Axis TEMPO

"Verification of timed systems"

AERES evaluation of LSV – 2 December 2013

Outline

Tempo as of 2008

Tempo in 2008 (AERES evaluation)

- **9 permanent researchers:**

D. Berwanger
CR CNRS

B. Bollig
CR CNRS

P. Bouyer
CR CNRS
team leader

Th. Chatain
MCF ENS Cachan

L. Fribourg
DR CNRS

P. Gastin
PU ENS Cachan

S. Haddad
PU ENS Cachan

N. Markey
CR CNRS

C. Picaronny
MCF ENS Cachan

- **8 PhD students**
- **1 post-doc**

Tempo as of 2009

Tempo in 2009 (creation of MExICo)

- **5 permanent researchers:**

D. Berwanger
CR CNRS

P. Bouyer
CR CNRS

L. Fribourg
DR CNRS

N. Markey
CR CNRS
team leader

C. Picaronny
MCF ENS Cachan

- **3 PhD students:**

N. Chamseddine

É. André

A. Da Costa

- **1 post-doc:**

D. Longuet

Tempo as of 30 June 2013

Tempo in June 2013

- **6 permanent researchers:**


D. Berwanger
CR CNRS


P. Bouyer
DR CNRS


L. Fribourg
DR CNRS


G. Lipari
Prof., S.Sup.Sant'Anna Pisa, Italy


N. Markey
CR CNRS
team leader


C. Picaronny
MCF ENS Cachan

- **3 PhD students:**


B. Barbot


J. Reichert


R. Soulat


M. Van den Bogaard

- **1 post-doc:**


Raj Mohan M.

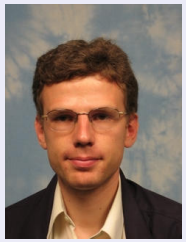
Tempo as of today

Tempo in 2013 (AERES evaluation)

- **6 permanent researchers:**

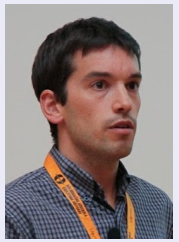

D. Berwanger
CR CNRS


P. Bouyer
DR CNRS


L. Doyen
CR CNRS

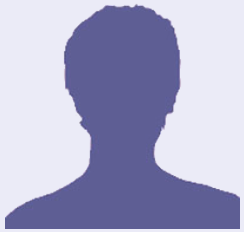

L. Fribourg
DR CNRS


G. Lipari
Prof., S.Sup.Sant'Anna Pisa, Italy


N. Markey
CR CNRS
team leader

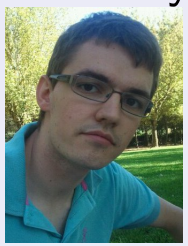
- **8 PhD students:**


P. Gardy


S. Mohamed


J. Reichert


M. Shirmohammadi


R. Soulat


D. Stan


Y. Sun


M. Van den Bogaard

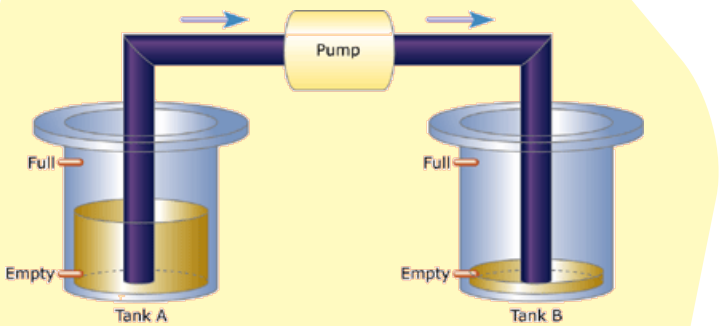
- **1 post-doc:**


Raj Mohan M.

Outline

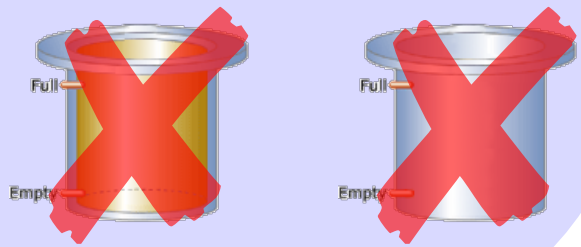
Research topics: model checking and synthesis

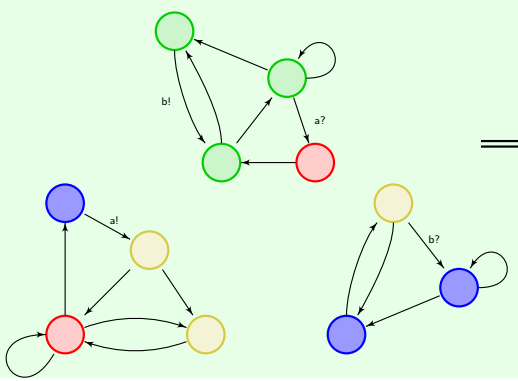
system:



[http://www.embedded.com]

property:





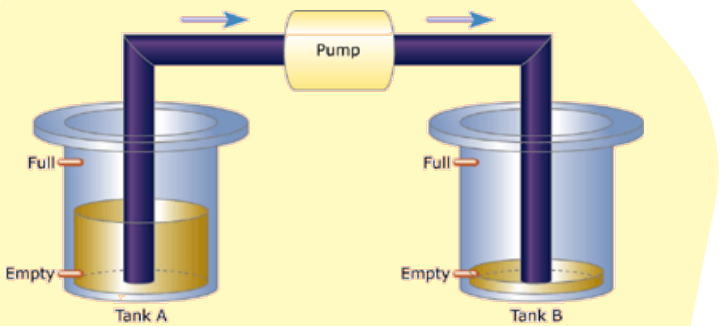
model-checking algorithm

$\mathbf{AG}(\neg\,\texttt{B.overfull} \wedge \neg\,\texttt{B.dried\_up})$

yes/no

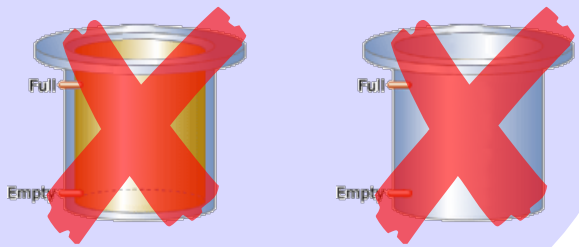
Research topics: model checking and synthesis

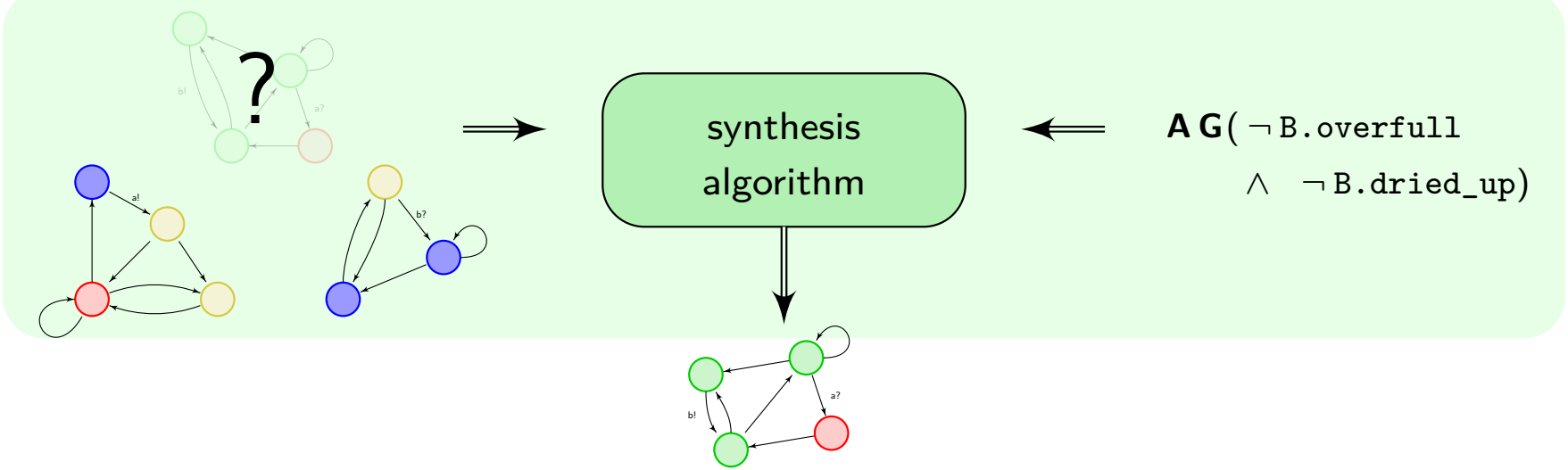
system:



[http://www.embedded.com]

property:





Embedded systems

en.wikipedia.org/wiki/Embedded_system#Characteristi

Create account Log in

Article Talk Read Edit View history Search

WIKIPEDIA
The Free Encyclopedia

Navigation
- Main page
- Contents
- Featured content
- Current events
- Random article
- Donate to Wikipedia

Interaction
- Help
- About Wikipedia
- Community portal
- Recent changes
- Contact page

Tools
- What links here
- Related changes
- Upload file
- Special pages
- Permanent link
- Page information
- Data item
- Cite this page

Print/export
- Create a book
- Download as PDF
- Printable version

Languages
- Alemannisch

Embedded system

From Wikipedia, the free encyclopedia

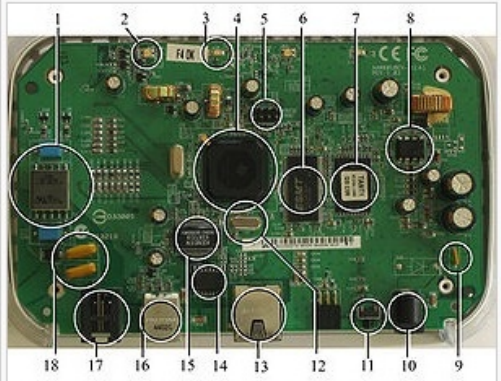
An **embedded system** is a computer system with a dedicated function within a larger mechanical or electrical system, often with real-time computing constraints.[1][2] It is *embedded* as part of a complete device often including hardware and mechanical parts. By contrast, a general-purpose computer, such as a personal computer (PC), is designed to be flexible and to meet a wide range of end-user needs. Embedded systems control many devices in common use today.[3]

Embedded systems contain processing cores that are either microcontrollers, or digital signal processors (DSP).[4]

A processor is an important unit in the embedded system hardware. It is the heart of the embedded system.[5]

The key characteristic, however, is being dedicated to handle a particular task. Since the embedded system is dedicated to specific tasks, design engineers can optimize it to reduce the size and cost of the product and increase the reliability and performance. Some embedded systems are mass-produced, benefiting from economies of scale.

Physically, embedded systems range from portable devices such as digital watches and MP3 players, to large stationary installations like traffic lights, factory controllers, and largely complex systems like hybrid vehicles, MRI, and avionics. Complexity varies from low, with a single microcontroller chip, to very high with multiple units, peripherals and networks mounted inside a large chassis or enclosure.



Picture of the internals of an ADSL modem/router. A modern example of an embedded system. Labelled parts include a microprocessor (4), RAM (6), and flash memory (7).

Characteristics [edit]

Embedded systems are designed to do some specific task, rather than be a general-purpose computer for multiple tasks. Some also have real-time performance constraints that must be met, for reasons such as safety and usability; others may have low or no performance requirements, allowing the system hardware to be simplified to reduce costs.

Embedded systems are not always standalone devices. Many embedded systems consist of small, computerized parts within a larger device that serves a more general purpose. For example, the Gibson Robot Guitar features an embedded system for tuning the strings, but the overall purpose of the Robot Guitar is, of course, to play music.[8] Similarly, an embedded system in an automobile provides a specific function as a subsystem of the car itself.

The program instructions written for embedded systems are referred to as firmware, and are stored in read-only memory or Flash memory chips. They run with limited computer hardware resources: little memory, small or non-existent keyboard or screen.

Embedded systems

en.wikipedia.org/wiki/Embedded_system#Characteristi

Create account Log in

WIKIPEDIA
The Free Encyclopedia

Article Talk | Read Edit View history | Search

Embedded system

From Wikipedia, the free encyclopedia

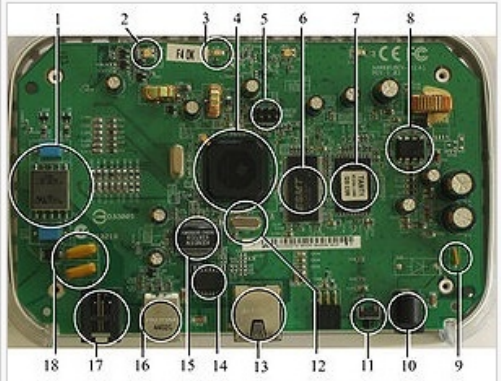
An **embedded system** is a computer system with a dedicated function within a larger mechanical or electrical system, often with real-time computing constraints.[1][2] It is *embedded* as part of a complete device often including hardware and mechanical parts. By contrast, a general-purpose computer, such as a personal computer (PC), is designed to be flexible and to meet a wide range of end-user needs. Embedded systems control many devices in common use today.[3]

Embedded systems contain processing cores that are either microcontrollers, or digital signal processors (DSP).[4]

A processor is an important unit in the embedded system hardware. It is the heart of the embedded system.[5]

The key characteristic, however, is being dedicated to handle a particular task. Since the embedded system is dedicated to specific tasks, design engineers can optimize it to reduce the size and cost of the product and increase the reliability and performance. Some embedded systems are mass-produced, benefiting from economies of scale.

Physically, embedded systems range from portable devices such as digital watches and MP3 players, to large stationary installations like traffic lights, factory controllers, and largely complex systems like hybrid vehicles, MRI, and avionics. Complexity varies from low, with a single microcontroller chip, to very high with multiple units, peripherals and networks mounted inside a large chassis or enclosure.



Picture of the internals of an ADSL modem/router. A modern example of an embedded system. Labelled parts include a microprocessor (4), RAM (6), and flash memory (7).

Navigation
- Main page
- Contents
- Featured content
- Current events
- Random article
- Donate to Wikipedia

Interaction
- Help
- About Wikipedia
- Community portal
- Recent changes
- Contact page

Tools
- What links here
- Related changes
- Upload file
- Special pages
- Permanent link
- Page information
- Data item
- Cite this page

Print/export
- Create a book
- Download as PDF
- Printable version

Languages
- Alemannisch

Characteristics [edit]

Embedded systems are designed to do some specific task, rather than be a general-purpose computer for multiple tasks. Some also have real-time performance constraints that must be met, for reasons such as safety and usability; others may have low or no performance requirements, allowing the system hardware to be simplified to reduce costs.

Embedded systems are not always standalone devices. Many embedded systems consist of small, computerized parts within a larger device that serves a more general purpose. For example, the Gibson Robot Guitar features an embedded system for tuning the strings, but the overall purpose of the Robot Guitar is, of course, to play music.[8] Similarly, an embedded system in an automobile provides a specific function as a subsystem of the car itself.

The program instructions written for embedded systems are referred to as firmware, and are stored in read-only memory or Flash memory chips. They run with limited computer hardware resources: little memory, small or non-existent keyboard or screen.

Research topics: model checking and synthesis

Verification of timed and hybrid automata

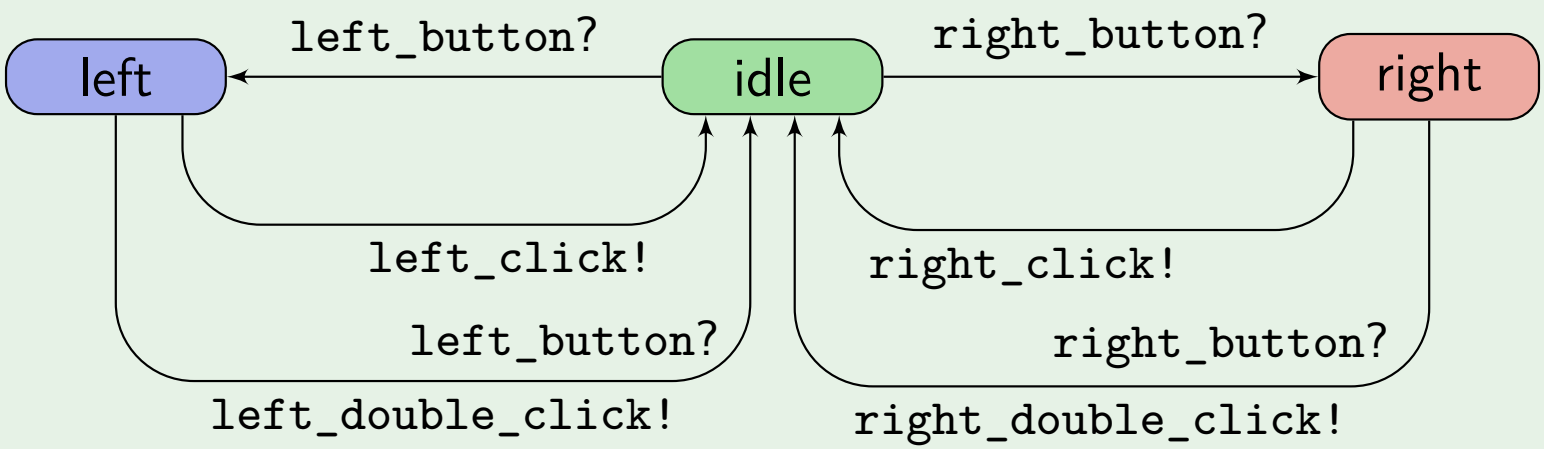
- robustness issues in timed automata
- parameter synthesis for timed and hybrid systems
- modelling resources in real-time systems

Games for synthesis of complex systems

- temporal logics for games
- equilibria in non-zero-sum games
- games with partial observation

1. Reasoning about real-time systems

Example (A computer mouse)



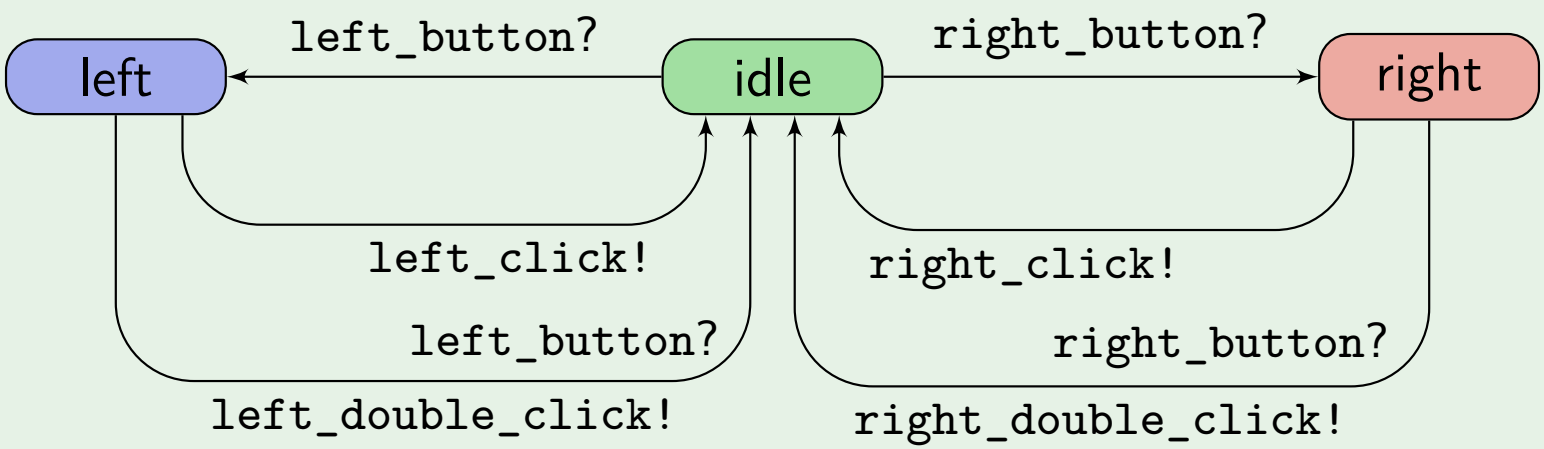
1. Reasoning about real-time systems

Timed automata

A timed automaton is made of

- a transition system,

Example (A computer mouse)



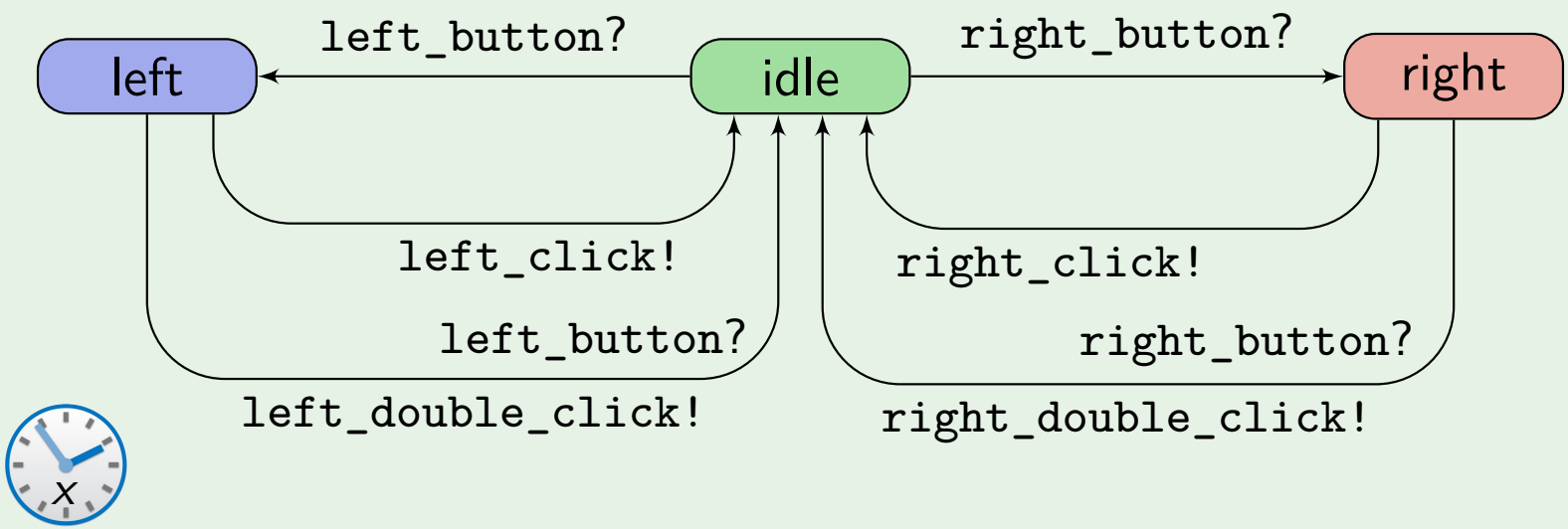
1. Reasoning about real-time systems

Timed automata

A timed automaton is made of

- a transition system,
- a set of clocks,

Example (A computer mouse)



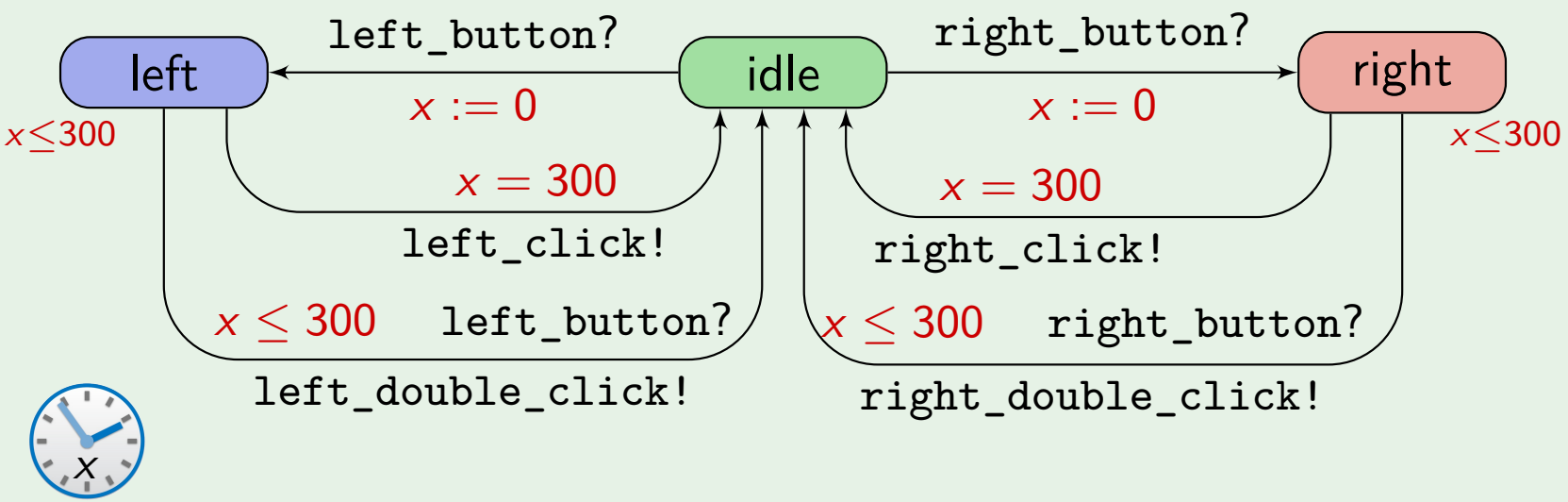
1. Reasoning about real-time systems

Timed automata

A timed automaton is made of

- a transition system,
- a set of clocks,
- timing constraints on states and transitions.

Example (A computer mouse)



1.1 Robustness issues in timed automata

Timed automata vs real-time systems

- timed automata use *real-valued clocks*,
- physical systems are *digital*.

$\rightsquigarrow$ the possible (very small) delay between the evaluation of a guard and the effective transition is not modelled.

Properties proven to hold on a model
might fail to hold on its implementation.

1.1 Robustness issues in timed automata

Several approaches

- **guard enlargement:** to model the imprecisions

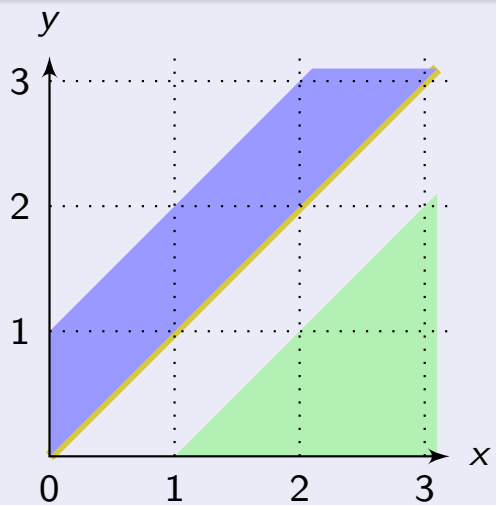
  a transition can be taken at any time in $[t - \delta; t + \delta]$.

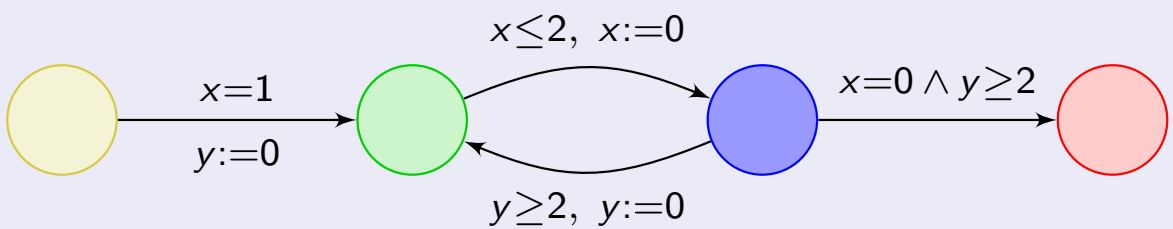
1.1 Robustness issues in timed automata

Several approaches

- **guard enlargement:** to model the imprecisions

a transition can be taken at any time in $[t-\delta; t+\delta]$.



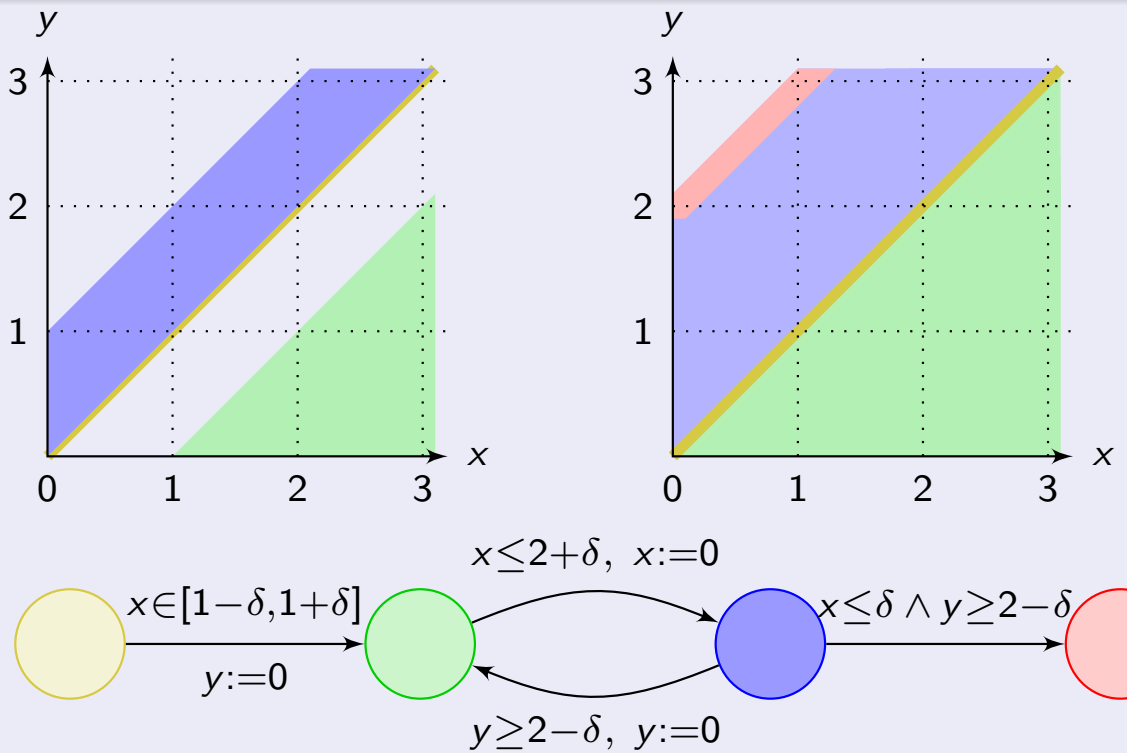


1.1 Robustness issues in timed automata

Several approaches

- **guard enlargement:** to model the imprecisions

a transition can be taken at any time in $[t-\delta; t+\delta]$.

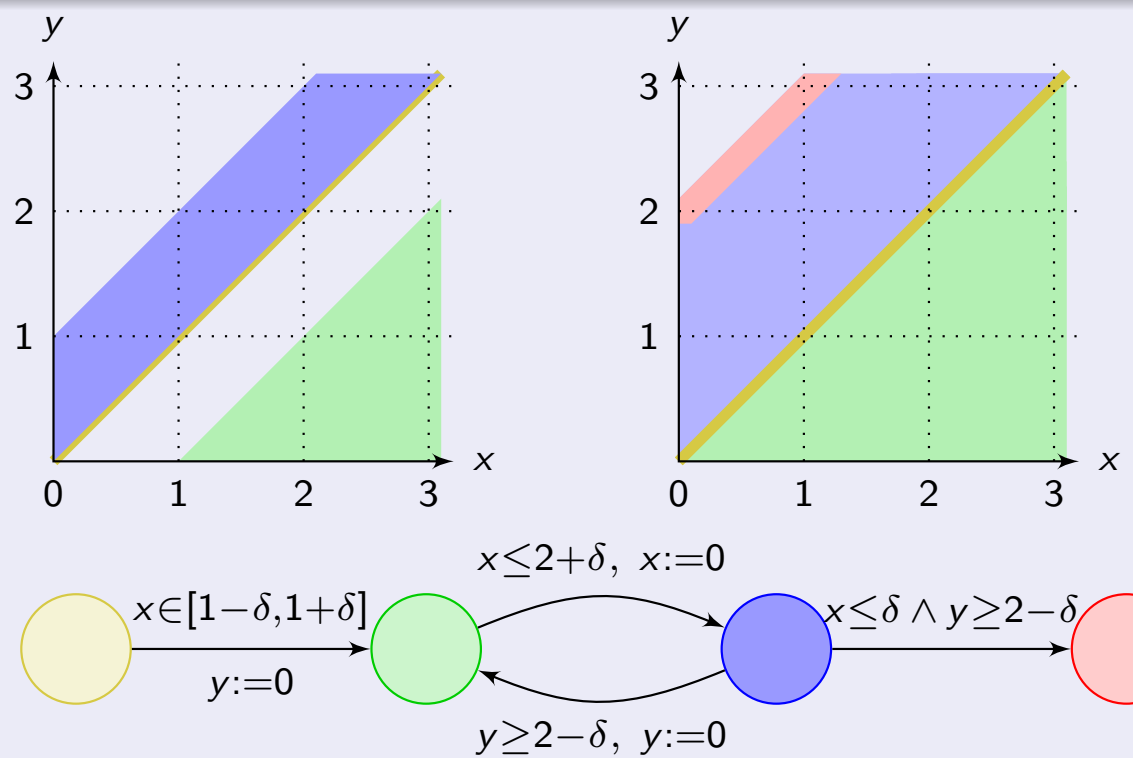


1.1 Robustness issues in timed automata

Several approaches

- **guard enlargement:** to model the imprecisions

a transition can be taken at any time in $[t-\delta; t+\delta]$.



$\rightsquigarrow$ Parametric robust safety is decidable. [FMSD'08, FORMATS'11]

1.1 Robustness issues in timed automata

Several approaches

- **guard enlargement:** to model the imprecisions
- **shrinking:** to counteract enlargement

guards $[a, b]$ are replaced with $[a + \delta, b - \delta]$.

1.1 Robustness issues in timed automata

Several approaches

- **guard enlargement:** to model the imprecisions
- **shrinking:** to counteract enlargement

guards $[a, b]$ are replaced with $[a + \delta, b - \delta]$.

A timed automaton is *shrinkable* if its shrunk automaton contains the (time-abstracted) behaviours of the original automaton.

$\rightsquigarrow$ Shrinkability is decidable. [FSTTCS'11]

$\rightsquigarrow$ implementation of a prototype (Shrinktech) [CAV'13]

1.1 Robustness issues in timed automata

Several approaches

- **guard enlargement:** to model the imprecisions
- **shrinking:** to counteract enlargement
- **robust synthesis**: to react to perturbations

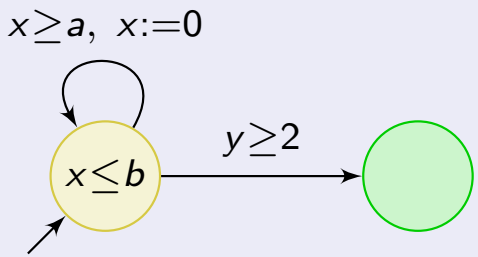
the controller selects a delay d
this delay is perturbed by at most δ.

$\rightsquigarrow$ Parametric robust (repeated) reachability is decidable.

[ICALP'12, CONCUR'13]

1.2 Parameter synthesis for timed and hybrid automata

Timed automata with parameters in clock constraints

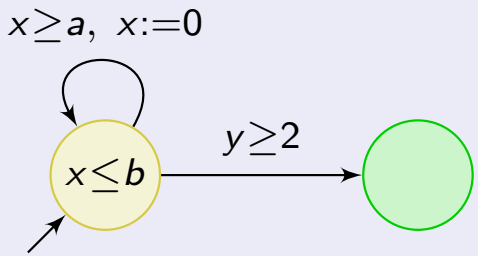


Inverse method

generalise a given valuation of the parameters.

1.2 Parameter synthesis for timed and hybrid automata

Timed automata with parameters in clock constraints



Inverse method

generalise a given valuation of the parameters.

$\rightsquigarrow$ algorithms for computing regions of good parameter valuations [RP'08,RP'10,RP'11,NFM'12]

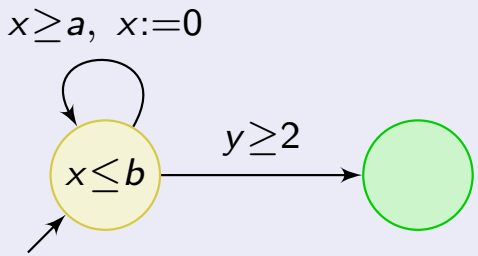
$\rightsquigarrow$ development of a tool: Imitator [ICTAC'09,INFINITY'10]

$\rightsquigarrow$ applications to circuits, scheduling, ... [TIME'12,NCMIP'13,FTSCS'13]

$\rightsquigarrow$ a book:

1.2 Parameter synthesis for timed and hybrid automata

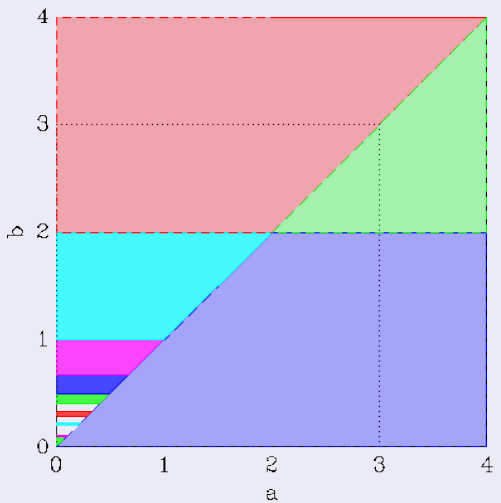
Timed automata with parameters in clock constraints



Inverse method

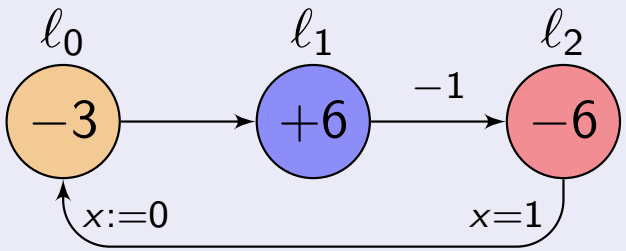
generalise a given valuation of the parameters.

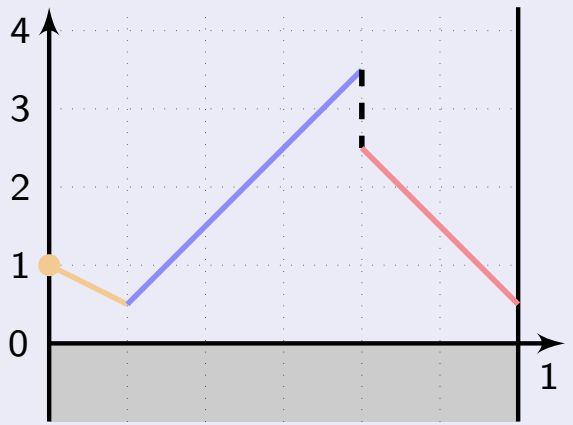
```
% ./imitator example.imi example.v0
   -mode cover -cart -step 1/10
  [...]
  11 different constraints were computed.
  [...]
  Global time spent :  0.392519950867 s
%
```



1.3 Modelling resources in (real-time) systems

Energy constraints in weighted timed automata





Hybrid variables cannot be used in guards.

$\rightsquigarrow$ Reachability under (lower-bound) energy constraints is

- unecidable in timed automata with four clocks;
- decidable in timed automata with one clock.

[FORMATS'08,HSCC'10,QEST'12, Comm.ACM'11]

Research topics: model checking and synthesis

Verification of timed and hybrid automata

- robustness issues in timed automata
- parameter synthesis for timed and hybrid systems
- modelling resources in real-time systems

Games for synthesis of complex systems

- temporal logics for games
- equilibria in non-zero-sum games
- games with partial observation

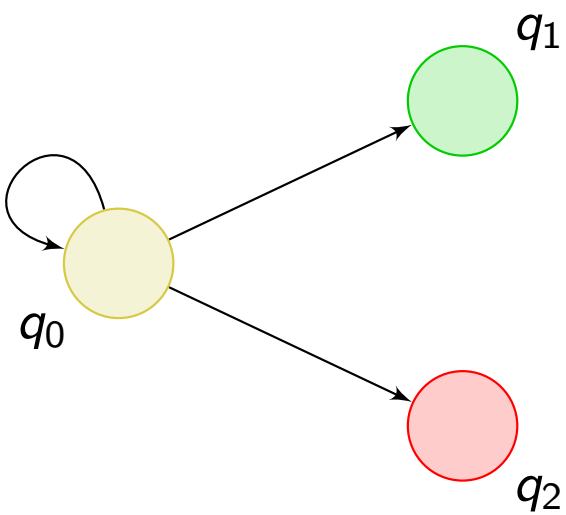
2. Games for synthesis

2. Games for synthesis

Concurrent games

A concurrent game is made of

- a transition system,

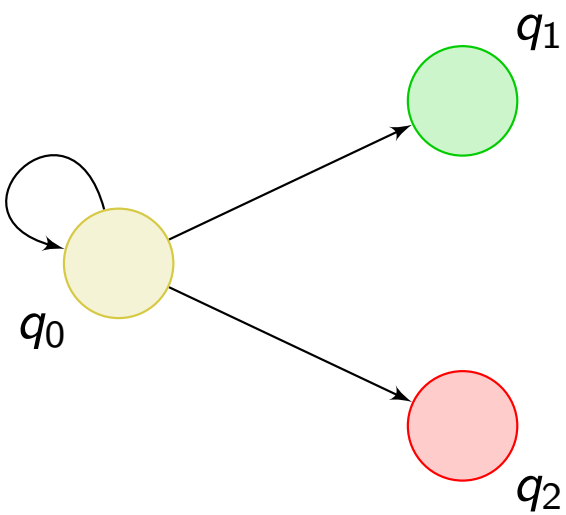


2. Games for synthesis

Concurrent games

A concurrent game is made of

- a transition system,
- a set of agents,

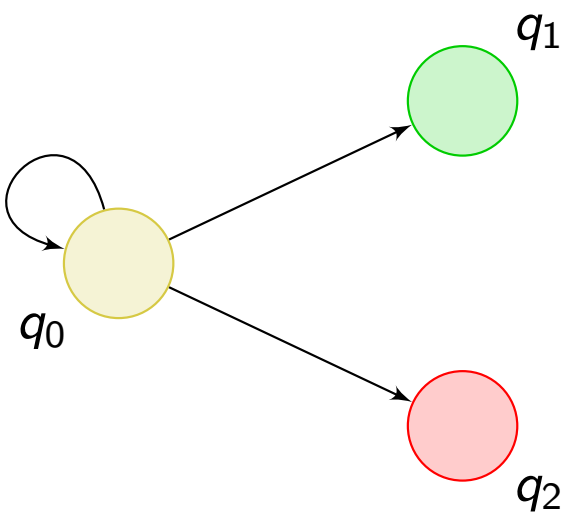


2. Games for synthesis

Concurrent games

A concurrent game is made of

- a transition system,
- a set of agents,
- a transition table indicating the effect of the actions of the players.



player 2 \ player 1			
	q_0	q_2	q_1
	q_1	q_0	q_2
	q_2	q_1	q_0

2.1 Temporal logics for games

Alternating-time temporal logics

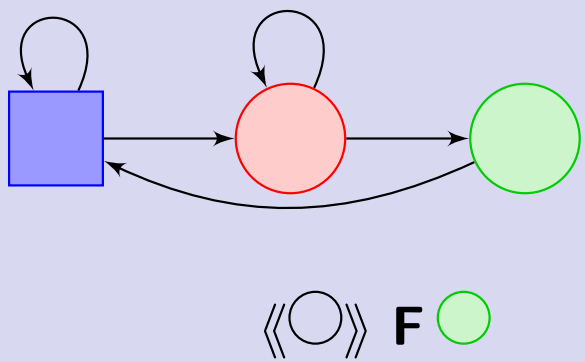
ATL extends CTL with strategy quantifiers.

2.1 Temporal logics for games

Alternating-time temporal logics

ATL extends CTL with strategy quantifiers.

Classical semantics

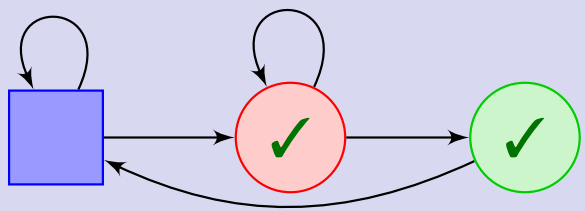


2.1 Temporal logics for games

Alternating-time temporal logics

ATL extends CTL with strategy quantifiers.

Classical semantics



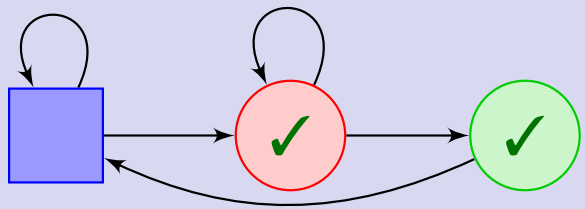
$\langle\langle \bigcirc \rangle\rangle$ **F** ○

2.1 Temporal logics for games

Alternating-time temporal logics

ATL extends CTL with strategy quantifiers.

Classical semantics



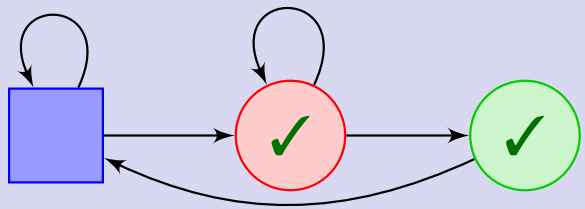
$\langle\langle \Box \rangle\rangle$ **G**($\langle\langle \bigcirc \rangle\rangle$ **F** ◯)

2.1 Temporal logics for games

Alternating-time temporal logics

ATL extends CTL with strategy quantifiers.

Classical semantics



$\langle\langle\Box\rangle\rangle$ **G**($\langle\langle\bigcirc\rangle\rangle$ **F** ●)

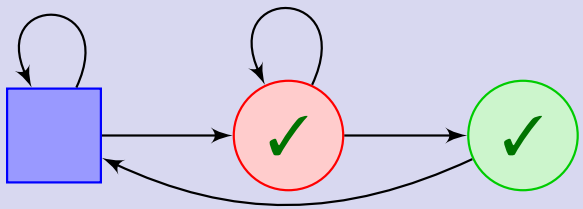
Model checking is in PTIME.

2.1 Temporal logics for games

Alternating-time temporal logics

ATL extends CTL with strategy quantifiers.

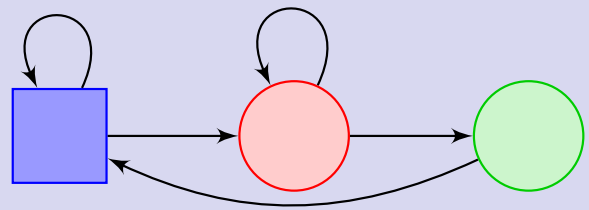
Classical semantics



$\langle\langle \Box \rangle\rangle$ **G**($\langle\langle \bigcirc \rangle\rangle$ **F**)

Model checking is in PTIME.

ATL with strategy contexts



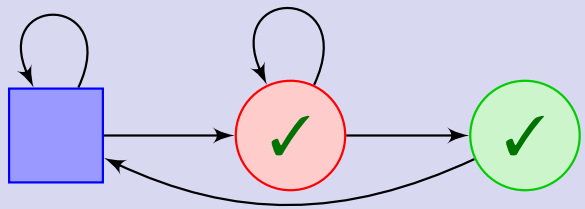
$\langle \cdot \Box \cdot \rangle$ **G**($\langle \cdot \bigcirc \cdot \rangle$ **F**)

2.1 Temporal logics for games

Alternating-time temporal logics

ATL extends CTL with strategy quantifiers.

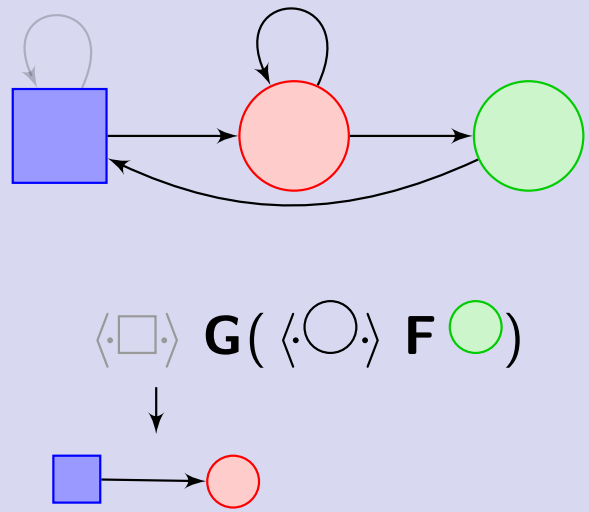
Classical semantics



$\langle\langle \Box \rangle\rangle$ **G**($\langle\langle \bigcirc \rangle\rangle$ **F**)

Model checking is in PTIME.

ATL with strategy contexts



$\langle \cdot \Box \cdot \rangle$ **G**($\langle \cdot \bigcirc \cdot \rangle$ **F**)

2.1 Temporal logics for games

Alternating-time temporal logics

ATL extends CTL with strategy quantifiers.

ATL with strategy contexts

- **access control to a shared resource:**

$$\langle\cdot \mathit{server} \cdot\rangle\, \mathbf{G} \left[\texttt{mutual exclusion} \wedge \bigwedge_{\mathit{client}} \langle\cdot \mathit{client} \cdot\rangle\, \mathbf{F}\, \texttt{access} \right]$$

2.1 Temporal logics for games

Alternating-time temporal logics

ATL extends CTL with strategy quantifiers.

ATL with strategy contexts

- **access control to a shared resource:**

$$\langle\cdot \mathit{server} \cdot\rangle\ \mathbf{G}\left[\texttt{mutual exclusion} \wedge \bigwedge_{\mathit{client}} \langle\cdot \mathit{client} \cdot\rangle\ \mathbf{F}\,\texttt{access}\right]$$

- **existence of a Nash equilibrium:**

$$\langle\cdot p_1, p_2, \ldots, p_n \cdot\rangle\left[\bigwedge_i (\langle\cdot p_i \cdot\rangle\, goal_i) \Rightarrow goal_i\right]$$

2.1 Temporal logics for games

Alternating-time temporal logics

ATL extends CTL with strategy quantifiers.

$\rightsquigarrow$ ATL_{sc} is much more expressive than ATL, and well-suited for describing non-zero-sum objectives [LFCS'09]

$\rightsquigarrow$ model checking ATL_{sc} is decidable, and k-EXPTIME-complete when limited to k nested quantifier [FSTTCS'10,CONCUR'12]

$\rightsquigarrow$ satisfiability is undecidable; it is decidable when restricted to turn-based games. [GandALF'13]

2.2 Equilibria in non-zero-sum games

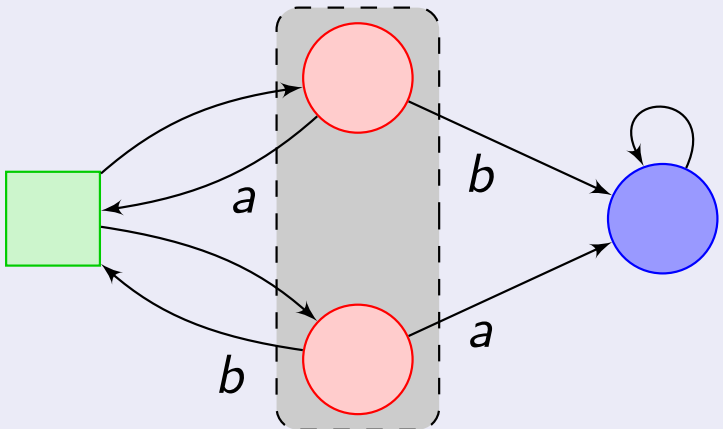
Nash equilibria

No single player can improve her payoff by changing her strategy

- $\rightsquigarrow$ algorithm for computing Nash equilibria in timed games [FORMATS'10,CONCUR'10]
- $\rightsquigarrow$ reduction to a two-player zero-sum game [FSTTCS'11,FoSSaCS'12]
- $\rightsquigarrow$ implementation of a prototype (Praline) [CAV'13]

2.3 Games with imperfect information

Games with imperfect information



strategies may only depend on the sequence of observations

- $\rightsquigarrow$ antichain-based algorithm for parity games with imperfect information, implemented in the tool Alpaga [CONCUR'08,TACAS'09,I&C'10]
- $\rightsquigarrow$ polynomial-time reduction from parity games to safety games under imperfect information [FSTTCS'08]
- $\rightsquigarrow$ application of imperfect information games for solving counter parity games [MFCS'12]

Outline

How we addressed the recommendations of AERES 2009

Preserve interactions between Tempo and MExICo:

- joint Tempo/MExICo *groupe de travail*;
- several Tempo/MExICo collaborations; B. Barbot PhD. thesis;
- papers on timed stochastic systems (P. Bouyer, S. Haddad), on timed distributed systems (S. Balaguer, S. Akshay).

How we addressed the recommendations of AERES 2009

Preserve interactions between Tempo and MExICo:

- joint Tempo/MExICo *groupe de travail*;
- several Tempo/MExICo collaborations; B. Barbot PhD. thesis;
- papers on timed stochastic systems (P. Bouyer, S. Haddad), on timed distributed systems (S. Balaguer, S. Akshay).

Develop prototypes and tools:

- tool Imitator, applications to several cases;
- tool Minimator, application to power electronics;
- several prototypes: alpaga, praline, shrinktech.

How we addressed the recommendations of AERES 2009

Preserve interactions between Tempo and MExICo:

- joint Tempo/MExICo *groupe de travail*;
- several Tempo/MExICo collaborations; B. Barbot PhD. thesis;
- papers on timed stochastic systems (P. Bouyer, S. Haddad), on timed distributed systems (S. Balaguer, S. Akshay).

Develop prototypes and tools:

- tool Imitator, applications to several cases;
- tool Minimator, application to power electronics;
- several prototypes: alpaga, praline, shrinktech.

Develop original and ambitious research topics:

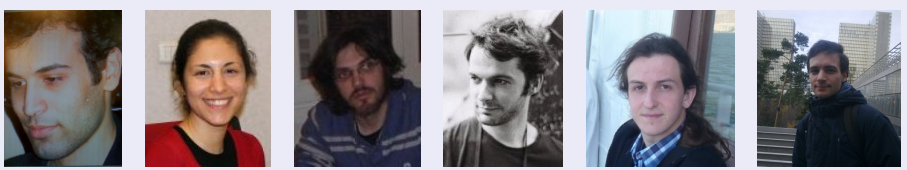
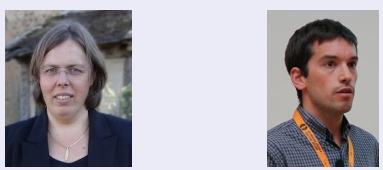
- framework for robust verification of timed automata;
- non-zero-sum games for synthesis of complex systems;
- development of algorithms and a tool suite for parameter synthesis.

Publications of TEMPO for 2008-2013

Publications

Number of publications	139
books, edited books, chapters in books	11
articles in int. journals	33
articles in int. conferences	72
other publications	23

PhD and habilitations defended

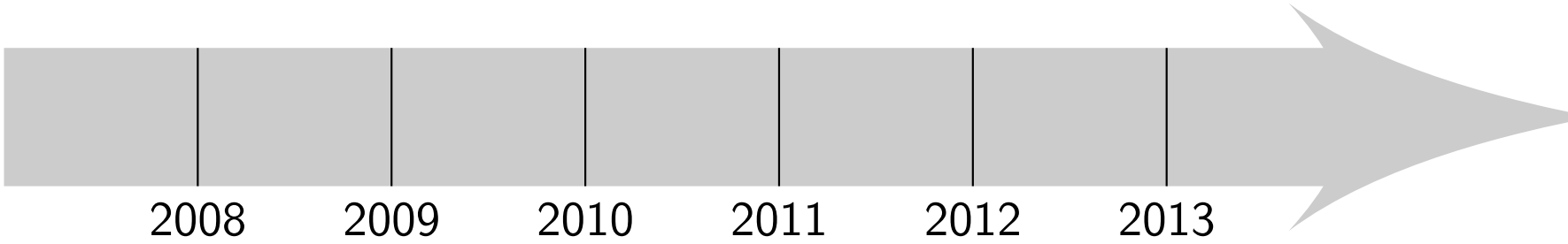
PhD theses	
Habilitation theses	

Highlights of TEMPO for 2008-2013

Highlights

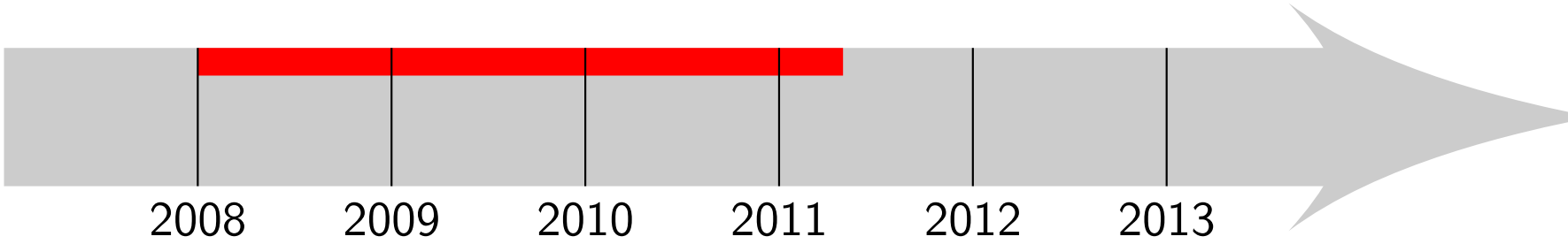
- **Award:**
  - Patricia Bouyer receives the EATCS Presburger Award in 2011
- **Invited talks and tutorials:**
  - MOVEP'08, GAMES'08, WATA'10, QMC'10, SIES'11, RP'13
- **Organization of conferences:**
  - TIME'10 (Paris, Sep. 2010)
  - GAMES'11 (Paris, Sep. 2011)
  - FORMATS'13 (Buenos Aires, Aug. 2013)
  - HIGHLIGHTS'13 (Paris, Sep. 2013)
  - P. Bouyer is workshop chair for LICS (2013-2015)
- **Long-term visitors:**
  - Giuseppe Lipari (Marie-Curie RBUCE-UP chair, 2 year)
  - Jörg Olschewski, Claus Thrane (PhD students, 6 months)

Research contracts



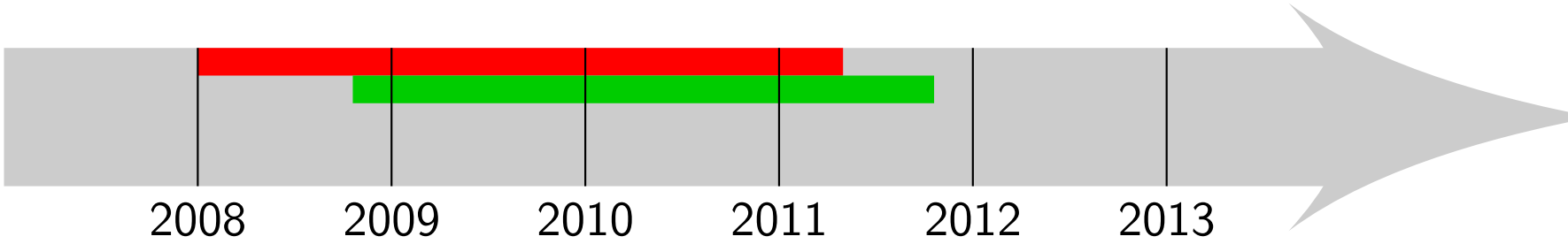
- STREP Quasimodo: quantitative verification
  Academic partners: Aalborg (coord), Aachen, LSV, Saarbrücken, Twente, ULB
  Industrial partners: Hydac (D), Chess (D), Terma (DK).
- ESF Gasics: games for synthesis
  Partners: ULB (coord), Aalborg, Aachen, LSV, Warwick
- ANR DOTS: distributed, timed, open systems
  Partners: LSV (coord), IRCCyN, IRISA, LaBRI, LAMSADE
- ANR Valmem: verification of memory circuits
  Academic partner: LSV (coord), LIP6
  Industrial partner: STMicroelectronics
- Farman projects: TOAST, SIMOP, EMOTICON, CRAFT, BOOST, ...

Research contracts



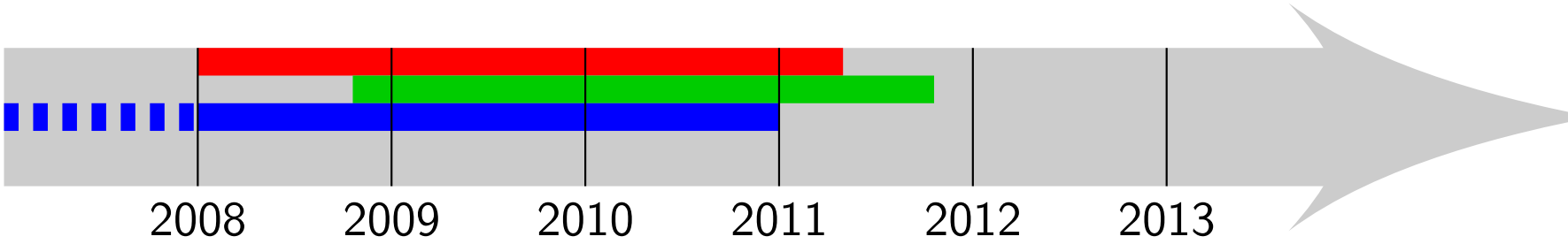
- STREP Quasimodo: quantitative verification
  Academic partners: Aalborg (coord), Aachen, LSV, Saarbrücken, Twente, ULB
  Industrial partners: Hydac (D), Chess (D), Terma (DK).
- ESF Gasics: games for synthesis
  Partners: ULB (coord), Aalborg, Aachen, LSV, Warwick
- ANR DOTS: distributed, timed, open systems
  Partners: LSV (coord), IRCCyN, IRISA, LaBRI, LAMSADE
- ANR Valmem: verification of memory circuits
  Academic partner: LSV (coord), LIP6
  Industrial partner: STMicroelectronics
- Farman projects: TOAST, SIMOP, EMOTICON, CRAFT, BOOST, ...

Research contracts



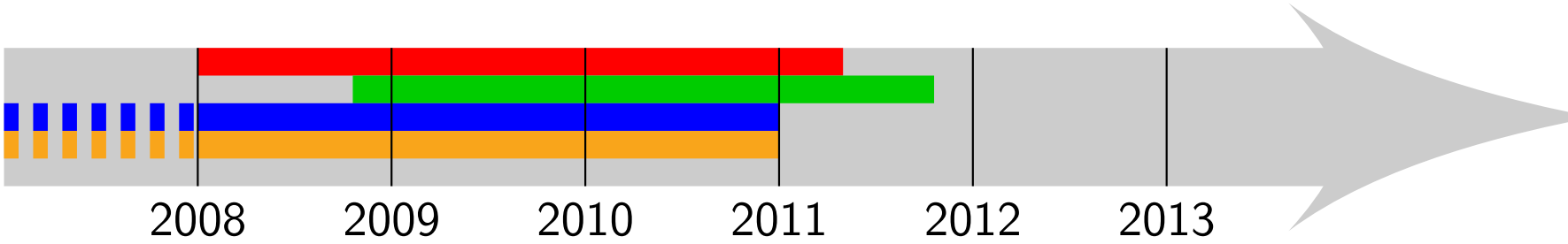
- STREP Quasimodo: quantitative verification
  Academic partners: Aalborg (coord), Aachen, LSV, Saarbrücken, Twente, ULB
  Industrial partners: Hydac (D), Chess (D), Terma (DK).
- ESF Gasics: games for synthesis
  Partners: ULB (coord), Aalborg, Aachen, LSV, Warwick
- ANR DOTS: distributed, timed, open systems
  Partners: LSV (coord), IRCCyN, IRISA, LaBRI, LAMSADE
- ANR Valmem: verification of memory circuits
  Academic partner: LSV (coord), LIP6
  Industrial partner: STMicroelectronics
- Farman projects: TOAST, SIMOP, EMOTICON, CRAFT, BOOST, ...

Research contracts



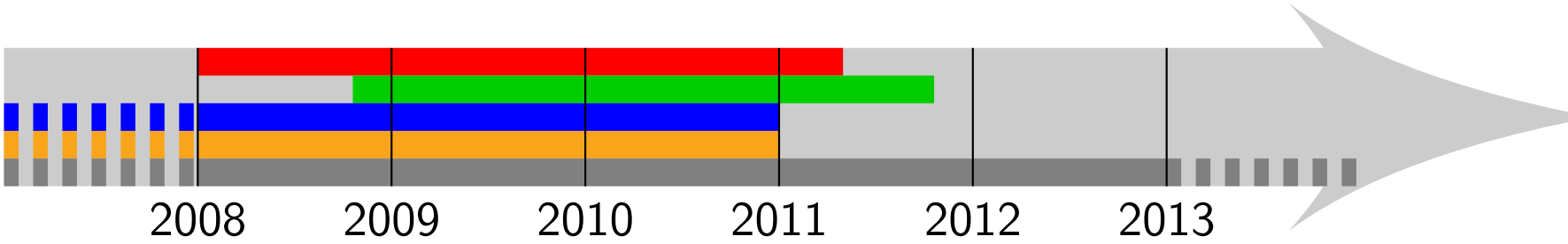
- STREP Quasimodo: quantitative verification
  Academic partners: Aalborg (coord), Aachen, LSV, Saarbrücken, Twente, ULB
  Industrial partners: Hydac (D), Chess (D), Terma (DK).
- ESF Gasics: games for synthesis
  Partners: ULB (coord), Aalborg, Aachen, LSV, Warwick
- ANR DOTS: distributed, timed, open systems
  Partners: LSV (coord), IRCCyN, IRISA, LaBRI, LAMSADE
- ANR Valmem: verification of memory circuits
  Academic partner: LSV (coord), LIP6
  Industrial partner: STMicroelectronics
- Farman projects: TOAST, SIMOP, EMOTICON, CRAFT, BOOST, ...

Research contracts



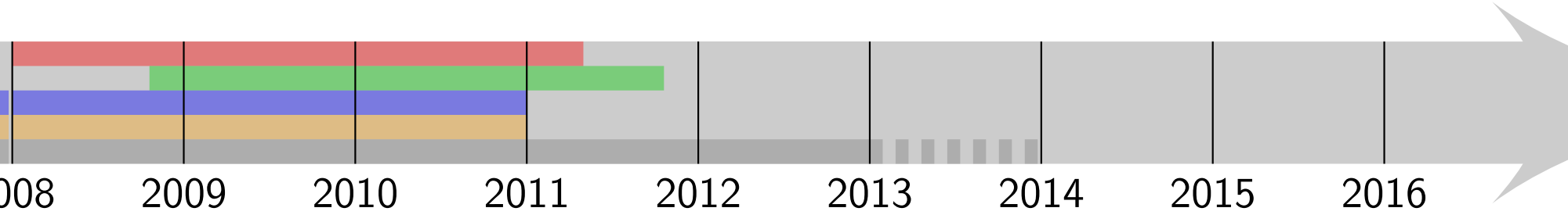
- STREP Quasimodo: quantitative verification
  Academic partners: Aalborg (coord), Aachen, LSV, Saarbrücken, Twente, ULB
  Industrial partners: Hydac (D), Chess (D), Terma (DK).
- ESF Gasics: games for synthesis
  Partners: ULB (coord), Aalborg, Aachen, LSV, Warwick
- ANR DOTS: distributed, timed, open systems
  Partners: LSV (coord), IRCCyN, IRISA, LaBRI, LAMSADE
- ANR Valmem: verification of memory circuits
  Academic partner: LSV (coord), LIP6
  Industrial partner: STMicroelectronics
- Farman projects: TOAST, SIMOP, EMOTICON, CRAFT, BOOST, ...

Research contracts



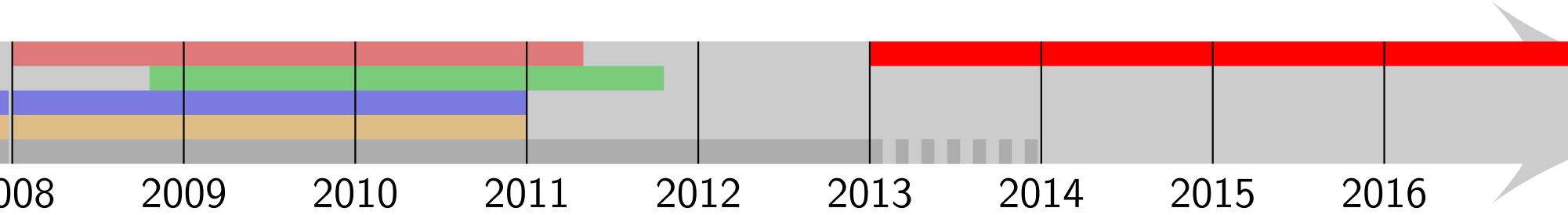
- STREP Quasimodo: quantitative verification
  Academic partners: Aalborg (coord), Aachen, LSV, Saarbrücken, Twente, ULB
  Industrial partners: Hydac (D), Chess (D), Terma (DK).
- ESF Gasics: games for synthesis
  Partners: ULB (coord), Aalborg, Aachen, LSV, Warwick
- ANR DOTS: distributed, timed, open systems
  Partners: LSV (coord), IRCCyN, IRISA, LaBRI, LAMSADE
- ANR Valmem: verification of memory circuits
  Academic partner: LSV (coord), LIP6
  Industrial partner: STMicroelectronics
- Farman projects: TOAST, SIMOP, EMOTICON, CRAFT, BOOST, ...

Research contracts



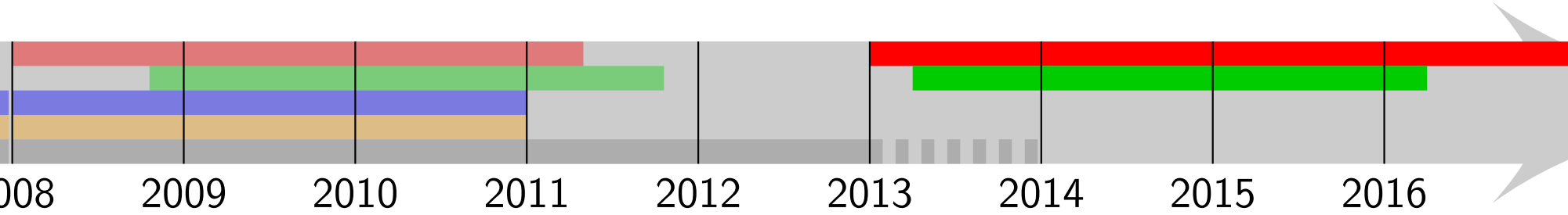
- ERC EQualIS: quantitative verification of complex systems
  Principal Investigator: Patricia Bouyer
- STREP Cassting: non-zero-sum games for synthesis
  Academic partners: LSV (coord), Aachen, Aalborg, ULB, U.MONS
  Industrial partners: Seluxit (DK), Energi Nord (DK)
- ANR ImpRo: robustness of timd systems
  Partners: IRCCyN (coord), IRISA, LIP6, LSV
- Digiteo project SIMS: controller synthesis for switched systems
  Partners: CEA-List (coord), LSV
- Farman project: BOOST2, ROSCOV, INVERSYM, DECORR

Research contracts



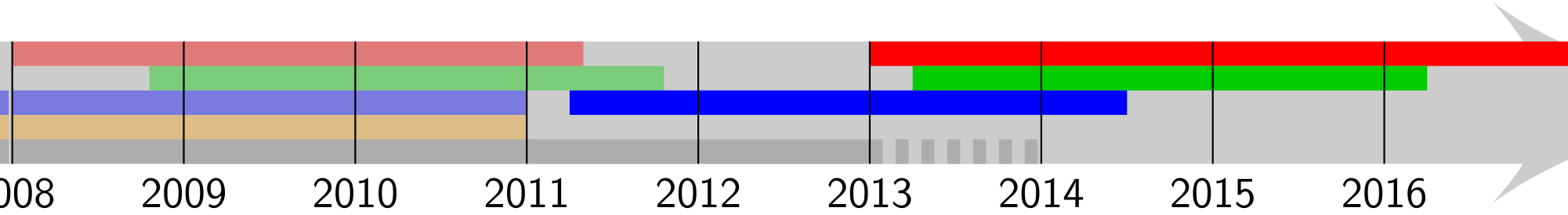
- ERC EQualIS: quantitative verification of complex systems
  Principal Investigator: Patricia Bouyer
- STREP Cassting: non-zero-sum games for synthesis
  Academic partners: LSV (coord), Aachen, Aalborg, ULB, U.MONS
  Industrial partners: Seluxit (DK), Energi Nord (DK)
- ANR ImpRo: robustness of timd systems
  Partners: IRCCyN (coord), IRISA, LIP6, LSV
- Digiteo project SIMS: controller synthesis for switched systems
  Partners: CEA-List (coord), LSV
- Farman project: BOOST2, ROSCOV, INVERSYM, DECORR

Research contracts



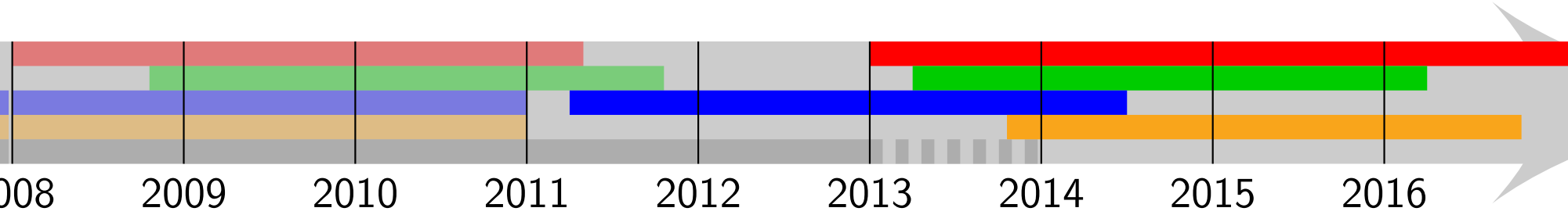
- ERC EQualIS: quantitative verification of complex systems
  Principal Investigator: Patricia Bouyer
- STREP Cassting: non-zero-sum games for synthesis
  Academic partners: LSV (coord), Aachen, Aalborg, ULB, U.MONS
  Industrial partners: Seluxit (DK), Energi Nord (DK)
- ANR ImpRo: robustness of timd systems
  Partners: IRCCyN (coord), IRISA, LIP6, LSV
- Digiteo project SIMS: controller synthesis for switched systems
  Partners: CEA-List (coord), LSV
- Farman project: BOOST2, ROSCOV, INVERSYM, DECORR

Research contracts



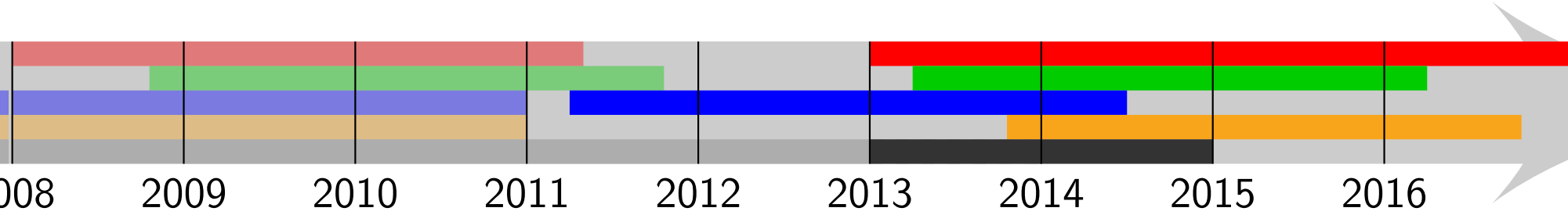
- ERC EQualIS: quantitative verification of complex systems
  Principal Investigator: Patricia Bouyer
- STREP Cassting: non-zero-sum games for synthesis
  Academic partners: LSV (coord), Aachen, Aalborg, ULB, U.MONS
  Industrial partners: Seluxit (DK), Energi Nord (DK)
- ANR ImpRo: robustness of timd systems
  Partners: IRCCyN (coord), IRISA, LIP6, LSV
- Digiteo project SIMS: controller synthesis for switched systems
  Partners: CEA-List (coord), LSV
- Farman project: BOOST2, ROSCOV, INVERSYM, DECORR

Research contracts



- ERC EQualIS: quantitative verification of complex systems
  Principal Investigator: Patricia Bouyer
- STREP Cassting: non-zero-sum games for synthesis
  Academic partners: LSV (coord), Aachen, Aalborg, ULB, U.MONS
  Industrial partners: Seluxit (DK), Energi Nord (DK)
- ANR ImpRo: robustness of timd systems
  Partners: IRCCyN (coord), IRISA, LIP6, LSV
- Digiteo project SIMS: controller synthesis for switched systems
  Partners: CEA-List (coord), LSV
- Farman project: BOOST2, ROSCOV, INVERSYM, DECORR

Research contracts



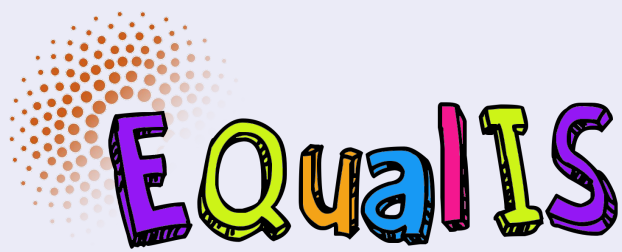
- ERC EQualIS: quantitative verification of complex systems
  Principal Investigator: Patricia Bouyer
- STREP Cassting: non-zero-sum games for synthesis
  Academic partners: LSV (coord), Aachen, Aalborg, ULB, U.MONS
  Industrial partners: Seluxit (DK), Energi Nord (DK)
- ANR ImpRo: robustness of timd systems
  Partners: IRCCyN (coord), IRISA, LIP6, LSV
- Digiteo project SIMS: controller synthesis for switched systems
  Partners: CEA-List (coord), LSV
- Farman project: BOOST2, ROSCOV, INVERSYM, DECORR

Outline

Research project

Quantitative verification and synthesis

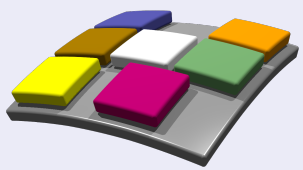
- **ERC project EQuaIIS:**
  - 01/2013-12/2017; 1.5 M€
  - led by Patricia Bouyer
  - Measures of correctness, quantitative model checking
  - Timed systems, robustness
  - Interacting systems, games, optimal strategies



Research project

Synthesis of Complex Systems

- **STREP project Cassting:**
  - 04/2013-03/2016; 2 M€
  - led by Nicolas Markey
  - LSV, Aalborg, Aachen, ULB, UMons + 2 industrial partners
  - Non-zero-sum games for synthesis
  - Imperfect information, networks of games
  - Equilibria, temporal logics for non-zero-sum games
  - Case studies: smart energy grids, smart houses

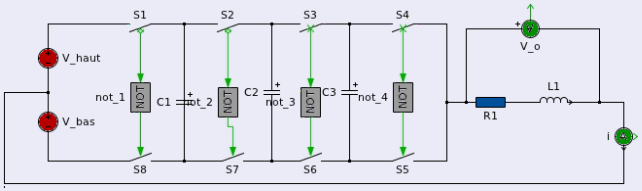


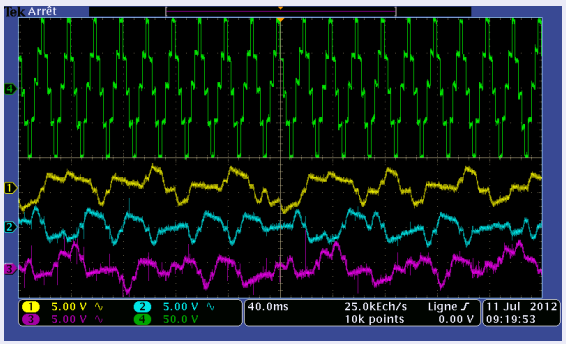
Cassting

Tempo after 2013

Controller Synthesis for Switched Systems

- **Farman project BOOST2:**
  - 2012-2013; led by Laurent Fribourg
  - controller synthesis for switched systems with ODE;
  - development of tool Minimator;
  - application to DC-DC converters in electronics.





- **Digiteo project SIMS:**
  - 2013-2016; Éric Goubault (CEA List, coord.), Sylvie Putot (CEA List), Laurent Fribourg (LSV); PhD grant for S. Mohamed.
  - controller synthesis for switched systems with PDE;
  - stability, safety, optimization;
  - applications: active control of vibrations, ...