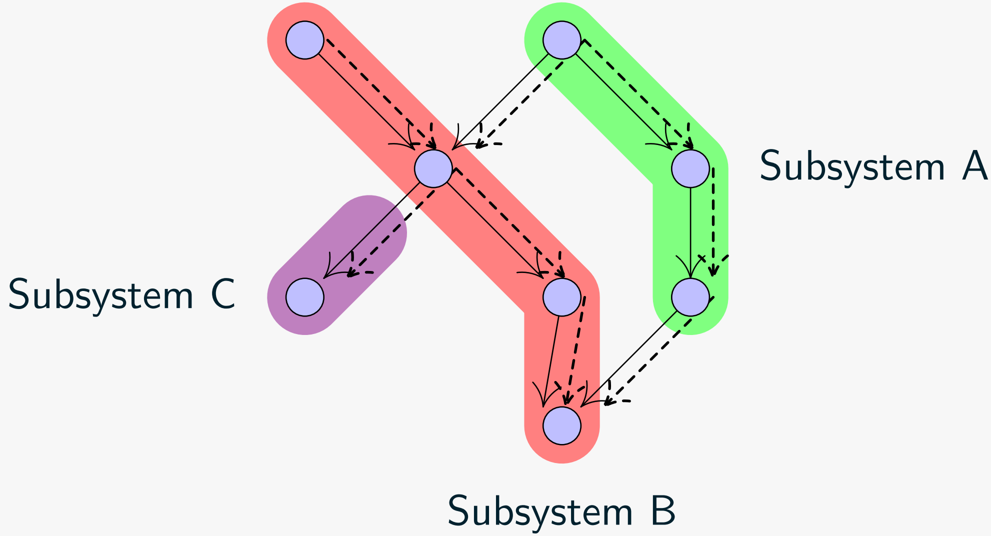
Coupled Dynamic programs for quasiclassical info structure

Any quasiclassical system can be broken into a collection of coupled systems where each subsystem has a classical info structure



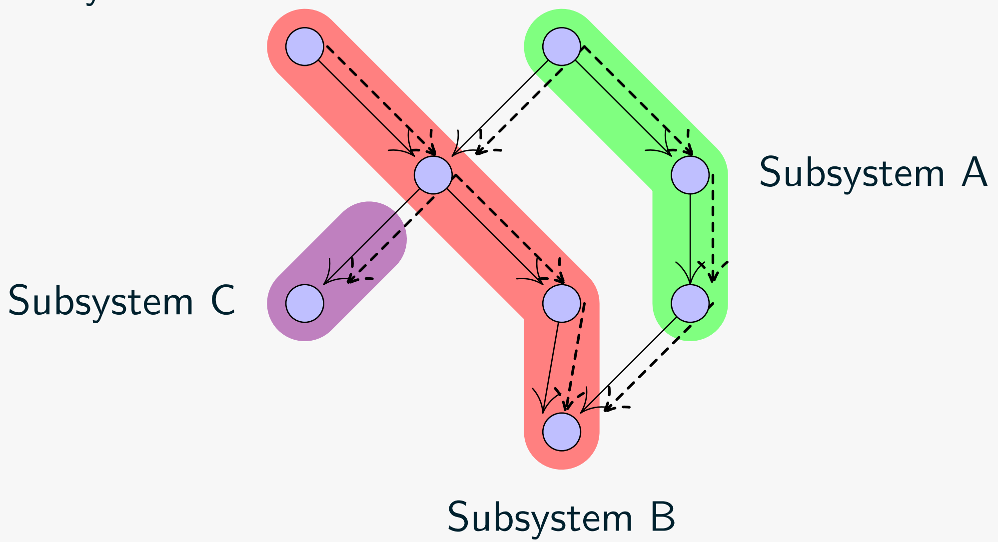
Coupled Dynamic programs for quasiclassical info structure

Any quasiclassical system can be broken into a collection of coupled systems where each subsystem has a classical info structure



Coupled Dynamic programs for quasiclassical info structure

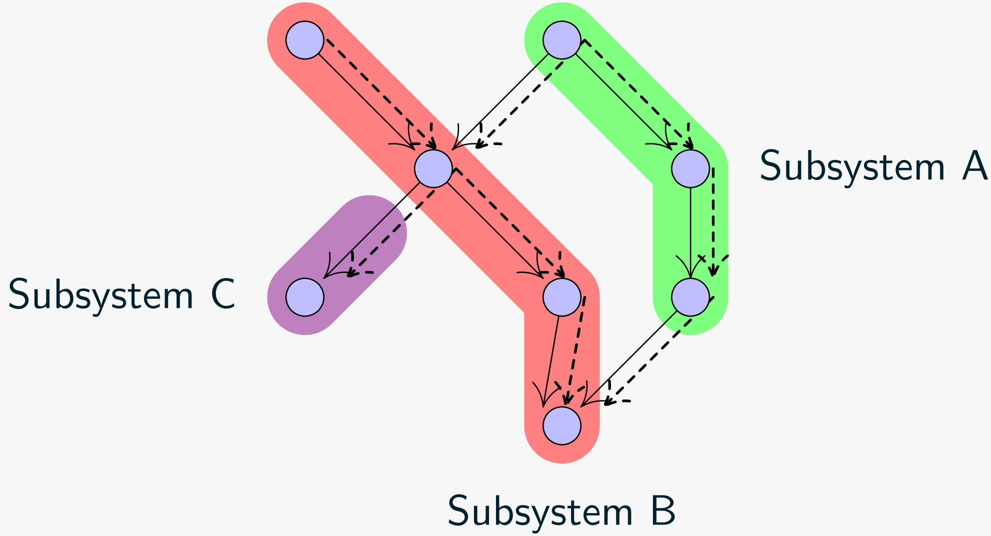
Any quasiclassical system can be broken into a collection of coupled systems where each subsystem has a classical info structure



- Subsystems A, B, and C are classical

Coupled Dynamic programs for quasiclassical info structure

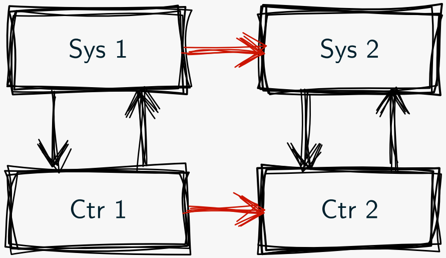
Any quasiclassical system can be broken into a collection of coupled systems where each subsystem has a classical info structure



- Subsystems A, B, and C are classical
- Write a DP for each subsystem and solve them iteratively

Idea originally proposed in Teneketzis and Ho, 1987

An Example



$$x^1_{t+1} = f^1(x^1_t, u^1_t, w^1_t) \qquad x^2_{t+1} = f^2(x^1_t, x^2_t, u^2_t, w^2_t)$$

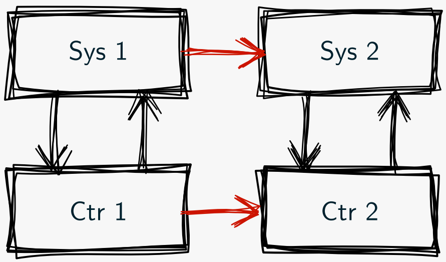
$$y^1_t = h^1(x^1_t, n^1_t) \qquad y^2_t = h^2(x^2_t, n^2_t)$$

$$u^1_t = g^1_t(y^1_{[t]}, u^1_{[t-1]}) \qquad u^2_t = g^2_t(y^1_{[t]}, y^2_{[t]}, u^1_{[t-1]}, u^2_{[t-1]})$$

Choose $G^1 := (g^1_1, \dots, g^1_T)$ and $G^2 := (g^2_1, \dots, g^2_T)$ to minimize

$$\mathbb{E}\left\{\sum_{t=1}^{T} \rho(x^1_t, x^2_t, u^1_t, u^2_t)\right\}$$

An Example



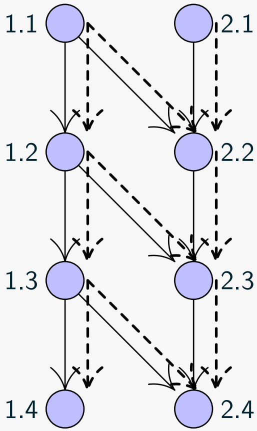
$$x^1_{t+1} = f^1(x^1_t, u^1_t, w^1_t) \qquad x^2_{t+1} = f^2(x^1_t, x^2_t, u^2_t, w^2_t)$$
$$y^1_t = h^1(x^1_t, n^1_t) \qquad y^2_t = h^2(x^2_t, n^2_t)$$
$$u^1_t = g^1_t(y^1_{[t]}, u^1_{[t-1]}) \qquad u^2_t = g^2_t(y^1_{[t]}, y^2_{[t]}, u^1_{[t-1]}, u^2_{[t-1]})$$

Choose $G^1 := (g^1_1, \dots, g^1_T)$ and $G^2 := (g^2_1, \dots, g^2_T)$ to minimize

$$\mathbb{E}\left\{ \sum_{t=1}^{T} \rho(x^1_t, x^2_t, u^1_t, u^2_t) \right\}$$

- Quasiclassical info structure
- Non-linear dynamics
- Noisy observations

An Example



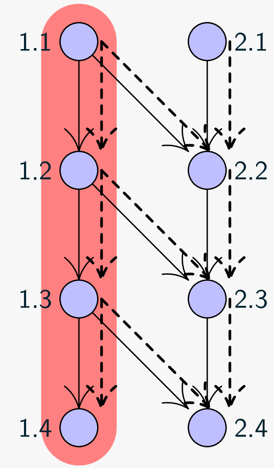
An Example

- Subsystem 1

Fix policy G^2 and solve for G^1

$$V_T^1(y_{[T]}^1, u_{[T-1]}^1) = \mathbb{E}\Big\{\rho(x_T^1, x_T^2, u_T^1, u_T^2)\Big| y_{[T]}^1, u_{[T-1]}^1\Big\}$$

$$V_t^2(y_{[t]}^1, u_{[t-1]}^1) = \mathbb{E}\Big\{\rho(x_t^1, x_t^2, u_t^1, u_t^2) + V_{t+1}^1(y_{[t+1]}^1, u_{[t]}^1)\Big| y_{[t]}^1, u_{[t-1]}^1\Big\}$$



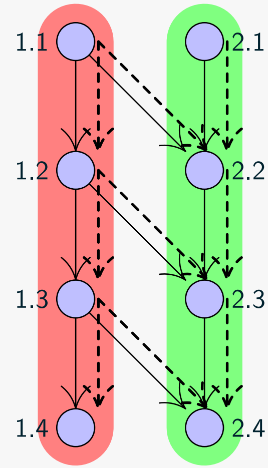
An Example

- **Subsystem 1**

  Fix policy G^2 and solve for G^1

$$V_T^1(y_{[T]}^1, u_{[T-1]}^1) = \mathbb{E}\Big\{\rho(x_T^1, x_T^2, u_T^1, u_T^2)\Big| y_{[T]}^1, u_{[T-1]}^1\Big\}$$

$$V_t^2(y_{[t]}^1, u_{[t-1]}^1) = \mathbb{E}\Big\{\rho(x_t^1, x_t^2, u_t^1, u_t^2) + V_{t+1}^1(y_{[t+1]}^1, u_{[t]}^1)\Big| y_{[t]}^1, u_{[t-1]}^1\Big\}$$



- **Subsystem 2**

  Fix policy G^1 and solve for G^2

$$V_T^2(y_{[T]}^1, y_{[T]}^2, u_{[T-1]}^1 u_{[T-1]}^2) = \mathbb{E}\Big\{\rho(x_T^2, x_T^2, u_T^2, u_T^2)\Big| y_{[T]}^1, y_{[T]}^2, u_{[T-1]}^1, u_{[T-1]}^2\Big\}$$

$$V_t^2(y_{[t]}^1, y_{[t]}^2, u_{[t-1]}^1, u_{[t-1]}^2) = \mathbb{E}\Big\{\rho(x_t^2, x_t^2, u_t^2, u_t^2) + V_{t+1}^2(y_{[t+1]}^1, y_{[t+1]}^2, u_{[t]}^1, u_{[t]}^2)\Big| y_{[t]}^1, y_{[t]}^2, u_{[t-1]}^1, u_{[t-1]}^2\Big\}$$

Conclusion

- Defined a P-generalization of info structure

> The solution technique for any info structure
> is also applicable to its P-generalization

- Present coupled dynamic programs to find person by person optimal solution of quasiclassical info structures

Thank you