Formal Methods for the Control of Large-scale Networked Nonlinear Systems with Logic Specifications

Lecture L13: Decentralized control of networks of nonlinear systems

Speaker: Giordano Pola
What’s new?

In this course:

Control of
Large-scale Heterogeneous
Networked Nonlinear Systems
with Logic Specifications

Lecture based on:


What’s new?

To recap:

Control of

Large-scale **Heterogeneous**

Networked **Nonlinear Systems**

with Logic Specifications

Lectures

- L3, L5, L6, L8, L10, L11

**Lecture based on:**


What’s new?

To recap:

Control of Large-scale Heterogeneous Networked Nonlinear Systems with Logic Specifications

Lectures

- L3, L5, L6, L8, L10, L11
- L4, L7

Lecture based on:


What’s new?

To recap:

Control of Large-scale Heterogeneous Networked Nonlinear Systems with Logic Specifications

Lectures

- L3, L5, L6, L8, L10, L11
- L4, L7
- L12

Lecture based on:


What's new?

Here:

Control of
Large-scale Heterogeneous
Networked Nonlinear Systems
with Logic Specifications

Lectures

- L3, L5, L6, L8, L10, L11
- L4, L7
- L12

Lecture based on:


Decentralized control architecture

Approximate equivalence notions

Time-delay systems

Introduction

Decentralized control architecture
The model we consider here ...  

Main assumptions: Ideal communication infrastructures, no disturbance inputs, no state delays and systems in discrete-time
The model we consider here ...

Main assumptions: Ideal communication infrastructures, no disturbance inputs, no state delays and systems in discrete-time
Plant $P_i$ described by the nonlinear discrete-time system

$$P_i:\begin{cases} x_i(t + 1) = f_i(x_i(t), x_j(t), \ldots, u_i(t)) \\ x_i(t) \in \mathbb{R}^{n_i}, x_j(t) \in \mathbb{R}^{n_j}, u_i(t) \in U_i \end{cases}$$

where:

- $x_i(t)$ is the internal state
- $x_j(t)$ is an external measurable input (corresponding to the internal state of $P_j$)
- $u_i(t)$ is the control input where set $U_i$ is finite
The model we consider here …

Controller $C_i$ described by the automaton

$$ C_i: \begin{cases} z_i(t + 1) = g_i(z_i(t)) \\ u_i(t) \in h_i(z_i(k)) \\ z_i(t) \in Z_i, z_i(t) \in Z_{i,0} \subseteq Z_i, u_i(t) \in U_i \end{cases} $$

where:
- $z_i(t) \in Z_i$ is the internal state and $Z_i$ is a finite set
- $u_i(t) \in U_i$ is the output and set $U_i$ is a finite set

Features of $C_i$:
- finite
- dynamic
- open-loop
- state deterministic, but
- output nondeterministic
Control Problem Formulation

Given

- the network of control systems $P_i$
- a regular language specification $L_Q$
  defined over a finite subset $Y_Q$ of $\mathbb{R}^n$
- a desired accuracy $\theta > 0$

Find

- a set of initial states $X_0 \subseteq \mathbb{R}^n$
- a collection of decentralized controllers $C_i$

such that the controlled network, denoted $P^C$, satisfies the specification $L_Q$ up to the accuracy $\theta$, i.e.

for any trajectory $x(.)$ of $P^C$ with $x(0) \in X_0$, there exists a word $q_0 q_1 \ldots q_{t_f}$ of the specification $L_Q$ such that

$$| x(t) - q_t | \leq \theta, \text{ for all } t \in [0; t_f]$$
The main idea …

From networks of nonlinear control systems …

… to networks of symbolic models $T_i$ each one approximating $P_i$
**Transition systems’ representation of the network**

**Given**

\[
P_i: \begin{cases} 
    x_i(t + 1) = f_i(x_i(t), x_i(t), ..., u_i(t)) \\
    x_i(t) \in \mathbb{R}^{n_i}, x_i(t) \in \mathbb{R}^{n_i'}, u_i(t) \in U_i 
\end{cases}
\]

**define**

\[
P: \begin{cases} 
    x(t + 1) = f(x(t), u(t)) \\
    x(t) \in \mathbb{R}^n, u(t) \in U 
\end{cases}
\]

**where**

\[ 
    x(t) = (x_1(t), x_2(t), ..., x_N(t)) 
\]

\[ 
    u(t) = (u_1(t), u_2(t), ..., u_N(t)) 
\]

\[ 
    f(x(t), u(t)) = (f_1(x_1(t), ..., u_1(t)), ..., f_N(x_N(t), ..., u_N(t))) 
\]

\[ 
    T(P) = (X, X_0, U, \rightarrow, X_m, Y, H) 
\]

**where:**

- \( X = X_0 = X_m = \mathbb{R}^n \)
- \( U = U_1 \times U_2 \times ... \times U_N \)
- \( x \rightarrow x' \text{ if } x' = f(x, u) \)
- \( Y = \mathbb{R}^n \)
- \( H(x) = x \)
Networks of symbolic models

Given a quantization vector $\eta = (\eta(1), \eta(2), \ldots, \eta(N))$ we first approximate

$$P_i: \begin{cases} x_i(t + 1) = f_i(x_i(t), x_i(t), \ldots, u_i(t)) \\ x_i(t) \in \mathbb{R}^{n_i}, x_i(t) \in \mathbb{R}^{n_i'}, u_i(t) \in U_i \end{cases}$$

by

$$T^\eta(P_i) = (X^\eta_i, X^\eta_{0,i}, W^\eta_i \times U_i, \rightarrow_i, X^\eta_{m,i}, Y^\eta_i, H^\eta_i)$$

where:

- $X^\eta_i = X_{0,i}^\eta = X_{m,i}^\eta = \eta(i) \mathbb{Z}^n$
- $W^\eta_i = X_{i_1}^\eta \times X_{i_2}^\eta \times \ldots$ where indices $i_j$ are those of $P_{ij}$ that affect dynamics of $P_i$
- $\xi_i \xrightarrow{(w_i,u_i)} \xi'_i$ if $\xi'_i = [f_i(\xi_i, w_i, u_i)]_{\eta(i)}$
- $Y^\eta_i = \mathbb{R}^n$
- $H^\eta_i(\xi_i) = \xi_i$

**Features of $T^\eta(P_i)$**: deterministic, countable and alive

08/20
Networks of symbolic models

Define the network of symbolic models

\[ T(\{T^n(P_i)\}_{i \in [1;N]}) \]

as the transition system obtained by interconnecting \( T^n(P_i) \), i.e.

\[ T(\{T^n(P_i)\}_{i \in [1;N]}) = (X^n, X_0^n, U, \rightarrow_{\eta} X_m^n, Y^n, H^n) \]

where:

- \( X^n = X_0^n = X_m^n = X_1^n \times X_2^n \times ... \times X_N^n \)
- \( U = U_1 \times U_2 \times ... \times U_N \)
- \( (\xi_1, \xi_2, ..., \xi_N)^{(u_1, u_2, ..., u_N)} \rightarrow_{\eta} (\xi'_1, \xi'_2, ..., \xi'_N) \) if \( \xi_i \rightarrow_{i} \xi'_i \) where \( w_i = (\xi_{i1}, \xi_{i2}, ...) \)
- \( Y^n = Y_1^n \times Y_2^n \times ... \times Y_N^n \)
- \( H^n(\xi_1, \xi_2, ..., \xi_N) = (H_1^n(\xi_1), H_2^n(\xi_2), ..., H_N^n(\xi_N)) \)
Networks of symbolic models

Proposition
Suppose that $P$ admits a locally Lipshitz $\delta$-GAS Lyapunov function $V$ satisfying

$$|V(x, y) - V(x, z)| \leq \sigma(|y - z|)$$

for some $K_\infty$ function $\sigma$.

Then, for any desired accuracy $\mu \in \mathbb{R}^+$ and for any quantization vector $\eta \in \mathbb{R}_N^+$ satisfying the following inequality

$$|\eta| \leq \min\{(\sigma^{-1} \circ \rho \circ \alpha_1)(\mu), (\alpha_2^{-1} \circ \alpha_1)(\mu)\}$$

Relation $R_\mu$ specified by

$$(x, \xi) \in R_\mu \iff V(x, \xi) \leq \alpha_1(\mu)$$

is a $\mu$-approximate bisimulation between $T(P)$ and $T\left(\{T^\eta(P_i)\}_{i \in [1;N]}\right)$.

Consequently, transition systems $T(P)$ and $T\left(\{T^\eta(P_i)\}_{i \in [1;N]}\right)$ are $\mu$-bisimilar.

How to find $V$? Use e.g. small gain theorem
Decentralized supervisory control

Example

- Network of two control systems $P_i: x_i(t + 1) = 0.5x_i(t) + u_i(t)$ with $U_i = \{-1,0,1\}$
- Specification $L_Q$ collection of words $(0,0)(1,1)$ and $(0,0)(-1,-1)$

Case 1: $C_i$ do not agree in advance on which word to enforce
- Starting from 0, $C_1$ picks $u_1(0) = 1$
- Starting from 0, $C_2$ picks $u_2(0) = -1$
- From $(0,0)$ to $(1,-1)$ $\Rightarrow$ Specification violated!

Case 2: $C_i$ do agree in advance on which word to enforce:
- If they want to enforce word $(0,0)(1,1)$ both $C_i$ pick $u_i(0) = 1$
- If they want to enforce word $(0,0)(-1,-1)$ both $C_i$ pick $u_i(0) = -1$
$\Rightarrow$ Specification satisfied!
Decentralized supervisory control

Example

- Network of two control systems $P_i: x_i(t + 1) = 0.5x_i(t) + u_i(t)$ with $U_i = \{-1, 0, 1\}$.
- Specification $L_Q$ collection of words $(0,0)(1,1)$ and $(0,0)(-1, -1)$

Case 1: $C_i$ do not agree in advance on which word to enforce
- Starting from 0, $C_1$ picks $u_1(0) = 1$
- Starting from 0, $C_2$ picks $u_2(0) = -1$
- From $(0,0)$ to $(1,-1)$ ⇒ Specification violated!

How to solve problem above?
1. Restriction of the class of specifications, from $L_Q$ to $L_{Q,1} \times L_{Q,2} \times \ldots \times L_{Q,N}$
2. Online agreement on which word to enforce
   (distributed control architecture needed)
3. Offline agreement on which word to enforce
   (decentralized control architecture enough)
Decentralized supervisory control

Example

- Network of two control systems $P_i: x_i(t + 1) = 0.5x_i(t) + u_i(t)$ with $U_i = \{-1, 0, 1\}$.
- Specification $L_Q$ collection of words $(0,0)(1,1)$ and $(0,0)(-1,-1)$

Case 1: $C_i$ do not agree in advance on which word to enforce
- Starting from 0, $C_1$ picks $u_1(0) = 1$
- Starting from 0, $C_2$ picks $u_2(0) = -1$
- From $(0,0)$ to $(1,-1)$ $\Rightarrow$ Specification violated!

How to solve problem above?

1. Restriction of the class of specifications, from $L_Q$ to $L_{Q,1} \times L_{Q,2} \times \ldots \times L_{Q,N}$
2. Online agreement on which word to enforce
   (distributed control architecture needed)
3. Offline agreement on which word to enforce
   (decentralized control architecture enough)
Decentralized supervisory control: solution

Recap from Lecture L7:

- Given $L_Q$ define the transition system $S'_Q$ such that its input marked language coincides with $L_Q$, i.e., $L^u_m(S'_Q) = L_Q$
- Construct the dual transition system $S_Q$ of $S'_Q$, where states of $S_Q$ are transitions of $S'_Q$ and vice versa; we get

$$L^y_m(S_Q) = L^u_m(S'_Q) = L_Q \text{ and } L^y(S_Q) = L^u(S'_Q)$$

EXAMPLE

- $L_Q =$ all words starting with $a$ and ending with $b$ over $Y_Q = \{a, b\}$
- Regular expression $a(a + b)^*b$

Transition system $S'_Q$

Transition system $S_Q$

12/20
Decentralized supervisory control: solution

- Let $S_Q = (X_Q, X_{Q,0}, U_Q, X_{Q,m}, \mathbb{R}^n, H_Q)$
- Let $H_{Q,i}$ be the natural projection of $H_Q$ onto $\mathbb{R}^{n_i}$, i.e.
  \[
  H_{Q,i}(x_Q) = q^i \text{ if } H_Q(x_Q) = (q^1, q^2, ..., q^N)
  \]
- Define operator $I_i: (\xrightarrow{Q}) \times \mathbb{R}^+_N \rightarrow \{\text{True, False}\}$ such that
  \[
  I_i(x_Q \xrightarrow{Q} x'_Q, \eta) = \text{True}, \text{ if } \exists u_i \text{ s.t. } [H_{Q,i}(x_Q)]_{\eta(i)} \xrightarrow{(w_i, u_i)} [H_{Q,i}(x'_Q)]_{\eta(i)}
  \]
  \[
  I_i(x_Q \xrightarrow{Q} x'_Q, \eta) = \text{False}, \text{ otherwise}
  \]
- Define operator $I: (\xrightarrow{Q}) \times \mathbb{R}^+_N \rightarrow \{\text{True, False}\}$ by
  \[
  I(x_Q \xrightarrow{Q} x'_Q, \eta) = \bigwedge_{i \in [1;N]} I_i(x_Q \xrightarrow{Q} x'_Q, \eta)
  \]
Decentralized supervisory control: solution

- Let $S_Q = (X_Q, X_{Q,0}, U_Q, \rightarrow, X_{Q,m}, \mathbb{R}^n, H_Q)$
- Let $H_{Q,i}$ be the natural projection of $H_Q$ onto $\mathbb{R}^{n_i}$, i.e.
  \[ H_{Q,i}(x_Q) = q^i \text{ if } H_Q(x_Q) = (q^1, q^2, ..., q^N) \]
- Define operator $I_i: (\rightarrow) \times \mathbb{R}^+_N \rightarrow \{\text{True, False}\}$ such that
  \[ I_i(x_Q \rightarrow x'_Q, \eta) = \text{True, if } \exists u_i \text{ s.t. } [H_{Q,i}(x_Q)]_{\eta(i)} \stackrel{(w_i, u_i)}{\longrightarrow} [H_{Q,i}(x'_Q)]_{\eta(i)} \]
  \[ I_i(x_Q \rightarrow x'_Q, \eta) = \text{False, otherwise} \]
- Define operator $I: (\rightarrow) \times \mathbb{R}^+_N \rightarrow \{\text{True, False}\}$ by
  \[ I(x_Q \rightarrow x'_Q, \eta) = \bigwedge_{i \in [1;N]} I_i(x_Q \rightarrow x'_Q, \eta) \]
Decentralized supervisory control: solution

Let $S_{Q,\eta}$ be the sub-transition system of $S_Q$ containing all and only transitions $x_Q \xrightarrow{Q} x'_Q$ of $S_Q$ for which

$$I(x_Q \xrightarrow{Q} x'_Q, \eta) = True$$

Remarks

1. $S_{Q,\eta}$ represents the part of the specification that can be matched by using the decentralized control architecture we consider
2. $S_{Q,\eta}$ is blocking in general

In order to overcome 2 compute $\text{Trim}(S_{Q,\eta})$

Remark

$\text{Trim}(S_{Q,\eta})$ contains all information needed to solve our control problem, i.e. to define the set of initial states and to design local controllers $C_i$
Consider any \( q \) word marked by \( \text{Trim}(S_{Q,\eta}) \) and let

\[
S_q = (X_q, x_{q,0}, U_q, \rightarrow_q, \{x_{q,m}\}, R^n, H_q)
\]

be a transition system marking \( q \)

**Remark**
- Remember that \( C_i \) need to agree offline on which word to enforce!
- \( S_q \) can be chosen wlog to be symbolic, accessible and nonblocking

Let \( H_{q,i} \) be the natural projection of \( H_q \) onto \( \mathbb{R}^{n_i} \)

Define

\[
X_0 = R^{-1}_\mu \left( \prod_{i=1}^{n} \left\{ [H_{q,i}(x_{q,0})] \eta(i) \right\} \right)
\]
Decentralized supervisory control: solution

Define entities of local controllers $C_i$ as follows:

$$
\begin{align*}
C_i: \quad & z_i(t + 1) = g_i(z_i(t)) \\
& u_i(t) \in h_i(z_i(k)) \\
& z_i(t) \in Z_i, z_i(t) \in Z_{i,0} \subseteq Z_i, u_i(t) \in U_i
\end{align*}
$$

- $Z_{i,0} = \{x_{q,0}\}$
- $Z_i = X_q$
- $g_i(z_i) = z'_i$ if $z_i \rightarrow z'_i$
- $h_i(z_i(t)) = \left\{ u_i \in U_i \mid g_i(z_i) = z'_i \text{ and } \left[H_{Q,i}(z_i)\right]_{\eta(i)} \xrightarrow{(w_i,u_i)}_{i} \left[H_{Q,i}(z'_i)\right]_{\eta(i)} \right\}$

Decentralized supervisory control: solution
Theorem
Suppose that \( P \) admits a locally Lipshitz \( \delta \)-GAS Lyapunov function \( V \) satisfying

\[
|V(x, y) - V(x, z)| \leq \sigma(|y - z|)
\]

for some \( K_\infty \) function \( \sigma \). For any desired accuracy \( \theta \in \mathbb{R}^+ \) select \( \mu \in \mathbb{R}^+ \) and \( \eta \in \mathbb{R}_N^+ \) satisfying

\[
|\eta| \leq \min\{(\sigma^{-1} \circ \rho \circ \alpha_1)(\mu), (\alpha_2^{-1} \circ \alpha_1)(\mu)\}
\]

\[
\mu + |\eta| \leq \theta
\]

Then set \( X_0 \) and local controllers \( C_i \) solve the decentralized supervisory control problem.
Remarks

- **Comparison with centralized control architectures**: achievable behavior in centralized and decentralized control architecture is the same.

- **Efficient on-the-fly control algorithms** allowing also parallel computing architectures to further speed up computations.

```plaintext
input: $S_Q = (X_Q, X_Q, U_Q, \rightarrow^Q, X_Q, m, R^n, H_Q)$;
for each $x_Q \rightarrow^Q x_Q^+$ do
  for each $i \in [1; N]$ do
    set $I_i(x_Q \rightarrow^Q x_Q^+, \eta) := \text{False}$;
    compute the set $h_{c,i}(x_Q)$ of all $u_i \in U_i^n$ satisfying (20);
    if $h_{c,i}(x_Q) \neq \emptyset$ then
      set $I_i(x_Q \rightarrow^Q x_Q^+, \eta) := \text{True}$;
    end
  end
set $I(x_Q \rightarrow^Q x_Q^+, \eta) := \bigwedge_{i \in [1; N]} I_i(x_Q \rightarrow^Q x_Q^+, \eta)$;
end
compute $S_{Q,\eta}$ in (23);
compute Trim($S_{Q,\eta}$);
output: Trim($S_{Q,\eta}$) and $h_{c,i}, i \in [1; N]$;
```

Algorithm 1: Decentralized local controllers design.
Remarks

- **Comparison with centralized control architectures**: achievable behavior in centralized and decentralized control architecture is the same.

- **Efficient on-the-fly control algorithms** allowing also parallel computing architectures to further speed up computations.

- **Computational complexity analysis**: 
  Decentralized approach: linear growth with the number $N$ of subsystems $P_i$
  Centralized approach: exponential growth with $N$. 
Example

Temperature regulation of a circular building with \( N \) rooms \( r_i \)

\[
T_i(t + 1) = T_i(t) + \alpha(T_{i+1}(t) + T_{i-1}(t) - 2T_i(t)) + \\
+ \beta(T_e - T_i(t)) + \gamma(T_h - T_i(t))u_i(t)
\]

\( u_i(t) \in U_i = 0.25 \mathbb{Z} \cap [0, 1] \)

\( \theta = 0.5 \)

specification:

<table>
<thead>
<tr>
<th>( t \mod(12) )</th>
<th>( T_1(t) )</th>
<th>( T_i(t), i \in {2; N} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>19</td>
<td>18</td>
</tr>
<tr>
<td>1</td>
<td>19</td>
<td>18.5</td>
</tr>
<tr>
<td>2</td>
<td>19</td>
<td>19</td>
</tr>
<tr>
<td>3</td>
<td>19</td>
<td>19.5</td>
</tr>
<tr>
<td>4</td>
<td>19</td>
<td>20</td>
</tr>
<tr>
<td>5</td>
<td>19</td>
<td>20</td>
</tr>
<tr>
<td>6</td>
<td>19</td>
<td>20</td>
</tr>
<tr>
<td>7</td>
<td>19</td>
<td>19.5</td>
</tr>
<tr>
<td>8</td>
<td>19</td>
<td>19</td>
</tr>
<tr>
<td>9</td>
<td>19</td>
<td>18.5</td>
</tr>
<tr>
<td>10</td>
<td>19</td>
<td>18.25</td>
</tr>
<tr>
<td>11</td>
<td>19</td>
<td>18</td>
</tr>
</tbody>
</table>
Example

Temperature regulation of a circular building with N rooms $r_i$

$$T_i(t+1) = T_i(t) + \alpha(T_{i+1}(t) + T_{i-1}(t) - 2T_i(t)) +$$
$$+ \beta(T_e - T_i(t)) + \gamma(T_h - T_i(t))u_i(t)$$

$u_i(t) \in U_i = 0.25 \mathbb{Z} \cap [0,1]$

$\theta = 0.5$

specification:

<table>
<thead>
<tr>
<th>$t \mod(12)$</th>
<th>$T_1(t)$</th>
<th>$T_i(t)$, $i \in [2;N]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>19</td>
<td>18</td>
</tr>
<tr>
<td>1</td>
<td>19</td>
<td>18.5</td>
</tr>
<tr>
<td>2</td>
<td>19</td>
<td>19.5</td>
</tr>
<tr>
<td>3</td>
<td>19</td>
<td>20</td>
</tr>
<tr>
<td>4</td>
<td>19</td>
<td>20.5</td>
</tr>
<tr>
<td>5</td>
<td>19</td>
<td>20</td>
</tr>
<tr>
<td>6</td>
<td>19</td>
<td>19.5</td>
</tr>
<tr>
<td>7</td>
<td>19</td>
<td>19.5</td>
</tr>
<tr>
<td>8</td>
<td>19</td>
<td>19.5</td>
</tr>
<tr>
<td>9</td>
<td>19</td>
<td>18.5</td>
</tr>
<tr>
<td>10</td>
<td>19</td>
<td>18.25</td>
</tr>
<tr>
<td>11</td>
<td>19</td>
<td>18</td>
</tr>
</tbody>
</table>

controller:

<table>
<thead>
<tr>
<th>$t \mod(12)$</th>
<th>$T_1(t)$</th>
<th>$T_i(t)$, $i \in [2;N]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>19.5000</td>
<td>18.5000</td>
</tr>
<tr>
<td>1</td>
<td>18.9788</td>
<td>18.8462</td>
</tr>
<tr>
<td>2</td>
<td>18.7329</td>
<td>19.2453</td>
</tr>
<tr>
<td>3</td>
<td>18.6496</td>
<td>19.6773</td>
</tr>
<tr>
<td>4</td>
<td>18.6042</td>
<td>20.1283</td>
</tr>
<tr>
<td>5</td>
<td>18.5992</td>
<td>20.1021</td>
</tr>
<tr>
<td>6</td>
<td>18.6087</td>
<td>20.0838</td>
</tr>
<tr>
<td>7</td>
<td>18.6176</td>
<td>19.5723</td>
</tr>
<tr>
<td>8</td>
<td>18.6176</td>
<td>19.5325</td>
</tr>
<tr>
<td>9</td>
<td>18.6176</td>
<td>19.5325</td>
</tr>
<tr>
<td>10</td>
<td>18.6176</td>
<td>19.5325</td>
</tr>
<tr>
<td>11</td>
<td>18.6176</td>
<td>19.5325</td>
</tr>
</tbody>
</table>

validation:

$$|20 - 19.5924| = 0.4076 < 0.5 = \theta$$

Computational complexity analysis (ONLY for $N = 4$):

- Decentralized architecture $0.1563s$
- Centralized architecture $163.6304s$

Gain: $\frac{163.6304}{0.1563} = 1046$!
Conclusions

We proposed decentralized control architectures enforcing regular language specifications on incrementally stable networks of discrete-time nonlinear control systems.