A General Architecture for Decentralized Supervisory Control of Discrete-Event Systems

Tae-Sic Yoo and Stéphane Lafontune
Dept. of Electrical Engineering and Computer Science
University of Michigan, Ann Arbor, MI.
E-mail: {tyoo, stephane}@eecs.umich.edu
Outline of Presentation

◆ Revisiting Existing Architecture

◆ The Disjunctive Architecture for Decentralized Control
  ✷ Supervisor Existence Condition
  ✷ Incomparable with the existing conventional architecture

◆ The General Architecture for Decentralized Control
  ✷ Supervisor Existence Condition
  ✷ Better than the existing conventional architecture
  ✷ Computational complexity Results
  ✷ Some Supervisor Synthesis Results

◆ Concluding Remarks
What is the General Goal?

**Goal:** Achieve the desired behavior!!
Formalism

\[ \Sigma = \Sigma_c \cup \Sigma_{uc}, \quad \Sigma_c: \text{Controllable events}, \quad \Sigma_{uc}: \text{Uncontrollable events} \]
\[ \Sigma = \Sigma_o \cup \Sigma_{uo}, \quad \Sigma_o: \text{Observable events}, \quad \Sigma_{uo}: \text{Unobservable events} \]
\[ P: \Sigma^* \rightarrow \Sigma_o, \]
\[ S_p: \Sigma_o^* \rightarrow \Gamma := \{ \gamma \in 2^\Sigma : \sum_{uc} \subseteq \gamma \} \]

Controlled Language: \( L(S_p / G) \)
Controlled Marked Language: \( L_m(S_p / G) = L(S_p / G) \cap L_m(G) \)

\( L_m(G) \): Uncontrolled marked language
\( K \): Desired marked language

where \( K \subseteq L_m(G) \)
Decentralization

Q: When enabled?

A General Decentralized Control Architecture for DES
Conventional Decentralization

Conjunctive coordination
Unanimous ‘Yes’ to enable!

Q: When can we achieve?
Theorem [Rudie and Wonham: 1992]:
There exist a set of local decision rules that achieve the desired language $K$ without blocking under conjunctive coordination iff the following conditions hold:

1. $K$ is **controllable**;
2. $K$ is **C&P co-observable**;
3. $K$ is **$Lm(G)$-closed**

In this case, $L(S_p / G) = \overline{K}$ and $L_m(S_p / G) = K$
Conventional Decentralization

Formal Definitions

A language $K$ is **controllable** w.r.t. $\Sigma_{uc}$ and $L(G)$ if

$$
\overline{K} \Sigma_{uc} \cap L(G) \subseteq \overline{K}
$$

A language $K$ is **C&P co-observable** w.r.t. $\Sigma_{o,1}, \Sigma_{c,1}, \Sigma_{o,2}, \Sigma_{c,2}, L(G)$ if

$$
\forall s \in \overline{K}, \sigma \in \Sigma_c,
$$

$$
s \sigma \in L(G) \setminus \overline{K}
$$

$$
\Rightarrow (\exists i)[P_i^{-1}[P_i(s)]\sigma \cap \overline{K} = \emptyset] \wedge [\sigma \in \Sigma_{c,i}]
$$

A language $K$ is **$L_m(G)$-closed** if

$$
\overline{K} \cap L_m(G) = K
$$

(\subseteq)
Conventional Decentralization

- **Intuition**
  - No uncontrollable illegal continuation $\sigma$ (*Controllability*)

$$\overline{K\Sigma}_{uc} \cap L(G) \subseteq \overline{K}$$
Every illegal controllable continuation $\sigma$ can be surely disabled by one of the supervisors (C&P co-observability)

$$\forall s \in \overline{K}, \sigma \in \Sigma_c, s\sigma \in L(G) \setminus \overline{K},$$

$$(\exists i)[P_i^{-1}[P_i(s)]\sigma \cap \overline{K} = \emptyset] \land [\sigma \in \Sigma_{c,i}]$$

Q: If so, What should the local decisions be?
Conventional Decentralization

‘Pass the buck’, **Permissive** local decisions:
Possible legal behavior induces ‘**Yes (enable)**’

Formal local decision rule:

\[
S_{P_i} (P_i(s)) = \sum \sum_{c,i} \cup \{ \gamma \in \sum_{c,i} : P_i^{-1} P_i(s) \gamma \cap \bar{K} \neq \emptyset \}
\]
The Disjunctive Architecture

Disjunctive coordination (Cf., Prosser et al., 1997)
Enable whenever ‘Yes’ exists

Q: When can we achieve?
The Disjunctive Architecture

Theorem [Yoo and Lafortune: 1999]:

There exist a set of local decision rules that achieve the desired language \( K \) without blocking under disjunctive coordination iff the following conditions hold:

1. \( K \) is controllable;
2. \( K \) is D&A co-observable;
3. \( K \) is \( Lm(G) \)-closed
Formal Definition

A language $K$ is **D&A co-observable** w.r.t. $\Sigma_{o,1}, \Sigma_{c,1}, \Sigma_{o,2}, \Sigma_{c,2}, L(G)$ if

$$\forall s \in \overline{K}, \sigma \in \Sigma_c,$$

$$s\sigma \in \overline{K}$$

$$\Rightarrow (\exists i)[(P_i^{-1}P_i(s) \cap \overline{K})\sigma \cap L(G) \subseteq \overline{K}] \land [\sigma \in \Sigma_{c,i}]$$
Every controllable legal continuation $\sigma$ surely enabled by one of the supervisors (D&A co-observability)

$$\forall s \in K, \sigma \in \Sigma_c, s\sigma \in K$$

$$\Rightarrow (\exists i)[(P_i^{-1} P_i(s) \cap \overline{K}) \sigma \cap L(G) \subseteq \overline{K}] \land [\sigma \in \Sigma_{c,i}]$$

Q: If so, What should the local decisions be?
The Disjunctive Architecture

Antipermissive local decisions:
Possible illegal behavior induces ‘No (disable)’

Formal local decision rule:
\[ S_{P_i}(P_i(s)) = \sum_{uc} \bigcup \{ \gamma \in \sum_{c,i} : (P_i^{-1}P_i(s) \cap \overline{K})\gamma \cap L(G) \subseteq \overline{K} \} \]
Minimally Coordinated Architectures

Q: Are they comparable?

- Coordination can be embedded in a system as ‘Default’ of controllable events (No physical implementation for coordination ⇒ Same complexity of coordination)
  - Conjunction ⇒ ‘Enablement’ default of controllable events
  - Disjunction ⇒ ‘Disablement’ default of controllable events

A General Decentralized Control Architecture for DES
To check C&P co-observability, apply the \textit{permissive rule} under conjunctive coordination.

\(\Sigma_{o,1} = \{\alpha\} , \Sigma_{c,1} = \{\gamma\}\)
\(\Sigma_{o,2} = \{\beta\} , \Sigma_{c,2} = \{\gamma\}\)

Illegal controllable continuation is not disabled.
\(\Rightarrow \) Not C&P co-observable
Incomparability

Now, let us apply the antipermissive rule under disjunctive coordination to check D&A co-observability.

Uncontrolled Behavior

Desired Behavior

Illegal controllable continuation is disabled and desired behavior is achieved.

⇒ D&A co-observable
Incomparability

To check D&A co-observability, apply the antipermissive rule under disjunctive coordination.

\[
\Sigma_{0,1} = \{\alpha\}, \quad \Sigma_{c,1} = \{\gamma\} \\
\Sigma_{0,2} = \{\beta\}, \quad \Sigma_{c,2} = \{\gamma\}
\]

Uncontrolled Behavior  Desired Behavior

\[S_{P_1}\]

\[S_{P_2}\]

Legal controllable continuation \(\gamma\) is not enabled

\(\Rightarrow\) Not D&A co-observable

A General Decentralized Control Architecture for DES
Incomparability

Uncontrolled Behavior | Desired Behavior

Now, let us apply the **permissive rule** under conjunctive coordination to check C&P co-observability.

Legal controllable continuation \( \gamma \) is enabled and the desired behavior is achieved.

\[
\Sigma_{o,1} = \{\alpha\}, \quad \Sigma_{c,1} = \{\gamma\} \\
\Sigma_{o,2} = \{\beta\}, \quad \Sigma_{c,2} = \{\gamma\}
\]

\( \Rightarrow \) C&P co-observable
Incomparability

- Now we can solve more!

We can do even more by using…. 
Partition the set of controllable events: \( \Sigma_c = \Sigma_{c,d} \cup \Sigma_{c,e} \)

Local controllable event partition: \( \Sigma_{c,i} = \Sigma_{c,d,i} \cup \Sigma_{c,e,i} \)

**When can we achieve ?**
The General Architecture

Theorem [Yoo and Lafontune: 1999]:

There exist a set of local decision rules that achieve the desired language $K$ without blocking under the general architecture iff the following conditions hold:

1. $K$ is controllable;
2. $K$ is co-observable;
3. $K$ is $L_m(G)$-closed
The General Architecture

Formal Definition

A language $K$ is **co-observable** w.r.t. $\Sigma_{o,1}, \Sigma_{c,e,1}, \Sigma_{c,d,1}, \Sigma_{o,2}, \Sigma_{c,d,2}, \Sigma_{c,e,2}, L(G)$, if

1. $K$ is C&P co-observable w.r.t. $\Sigma_{o,1}, \Sigma_{c,e,1}, \Sigma_{o,2}, \Sigma_{c,e,2}, L(G)$
2. $K$ is D&A co-observable w.r.t. $\Sigma_{o,1}, \Sigma_{c,d,1}, \Sigma_{o,2}, \Sigma_{c,d,2}, L(G)$

Q: If so, What should the local decisions be?
The General Architecture

Observation 1

\[ \sum_{c,d} \]

\[ \sum_{c,e} \]

\[ \gamma \in \sum_{c,dl,i} : (P_i^{-1}P_i(s) \cap \overline{K}) \gamma \cap L(G) \subseteq \overline{K} \cup \{ \gamma \in \sum_{c,el,i} : P_i^{-1}P_i(s)\gamma \cap \overline{K} \neq \emptyset \} \cup \sum_{c,e} \setminus \sum_{c,e,l} \cup \sum_{uc} \]

\[ \text{Antipermissive} \]

\[ \text{Permissive} \]
The General Architecture is Better

Uncontrolled Behavior

Uncontrolled Behavior

Desired Behavior

Illegal controllable continuation is not disabled.
⇒ Not C&P co-observable

Legal controllable continuation is not enabled.
⇒ Not D&A co-observable

However, it is co-observable with the partition

\[ \Sigma_{o,1} = \{a, g, d\}, \Sigma_{c,1} = \{g, d\} \]

\[ \Sigma_{o,2} = \{b, g, d\}, \Sigma_{c,2} = \{g, d\} \]

\[ \Sigma_{c,d} = \{g\}, \Sigma_{c,e} = \{d\} \]
The General Architecture is better

- Can solve even more with an appropriate partition!
Computational Complexity Issues

Two Issues

- Given a partition of the set of controllable events, how to verify co-observability of the desired language?

- How to find a partition of the set of controllable events that satisfies co-observability?
Computational Complexity Issues

◆ “Brute Force”
  - *Synthesize* a supervisor with combined decision rule and see if it achieves the desired language
    \[ \Rightarrow \text{Exponential time} \] (in the worst case) [Tsitsiklis: 1989]

◆ Power of Non-determinism
  - Possible to construct a non-deterministic machine marking all violations of observability in *Polynomial time* (the centralized architecture) [Tsitsiklis: 1989]
  - Generalized to the conjunctive decentralized architecture: *Polynomial time* construction of non-deterministic FSM ‘\( M \)’ whose state space is
    \[ Q^H \times Q^H \times Q^H \times Q^G \cup \{d\} \]
    All violations of C&P co-observability reach the marked state \( \{d\} \) [Rudie and Willems: 1995]
Can check D&A co-observability in polynomial time!

- Generalized to the disjunctive decentralized architecture: 
  \textit{Polynomial time} construction of non-deterministic FSM ‘\(M_d\)’
  whose state space is \(Q^H \times Q^G \times Q^H \times Q^G \times Q^H \cup \{d\}\)
  [Yoo and Lafortune: 1999]

- All violations of D&A co-observability reach the marked state \(\{d\}\)
**Computational Complexity Issues**

- **Theorem [Yoo and Lafortune:1999]:**

  \( K \) is co-observable w.r.t. a given partition of the controllable events iff

  \[ \sum_{\text{ter}} (L_m (M)) \cap \sum_{\text{ter}} (L_m (M_d)) = \emptyset \]

  where \( \sum_{\text{ter}} (K) = \{ \sigma \in \Sigma : s\sigma \in K \} \)

- **Condition** \( \sum_{\text{ter}} (L_m (M)) \cap \sum_{\text{ter}} (L_m (M_d)) = \emptyset \) is verifiable in polynomial time.
How to find a partition?

Verifying all combinations of partitions takes exponential time

If \( \Sigma_{\text{ter}}(L_m(M)) \cap \Sigma_{\text{ter}}(L_m(M_d)) = \emptyset \) holds,

\[
\Sigma_{c,d} = \Sigma_{\text{ter}}(L_m(M)), \quad \Sigma_{c,e} = \Sigma_c \setminus \Sigma_{c,d}
\]

is one of the partitions satisfying co-observability.

Therefore, a partition satisfying co-observability can be found in polynomial time.
Recall The Example...

**Uncontrolled Behavior**

1. \(\Sigma_{o,1} = \{a, g, d\}, \Sigma_{c,1} = \{g, d\}\)

2. \(\Sigma_{o,2} = \{b, g, d\}, \Sigma_{c,2} = \{g, d\}\)

**Desired Behavior**

- Illegal controllable continuation is not disabled.
  \(\Rightarrow\) **Not C&P co-observable**

- Legal controllable continuation is not enabled.
  \(\Rightarrow\) **Not D&A co-observable**

However, it is co-observable with the partition

\[\Sigma_{c,d} = \{g\}, \Sigma_{c,e} = \{d\}\]
Example

\[ M \]

\[ M_d \]

\[ \Sigma_{\text{ter}} (L_m (M)) = \{ g \} \]

\[ \Rightarrow \Sigma_{\text{ter}} (L_m (M)) \cap \Sigma_{\text{ter}} (L_m (M_d)) = \emptyset \]

\[ \Sigma_{c,d} = \Sigma_{\text{ter}} (L_m (M)) = \{ g \}, \Sigma_{c,e} = \Sigma_c \setminus \Sigma_{c,d} = \{ d \} \]
There are observable languages that cannot be achieved under the general architecture!
Supervisor Synthesis Results

- What to do if K is not co-observable for any partition?

- Details at the CDC!
Concluding Remarks

Summary of Contributions

- Introduction of a general decentralized supervisory control architecture
- Necessary and sufficient condition for the existence of a supervisor for the general architecture
- Performance comparison between the architectures
- Algorithm for polynomial testing of solvability