A Data Oriented Approach for Real Time Systems

RTNS 2009

Tanguy Le Berre,, Philippe Mauran,
Gérard Padiou, Philippe Quéinnec

Université de Toulouse - IRIT - ACADIE

2009, October 27th

Objectives

- Domain: distributed real time systems
- Objectives:
  - abstract and non operational specification of a system
  - detailed description of the architecture
  - real time properties defined on the architecture
- Verification:
  - feasibility of the architecture constrained by real time properties
  - conformity of an implementation
  - formal and automated verification

Our approach in a few words

Real time systems

The total correctness of an operation depends not only upon its logical correctness, but also upon the time in which it is performed

- description of the system as a set of tasks
- checking satisfaction of operational deadlines

Another definition for real time systems

The correctness of the system depends upon the logical and the temporal correctness of the data

Temporal correctness of variables values defined by:

- the semantics of the variables
- the interdependencies between variables: propagation of values

Our approach in a few words

Real time systems

The total correctness of an operation depends not only upon its logical correctness, but also upon the time in which it is performed

- description of the system as a set of tasks
- checking satisfaction of operational deadlines

Another definition for real time systems

The correctness of the system depends upon the logical and the temporal correctness of the data

Temporal correctness of variables values defined by:

- the semantics of the variables
- the interdependencies between variables: propagation of values

System specification

A specification is defined by a couple:

The system architecture

- the relation between the variables values
- the definition of propagation paths

The real time properties of the system

- properties on the variables behaviors
- properties on the propagation paths

System specification

A specification is defined by a couple:

The system architecture

- the relation between the variables values
- the definition of propagation paths

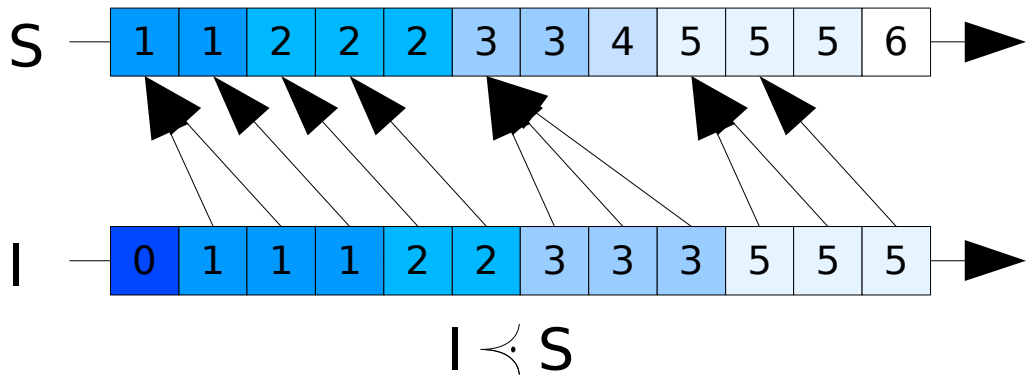
The real time properties of the system

- properties on the variables behaviors
- properties on the propagation paths

The observation relation

A relation between two expressions: an image and a source

- image values defined by previous source values
- image history: a sub-sequence of the source history
- respect of the chronological order
- possible loss
- logical delay



Models an arbitrary delayed flow of value

The observation relation

$`x \prec\cdot x$, an abstraction of a communication

- a relation between two variables
- $`x$ is a delayed copy of x through a communication link

$y \prec\cdot f(x_1, x_2)$: an abstraction of a computation

- a relation between an output variable and a set of input variables
- y is a delayed copy of $f(x_1, x_2)$
- models a synchronous computation
- asynchronous computation:

$$y \prec\cdot f(`x_1, `x_2) \ \wedge \ `x_1 \prec\cdot x_1 \ \wedge \ `x_2 \prec\cdot x_2$$

The observation relation

$`x \prec\cdot x$, an abstraction of a communication

- a relation between two variables
- $`x$ is a delayed copy of x through a communication link

$y \prec\cdot f(x_1, x_2)$: an abstraction of a computation

- a relation between an output variable and a set of input variables
- y is a delayed copy of $f(x_1, x_2)$
- models a synchronous computation
- asynchronous computation:

$$y \prec\cdot f(`x_1, `x_2) \ \wedge \ `x_1 \prec\cdot x_1 \ \wedge \ `x_2 \prec\cdot x_2$$

Description of a system architecture

Architecture of a system defined as a set of observation relations:

- defines an oriented graph
- nodes = variables
- edges = computation or communication

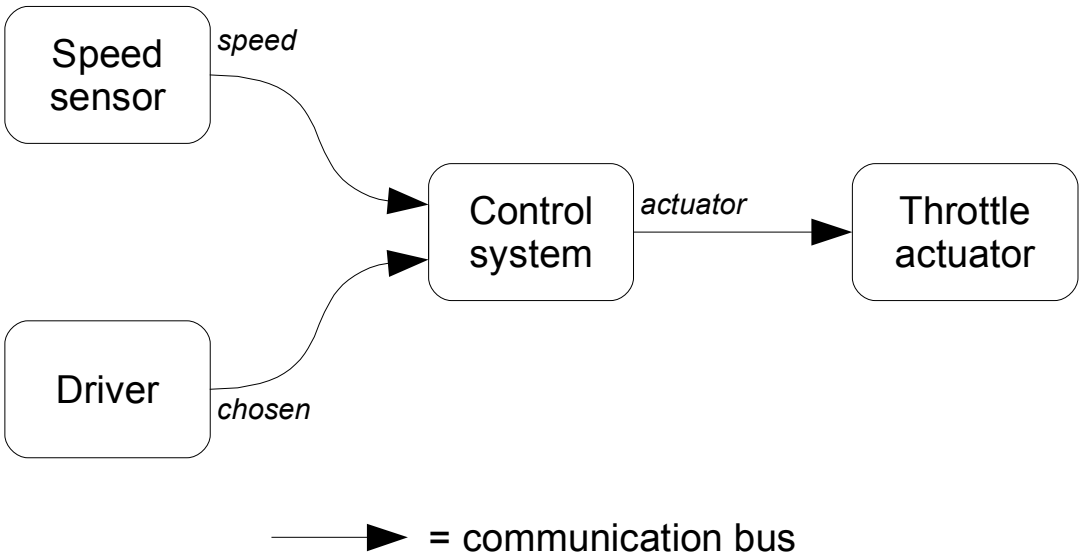
Propagation paths:

- values propagation paths defined by paths between graph nodes
- logical delays introduced by the observation relations

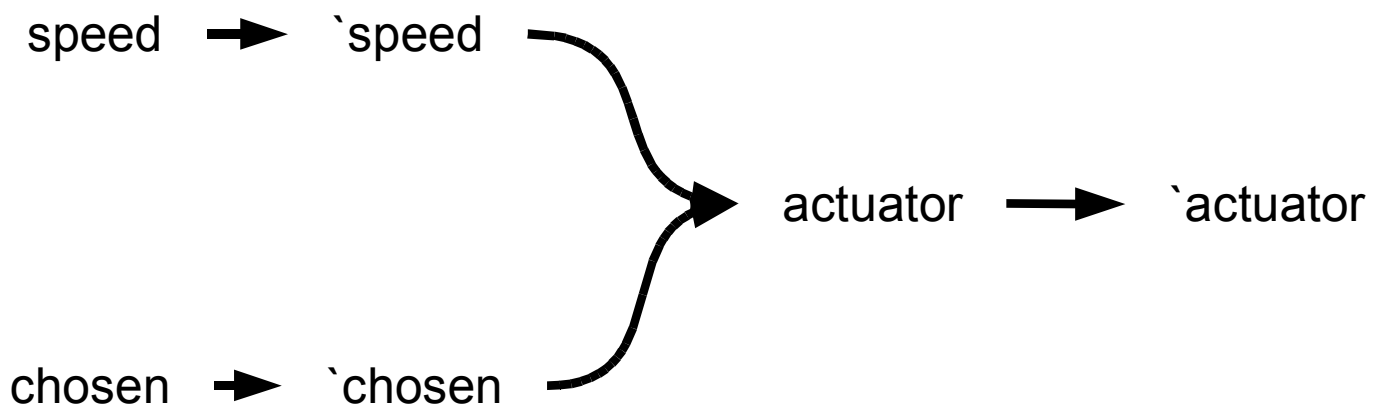
$$\Rightarrow \text{need for real time constraints}$$

A simplified car cruise control system

- a speed sensor
- a driver command
- a control system
- a throttle actuator
- a communication bus

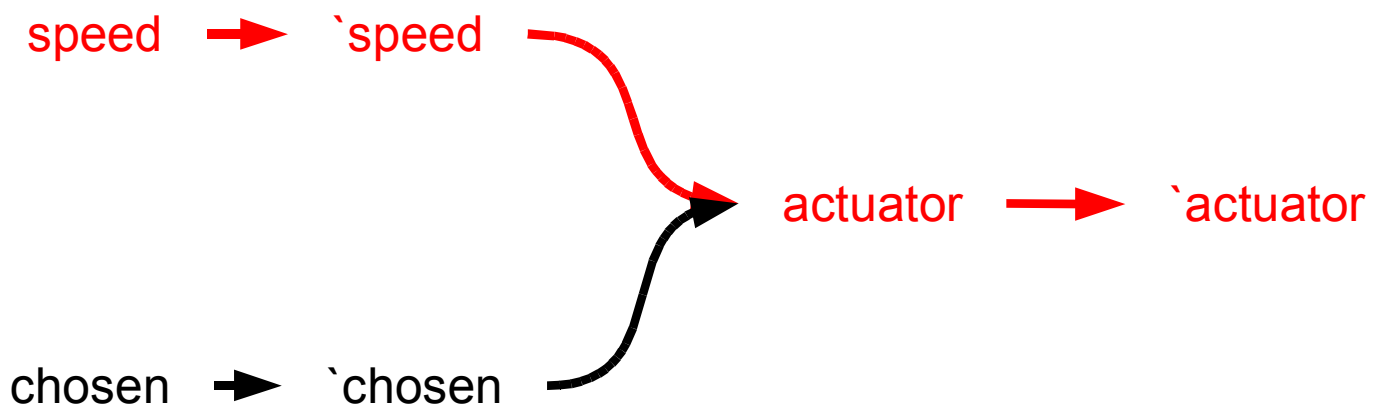


Example architecture



$$
\begin{aligned}
{}^{\backprime}speed &\prec\cdot\, speed \\
{}^{\backprime}chosen &\prec\cdot\, chosen \\
{}^{\backprime}actuator &\prec\cdot\, actuator \\
actuator &\prec\cdot\, control({}^{\backprime}speed, {}^{\backprime}chosen)
\end{aligned}
$$

A propagation path

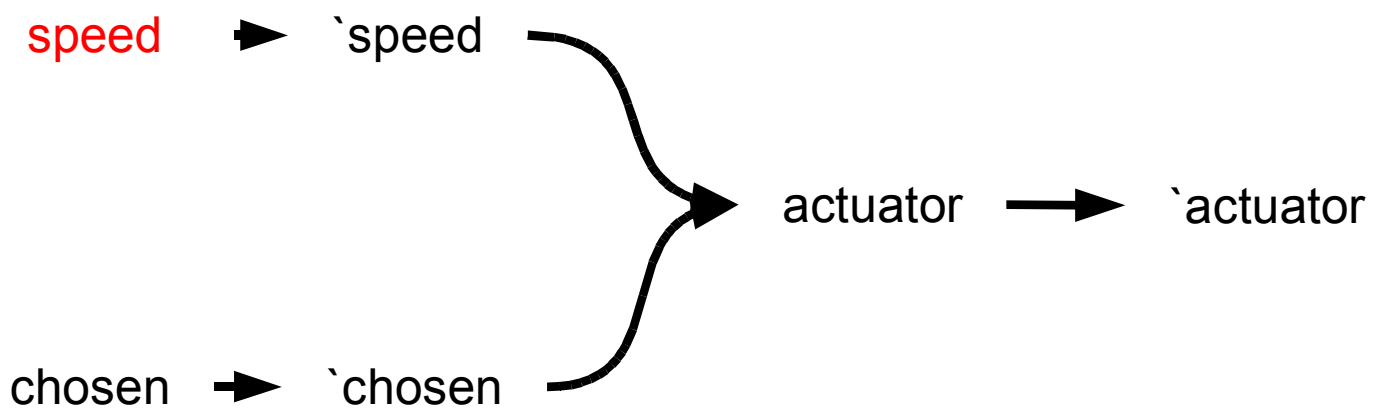


$$`speed \prec speed$$
$$`chosen \prec chosen$$
$$`actuator \prec actuator$$
$$actuator \prec control(`speed, `chosen)$$

Real time properties

- Properties on:
  - the variables
  - the architecture propagation paths
- Definitions based on:
  - updates instants of each variable along the execution
  - logical delay along a path
  - bounds on the differences between instants characterizing the variables behaviors
- Semantics:
  - sporadicity, periodicity
  - in each state, use of a temporally valid value of the source
  - bind the temporal delay along a path

Variables behaviors

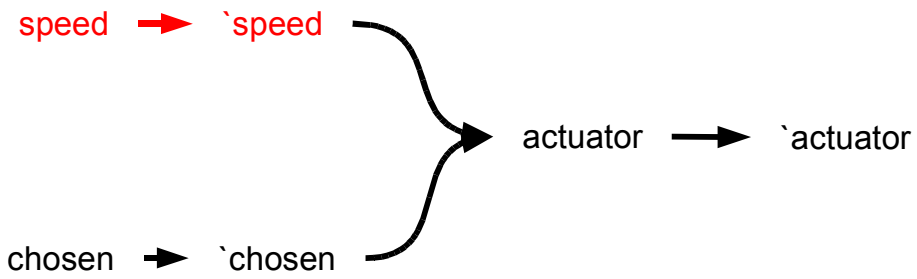


- *speed* $\{Sporadicity(\delta_1, \Delta_1)\}$
- *chosen* $\{Sporadicity(\delta_2, +\infty)\}$

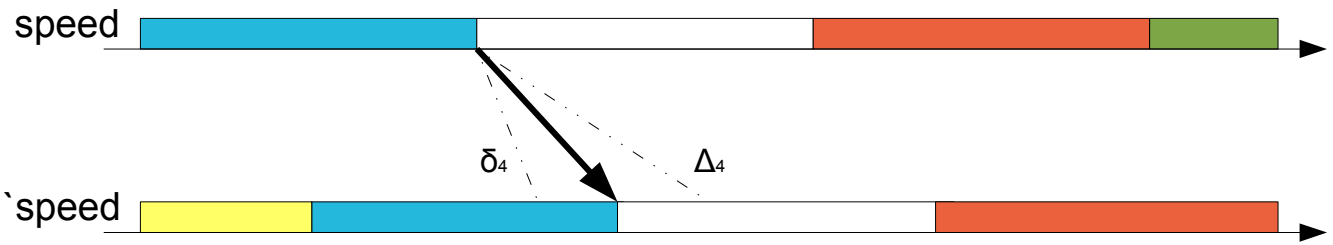
Bounds between two consecutive updates

Communications

$$speed \rightsquigarrow `speed \ \{Lag(\delta_4, \Delta_4)\}$$

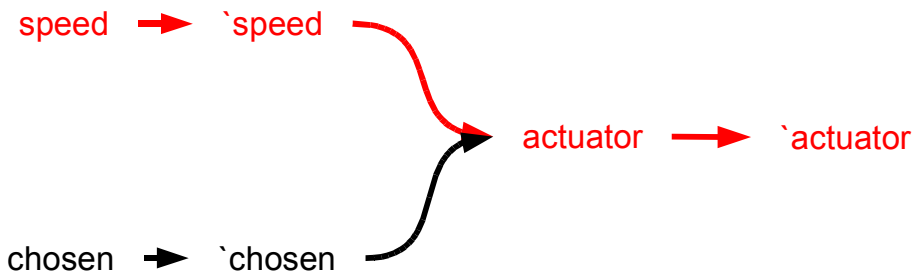


Bounds between the update of ‘*speed* and the corresponding update of *speed*

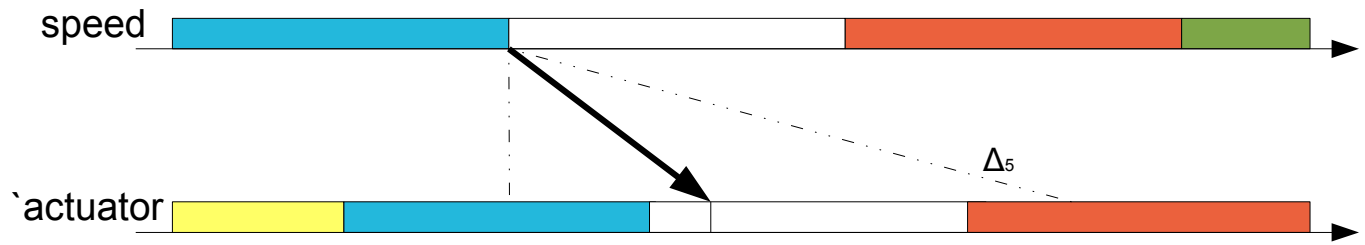


Global chains

$$speed \rightsquigarrow \text{`}actuator \ \{Latency(0, \Delta_5)\}$$

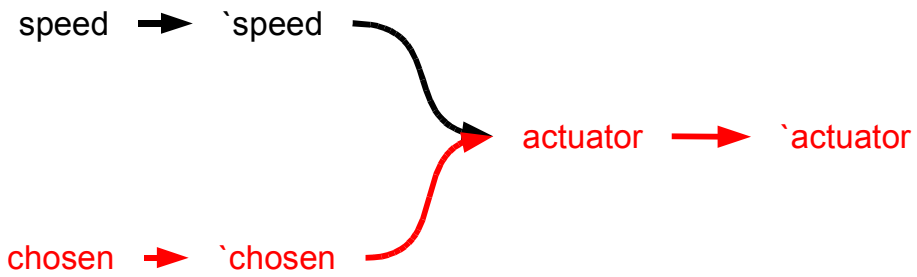


Bounds between the current value of ‘*actuator* and the corresponding update of *speed*

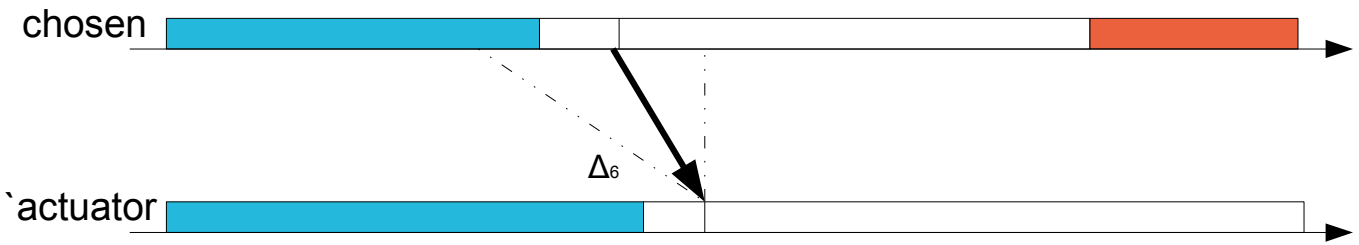


Global chains

$$chosen \rightsquigarrow \text{`}actuator\ \{Shift(0, \Delta_6)\}$$



Bounds between the current value of ‘*actuator* and the current value of *chosen*



Objectives and principles

Objectives:

- checking the specification consistency
- checking an implementation w.r.t. the specification

Analysis principles:

- build an equivalent state transition system
- exhibit a global period of the system
- build a finite system bisimilar with the specification using this period
- use this finite system to analyze the specification

Means:

- model checking

State transition system

Transition relation for each variable:

- introduction of a queue between images and sources
- definition of a temporal validity interval for the queue values:
  - depending on the variables behaviors
  - depending on paths real time properties
- global transition relation defined as
  the conjunction of the variable transition relations

Equivalent finite system:

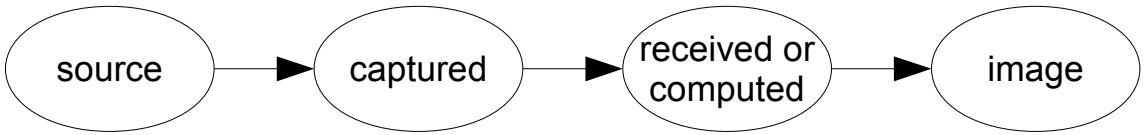
- existence condition:
  upper bound on all paths propagation times
- equivalence relation defined on an analysis interval
- finite number of states $\Rightarrow$ exploration of the executions in finite time

$$\text{Feasibility} = \text{existence of executions}$$

Implementation verification

Abstract model of an implementation:

- communication and computation:



- multiple queues
- extraction of each observation mechanism properties
- passing of values between the queues defined by the mechanism properties
- conjunction with the system defined by the specification

$$\text{Conformity} = \text{deadlock freedom}$$

Conclusion

Our approach:

- an approach focused on data
- description of the architecture as a set of relations between variables
- real time properties expressed on:
  - the variables updates
  - the propagation of values in the system
- specification and implementation analysis:
  - definition of a finite state transition system
  - model checking

Perspectives:

- generation of an implementation
- relation between data real time properties and task real time properties