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

Basilica di Santa Maria di Collemaggio, L'Aquila (Italy), 1287

Lecture L1: Introduction

Speaker: Maria Domenica Di Benedetto
Organizers and Speakers

Maria Domenica Di Benedetto
Dipartimento di Ingegneria e Scienze dell'informazione e Matematica
Center of Excellence DEWS
University of L'Aquila, Italy
mariadomenica.dibenedetto@univaq.it

Giordano Pola
Dipartimento di Ingegneria e Scienze dell'informazione e Matematica
Center of Excellence DEWS
University of L'Aquila, Italy
giordano.pola@univaq.it

Pierdomenico Pepe
Dipartimento di Ingegneria e Scienze dell'informazione e Matematica
Center of Excellence DEWS
University of L'Aquila, Italy
pierdomenico.pepe@univaq.it

Alessandro Borri
Istituto di Analisi dei Sistemi ed Informatica "A. Ruberti" (IASI)
Consiglio Nazionale delle Ricerche (CNR), Rome, Italy
alessandro.borri@iasi.cnr.it
Cyber-Physical Systems of Systems

What is a Cyber-Physical System (CPS)?

- First generation CPS (Embedded Systems):
  Computational systems, but not stand alone computers, interfacing sensors and actuators, reactive to physical environment stimuli, designed to perform one or a few dedicated functions, often with real-time computing constraints

- Focus on the computation
Second generation CPS: more importance to the link and communication between computational and physical elements

CPSoS (European working group CPSoS, [www.cpsos.eu](http://www.cpsos.eu)) are large-scale, complex, heterogeneous, distributed and networked systems

Tight combination of, and coordination between, physical process and computational and communication components

SoS: Autonomy of the subsystems, dynamic reconfiguration, emerging behaviors
Tight interaction of many distributed, real-time computing systems and physical systems

Examples
- Airplanes
- Cars
- Buildings with advanced HVAC controls
- Manufacturing plants
- Power plants
- ...

Cyber-Physical Systems of Systems
“Are today’s resources and knowledge in control, communications and computing adequate to deal with such a kind of complex systems?”

(E. Lee, UC Berkeley)
Heterogeneous systems, the motivation for hybrid models

- Combination of discrete and continuous time with a prescribed hierarchy

\[ \frac{dx}{dt} = -x \quad x \geq 2 \]

\[ \frac{dx}{dt} = x \quad x \leq 3 \]

- Models with “heterogeneous components”
Heterogeneous systems, the motivation for hybrid models

- **Continuous systems with a phased operation:**
  - walking robots
  - biological cell growth and division

- **Continuous systems controlled by discrete logic:** need for design techniques that can guarantee safety and performance specifications of *embedded systems*, or systems that couple discrete logic with the analog physical environment.
  - thermostat
  - chemical plants with valves, pumps
  - control modes for complex systems, eg. intelligent cruise control in automobiles, aircraft autopilot modes

- **Coordinating processes:** many interacting subsystems (multi-agent systems) typically feature continuous controllers to optimize performance of individual agents, and coordination among agents to compete for scarce resources, resolve conflicts, etc.
  - air and ground transportation systems
  - swarms of micro-air vehicles
Finite state machines

1. Insert coin
2. Pull handle
3. Win if the combination is good, otherwise lose

Slot machine
Features of finite state machines

- Events are **time-abstract**
- Events are not necessarily equipped with any notion of ‘internal-external’ or ‘input-output’
- **Compositionality** is possible
- There can be **non-determinism**
- Just like modeling of continuous systems, the level of detail is ‘modeler dependent’
Ariane V launched on 4th June 1996. It exploded 37s after launch.
The program had been running for 10 years, costing $7 billions.
Software worked perfectly on Ariane IV, the same was used in Ariane V.

What had changed, was the physical system around the software...
Networked systems

Nonidealities in communication infrastructures:
- Quantization errors
- Time-varying network access times
- Time-varying communication delays induced by the network
- Limited bandwidth
- Packet losses
- ...

(e.g. [Andersson, IEEE-CDC-05], [Antsaklis, IEEE-TAC-04], [Heemels, IEEE-TAC-10], [Hespanha, Proc. IEEE-07], [Murray, SMTNS-06])
Logic specifications

\[
\frac{dx}{dt} = f(x, m)
\]
Logic specifications

Automata theory, Linear Temporal Logic, Computational Tree Logic, Metric Temporal Logic ...

Examples:
- Language specifications
- Synchronization specifications
- Obstacle avoidance
- Switching specifications
- ...

\[ \varphi = \forall \left( \neg \phi_1 \land \neg \phi_2 \cup_{t \geq 0} \left( \neg \phi_1 \land \phi_2 \cup_{60 \leq t \leq 180} \phi_1 \land \phi_2 \right) \right) \]

[D’Innocenzo, Julius, Pappas, Di Benedetto, Di Gennaro; IEEE-CDC-2007]
Heterogeneity: plants, controllers and specifications described in different mathematical frameworks

Non-ideal communication infrastructure: control action delivered with delay on the basis of delayed and corrupted measure of the states of the plants, lack of information (packet drops), etc.

Complexity: large number of systems composed of several, possibly distributed sub-systems

Logic specifications
Triple C-convergence

Control Theory

Computer Science

Communication Systems

Heterogeneous, large-scale networked control systems
Todays’ research approaches to CPSoS

- Resource aware control
- Distributed control
- Networked control systems

(see e.g. 7th PhD School on CPS, Lucca, June 12-15, 2017)

- Co-design of physical, computational and communication systems
- Formal methods
Projects funded in the USA

- Correct-by-Design Control Software Synthesis for Highly Dynamic Systems (NSF 1239085)
- Towards robust cyber-physical systems (NSF 1035916)
- Closing the gap in controller synthesis (NSF 0953994)
- Automated Synthesis of Embedded Control Software (NSF 0717188)
- Formal Methods for Motion Planning and Control with Human-in-the-Loop (NSF NRI-1426907)
- A formal approach to control of hybrid systems (NSF CNS-0834260), etc.

Special issues

- Formal Methods in Control (Journal of DEDS 2016), etc.

Plenary lectures

- Tabuada, Bisimulation: From Differential Equations to Finite-State Machines and Back (ACC 2010)
- Pappas, Approximate Bisimulation: A Bridge Between Computer Science and Control Theory (CDC 2011)
- Di Benedetto - Pola, Networked Embedded Control Systems: from Modelling to Implementation (ICSCS 2012), etc.
Research at DEWS

- **Stable Control Systems:**
  - Stable switched systems [TAC-2010]
  - Stable control systems with disturbances & AEA bisimulation [SIAM-2009] [IJC-2012]
  - Stable control systems with regular language specifications & quantized measurements [CDC-2016]
  - Stable control systems [Automatica-2008]
  - Incremental stability [Angeli,TAC-2002]
  - Approximate bisimulation [Girard & Pappas,TAC-2007]

- **Unstable Control Systems:**
  - Unstable control systems [TAC-2012]

- **Efficient Control Algorithms:**
  - Efficient control algorithms [TAC-2012]

- **Networked Control Systems:**
  - Networked control systems [HSCC-2012] [ERCIM News ‘97]

- **Stable Time-Delay Systems:**
  - Stable time-delay systems [SCL-2010]

- **Stable Time-Varying Delay Systems:**
  - Stable time-varying delay systems [IJRNC-2015]

- **Networks of Control Systems:**
  - Networks of control systems [TAC-2017]

- **PWA Systems:**
  - PWA systems [TAC-2014]

- **Decentralized Control with Regular Language Specifications:**
  - Decentralized control with regular language specifications [NecSys 2013] [TAC-submitted]
Key idea: homogenizing heterogeneities in the formal description of plants controllers and specifications.

Bridget Riley, *Movement in Squares, 1961*

... from continuous to discrete systems!
A three phases process:
#1. Construct the finite/symbolic model $T$ approximating the plant system $P$
#2. Design a finite/symbolic controller $C$ that solves the specification $S$ for $T$
#3. Refine the controller $C$ to the controller $C'$ to be applied to $P$

Correct-by-design embedded control software
A three phases process:

#1. Construct the finite/symbolic model $T$ approximating the plant system $P$.

#2. Design a finite/symbolic controller $C$ that solves the specification $S$ for $T$.

#3. Refine the controller $C$ to the controller $C'$ to be applied to $P$.

Advantages:

- Integration of software and hardware constraints in the control design of purely continuous processes.
- Logic specifications can be addressed.
... towards

Controlling
Large-scale networked nonlinear systems with logic specifications
Large-scale networked nonlinear systems

\[
P_1 x_1 x'_1 x_2 x'_2 x_3 x'_3 \ldots x'_{N-1} x_N\]

\[
1 \rightarrow d_1 \rightarrow P_1 \rightarrow \ldots \rightarrow d_N \rightarrow P_N
\]

\[
u'_1 \rightarrow x_1 \rightarrow x'_1 \rightarrow x_2 \rightarrow x'_2 \rightarrow \ldots \rightarrow x'_{N-1} \rightarrow x_N \rightarrow u'_N
\]
Large-scale networked nonlinear systems

Distributed Control Architecture

\[ \begin{align*}
\mathbf{x}'_1 & \rightarrow \mathbf{x}_2 \\
\mathbf{x}_2 & \rightarrow \mathbf{x}_3 \\
\mathbf{x}_3 & \rightarrow \mathbf{x}_{N-1} \\
\mathbf{x}_{N-1} & \rightarrow \mathbf{x}_N
\end{align*} \]
Large-scale networked nonlinear systems

Decentralized Control Architecture

\[
\begin{align*}
P_1 & \xrightarrow{x_1, x'_2} C_1 \\
P_2 & \xrightarrow{x_1, x'_2, x'_3} C_2 \\
P_N & \xrightarrow{x_N, x'_{N-1}} C_N \\
\end{align*}
\]
Large-scale networked nonlinear systems

Plant $P_i$: nonlinear time-delay system

$P_i: \begin{cases} \dot{x}_i(t) = f_i(x_i(t), x_i(t - \Delta_i(t)), x'_j(t), \ldots, u_i(t), d_i(t)) \\ x_i(t) \in X_i \subseteq \mathbb{R}^{n_i}, x'_j(t) \in X_j \subseteq \mathbb{R}^{n_j}, u_i(t) \in U_i, d_i(t) \in D_i \subseteq \mathbb{R}^{l_i} \end{cases}$

where:
- $x_i(t) \in X_i \subseteq \mathbb{R}^{n_i}$ internal state
- $x'_j(t) \in X_j \subseteq \mathbb{R}^{n_j}$ external measurable input (corresponding to the internal state of $P_j$ corrupted by the network)
- $u_i(t) \in U_i$ control input, where set $U_i$ is finite
- $d_i(t) \in D_i \subseteq \mathbb{R}^{l_i}$ external non-measurable disturbance
- $\Delta_i(t)$ time-varying delay

$P_i$: infinite dimensional control system (because of delays)
Controller $C_i$: automaton

\[
C_i: \begin{cases}
    z_i(k + 1) \in g_i(z_i(k), x'_{i}(k), x'_{i'}(k), \ldots) \\
    u_i(k + 1) \in h_i(z_i(k), x'_{i}(k), x'_{i'}(k), \ldots) \\
    z_i(k) \in Z_i, x'_{i}(k) \in X'_i, x'_{i'}(k) \in X'_{i'}, u_i(k) \in U_i
\end{cases}
\]

where:

- $z_i(k) \in Z_i$ internal state and $Z_i$ finite set
- $x'_{i}(k) \in X_i$ external measurable input (corresponding to the internal state $x_i(k)$ of $P_i$ corrupted by the network)
- $x''_{j}(k) \in X_j$, external measurable input (corresponding to the internal state $x_j(k)$ of $P_j$ corrupted by the network)
- $u_i(t) \in U_i$ is the output and $U_i$ finite set

$C_i$: finite, dynamic feedback and nondeterministic
Large-scale networked nonlinear systems

Nonideal communication infrastructure
Quantization errors, time-varying network access times, time-varying communication delays, limited bandwidth, packet losses, ...

\[ \begin{align*}
    t_{2k} & \\
    \Delta_{ca} & \\
    \Delta_{sc} & \\
    \end{align*} \]
Recall

- Let $Y$ be a finite set representing an alphabet
- A word over $Y$ is a finite sequence with symbols in $Y$
- A language $L$ over $Y$ is a collection of words in $Y$

**Example**

$Y =$ the Latin alphabet
$L_1 =$ the Italian language $= \{ a, e, i, o, ad, al, \ldots \}$
$L_2 =$ all words over $Y$ with symbol $a =$ $\{ a, aa, aaa, aaaa, \ldots \}$
$L_3 =$ all words over $Y$ of the form $a^n b^n$ with $n$ integer $= \{ ab, aabb, aaabbb, \ldots \}$

$L_1$, $L_2$, $L_3$ are languages

Languages may be composed of a finite number or an infinite number of words

**Example (continued)**

$L_1$ is finite while $L_2$ and $L_3$ are not!
Recall
- Let \( Y \) be a finite set representing an alphabet
- A word over \( Y \) is a finite sequence with symbols in \( Y \)
- A language \( L \) over \( Y \) is a collection of words in \( Y \)

Example
\( Y = \) the Latin alphabet
\( L_1 = \) the Italian language = \{ a, e, i, o, ad, al, ... \}
\( L_2 = \) all words over \( Y \) with symbol \( a \) = \{ a, aa, aaa, aaaa, ... \}
\( L_3 = \) all words over \( Y \) of the form \( a^n b^n \) with \( n \) integer = \{ ab, aabb, aaabbb, ... \}
\( L_1, L_2, L_3 \) are languages

Definition
A language is regular if it can be represented by a finite state automaton

Example (continued)
\( L_1 \) is regular because it is finite
\( L_2 \) is regular because of existence of \( A_2 \)
\( L_3 \) is not regular!
Consider a collection $Y$ of left-closed right-open hyper-cubes $Y_i$ of $\mathbb{R}^n$

$$Y_i = c_i + \prod_{i=1}^{n} [-\eta, \eta)$$

$$c_i \in 2\eta \mathbb{Z}^n$$

$Y$ = Collection of $Y_i$
$Y$ is a partition of $\mathbb{R}^n$

We consider a specification expressed as a regular language $L_Q$ over $Y$

**Example** Starting from $I$ reach $T$ in finite time while avoiding $O$
Consider a collection $Y$ of left-closed right-open hyper-cubes $Y_i$ of $\mathbb{R}^n$

$$Y_i = c_i + \Pi_{i=1}^n [-\eta, \eta)$$

$$c_i \in 2\eta \mathbb{Z}^n$$

$Y = \text{Collection of } Y_i$

$Y$ is a partition of $\mathbb{R}^n$

We consider a specification expressed as a regular language $L_Q$ over $Y$

**Example** Starting from $I$ reach $T'$ in finite time while avoiding $O$

$L_Q = \text{collection of words starting with } \square, \text{ ending with } \bigcirc \text{ and with no } \Box$
Specifications: Regular languages

Consider a collection $Y$ of left-closed right-open hyper-cubes $Y_i$ of $\mathbb{R}^n$

$$Y_i = c_i + \prod_{i=1}^{n} [-\eta, \eta)$$

$$c_i \in 2\eta \mathbb{Z}^n$$

$Y$ = Collection of $Y_i$

$Y$ is a partition of $\mathbb{R}^n$

We consider a specification expressed as a regular language $L_Q$ over $Y$

Specifications handled via regular language formalism:

- Reachability
- Controlled invariance
- Obstacle avoidance
- Motion planning
- Enforcing periodic orbits
- State-based switching specifications
- ...
Specifications: Regular languages

Example #1: Robotics

Specification: Enforcing periodic orbits

Regular language:
Word obtained by concatenating symbols $w_1 w_2 \ldots w_{26}$
Specifications: Regular languages

Example #2: Robot motion planning

**Specification:** Starting from the green box, reach the red box while avoiding the blue obstacles

How to formalize this specification as a regular language?
Example #3: Vehicle platooning

**Specification:** Maintain security distance from the vehicle in front of you

How to formalize this specification as a regular language?

**Safety problem:** Define the set of good states as those for which \( |x_i - x_{i+1}| \geq d \) and consider all words with symbols in the set of good states (All words with no red symbols)
Specifications: Regular languages

Example #4: Systems Biology (synthetic gene network*)

Specification: Enforce low concentration of protein 1 and high concentration of protein 2 while avoiding intermediate concentrations of both proteins.

How to formalize this specification as a regular language?

Same approach as in Example #2

* Taken from:

IEEE TRANSACTIONS ON AUTOMATIC CONTROL, VOL. 57, NO. 6, JUNE 2012

Temporal Logic Control of Discrete-Time
Piecewise Affine Systems

Boyan Yordanov, Member, IEEE, Jana Tůmová, Ivana Černá, Jiří Barnat, and Calin Belta, Senior Member, IEEE
Given

- the network of control systems $P_i$
- a regular language specification $L_Q$
- a desired accuracy $\theta > 0$
- a sampling time $\tau > 0$

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 $q_0q_1...q_{sf}$ of the specification $L_Q$ such that

$$| x(s\tau) - q_s | \leq \theta, \text{ for all } s \in [0; sf]$$
The approach we take ... 

... a complementary approach:

- Single plants with no disturbances and delays
- Control design with logic specifications
- Efficient algorithms for control design
- Single plants with disturbances
- Single plants with delays
- Single, possibly unstable, plants
- Single plant, controller and communication infrastructure
- Decentralized control of networks of control systems with logic specifications
<table>
<thead>
<tr>
<th>day</th>
<th>morning/afternoon</th>
<th>lecture</th>
<th>title</th>
<th>who</th>
<th>duration (min)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 m</td>
<td></td>
<td>1</td>
<td>Introduction</td>
<td>Di Benedetto</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2a</td>
<td>Review on internal and external stability notions for nonlinear systems</td>
<td>Pepe</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Break</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2b</td>
<td>Review on internal and external stability notions for nonlinear systems</td>
<td>Pepe</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3</td>
<td>Metric transition systems</td>
<td>Di Benedetto</td>
<td>45</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>lunch</td>
</tr>
<tr>
<td>1 a</td>
<td></td>
<td>4</td>
<td>Regular languages</td>
<td>Pola</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Break</td>
</tr>
<tr>
<td>2 m</td>
<td></td>
<td>5</td>
<td>Relations among metric transition systems</td>
<td>Di Benedetto</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6</td>
<td>Symbolic models for stable nonlinear systems</td>
<td>Pola</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7</td>
<td>Control design and efficient algorithms</td>
<td>Borri</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>lunch</td>
</tr>
<tr>
<td>2 a</td>
<td></td>
<td>8</td>
<td>Symbolic models and control for nonlinear systems with disturbances and applications</td>
<td>Borri</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9a</td>
<td>Nonlinear time-delay systems: basic theory and stability</td>
<td>Pepe</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9b</td>
<td>Nonlinear time-delay systems: basic theory and stability</td>
<td>Pepe</td>
<td>90</td>
</tr>
<tr>
<td></td>
<td></td>
<td>10</td>
<td>Symbolic models for time-delay systems</td>
<td>Pola</td>
<td>30</td>
</tr>
<tr>
<td>3 m</td>
<td></td>
<td>11</td>
<td>Symbolic models for possibly unstable nonlinear systems</td>
<td>Pola</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td></td>
<td>12</td>
<td>Symbolic models and control for networked nonlinear systems</td>
<td>Borri</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>Break</td>
</tr>
<tr>
<td></td>
<td></td>
<td>13</td>
<td>Decentralized control of networks of nonlinear systems</td>
<td>Pola</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td></td>
<td>14</td>
<td>Tools</td>
<td>Borri</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td></td>
<td>15</td>
<td>Conclusions</td>
<td>Pola</td>
<td>30</td>
</tr>
</tbody>
</table>

**Keys:** background basic advanced