Experiments in the use of τ-simulations for the components-verification of real-time systems

F. Bellegarde, J. Julliand, H. Mountassir and **E. Oudot**

LIFC, University of Franche-Comté, France

11th November 2006

- SAVCBS'06 -

5th International Workshop on Specification And Verification of Component-Based Systems

Portland, Oregon, USA

Context

- Real-time systems modeled in a compositional framework using timed automata [Alur and Dill 90]
- (Timed) Properties expressed in MITL (Metric Interval Temporal Logic) [Alur, Feder and Henzinger 96]

- A verification method: Model-Checking

Model-checking

- Algorithmic verification method

Model-checking

- Algorithmic verification method

Advantages

- Exhaustive
- Automatic

Model-checking

- Algorithmic verification method

Advantages

- Exhaustive
- Automatic

Drawbacks

- State-space explosion: difficulties to handle large-sized models
- Accentuated for Timed Systems

Model-checking

- Algorithmic verification method

Advantages

- Exhaustive
- Automatic

Drawbacks

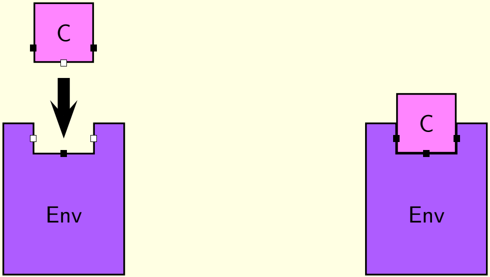
- State-space explosion: difficulties to handle large-sized models
- Accentuated for Timed Systems

$\longrightarrow$ A way out: using incremental development methods

Incremental Development for Component-Based Systems

Integration of Components / Local properties of the components

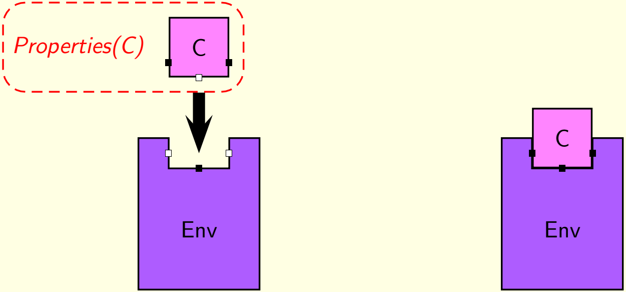
1. Consider a system with a component C and the rest of the components Env,



Incremental Development for Component-Based Systems

Integration of Components / Local properties of the components

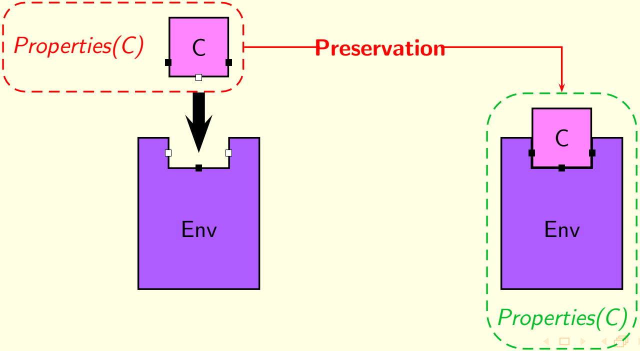
1. Consider a system with a component C and the rest of the components Env,
2. Check local properties of C on C,



Incremental Development for Component-Based Systems

Integration of Components / Local properties of the components

1. Consider a system with a component C and the rest of the components Env,
2. Check local properties of C on C,
3. Check that local properties of C are preserved when it is integrated in Env.



- How to ensure preservation ?

- Is incremental verification more efficient than classic verification ?

- How to ensure preservation ?
  - with timed τ-simulations

- Is incremental verification more efficient than classic verification ?

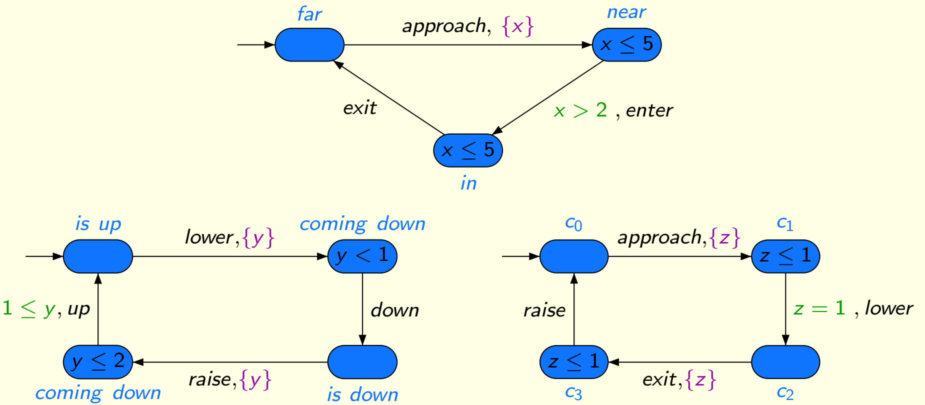
- How to ensure preservation ?
  - with timed τ-simulations
- Is incremental verification more efficient than classic verification ?
  - Need of experiments

1. Background on Timed systems
   - Modeling Timed systems with Timed Automata
   - Classic Composition Operator for Timed Automata
   - Specifying Timed Properties with MITL
2. Relations between components
   - Timed τ-Simulation
   - Divergence-sensitive and stability-respecting Timed τ-Simulation
3. Experiments
   - The Tool VeSTA
   - Production Cell
   - CSMA/CD Protocol

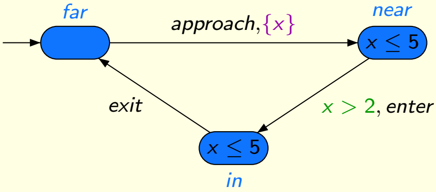
1. **Background on Timed systems**
   - **Modeling Timed systems with Timed Automata**
   - **Classic Composition Operator for Timed Automata**
   - **Specifying Timed Properties with Mitl**
2. Relations between components
   - Timed τ-Simulation
   - Divergence-sensitive and stability-respecting Timed τ-Simulation
3. Experiments
   - The Tool VeSTA
   - Production Cell
   - CSMA/CD Protocol

Timed Automata

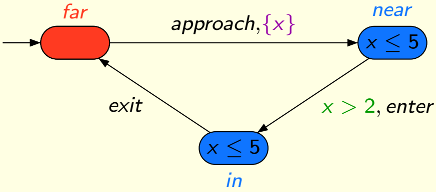
- Finite automata with real-valued variables called clocks.
- An example: the Railroad crossing:

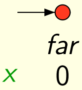


Runs of a Timed Automaton

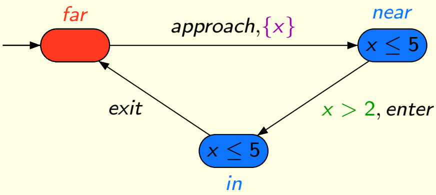


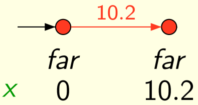
Runs of a Timed Automaton



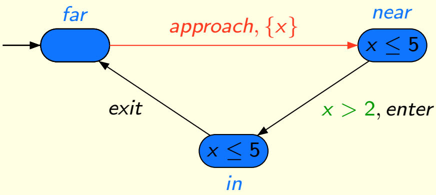


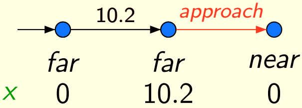
Runs of a Timed Automaton



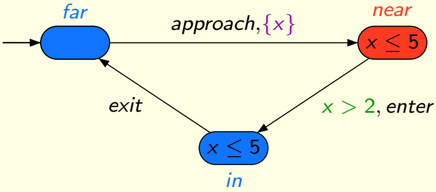


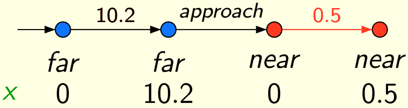
Runs of a Timed Automaton



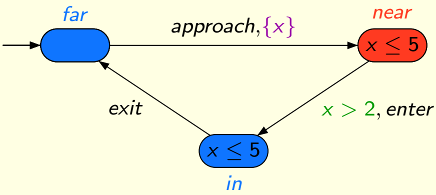


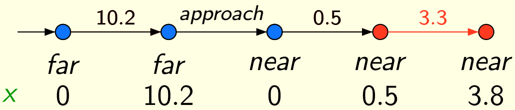
Runs of a Timed Automaton



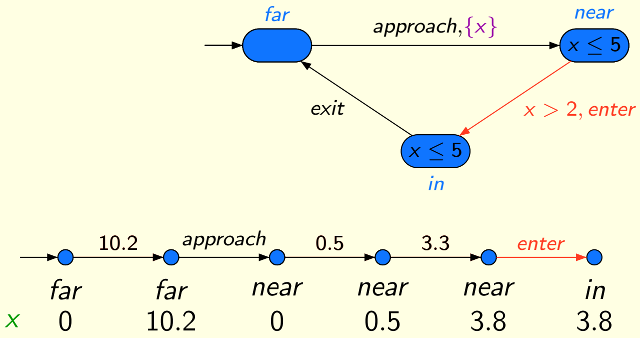


Runs of a Timed Automaton

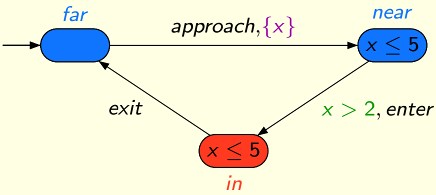


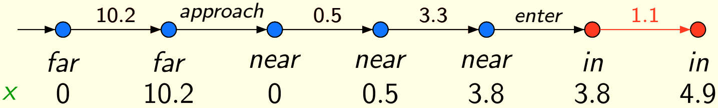


Runs of a Timed Automaton

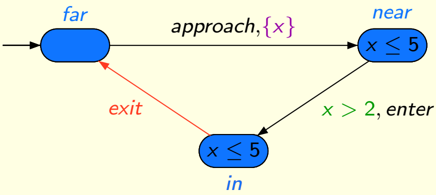


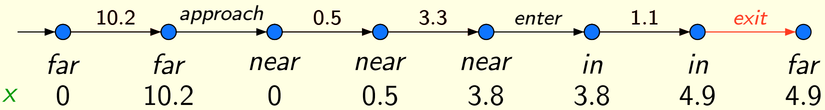
Runs of a Timed Automaton



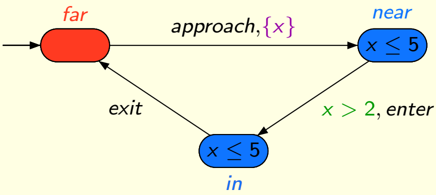


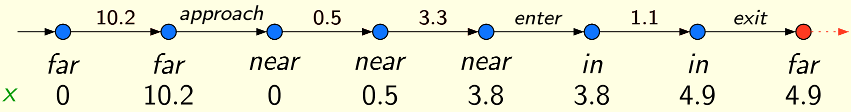
Runs of a Timed Automaton



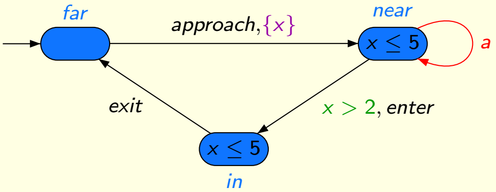


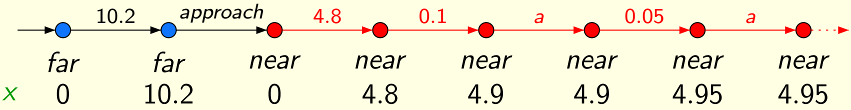
Runs of a Timed Automaton



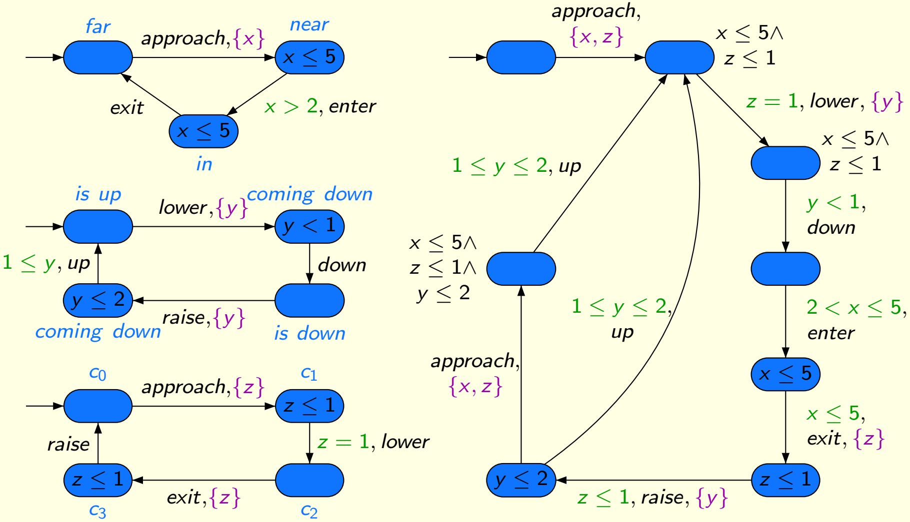


Zeno Runs should be ignored since they are not realistic



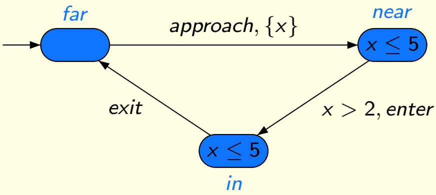


- Synchronization: actions with same label
- Other actions interleave



The Logic MITL

- Metric Interval Temporal Logic,
- Temporal operators (possibly) constrained by a time delay

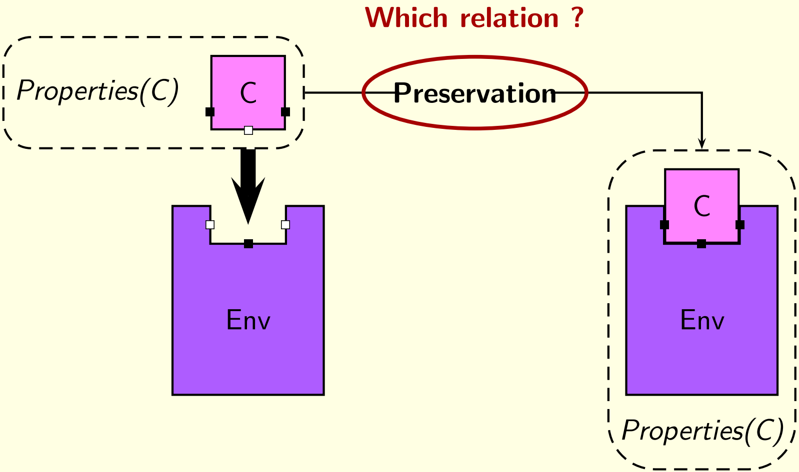


Examples: local properties of the train

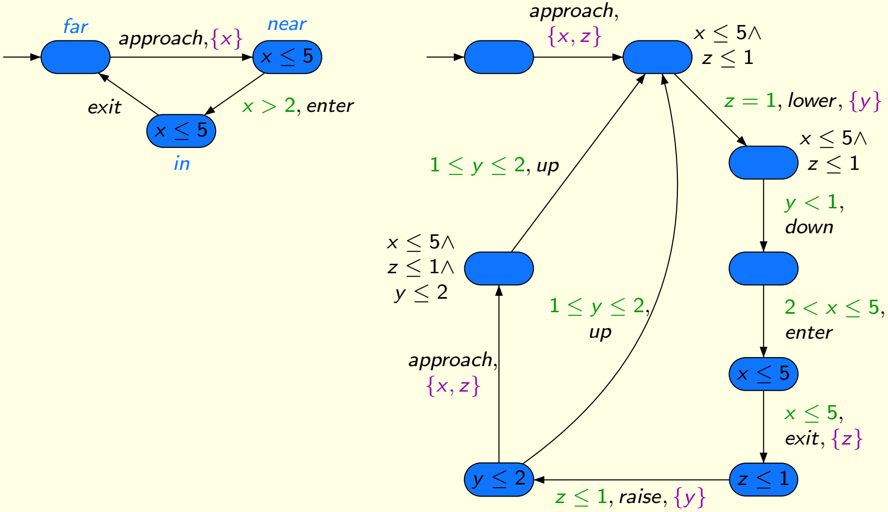
- The train is not on the railroad crossing within the two t.u. following the emission of the signal "approach": $\Box(near \Rightarrow \Box_{<2}\neg in) \rightarrow$ Safety,
- When the train approaches, it will eventually exit the railroad crossing: $\Box(near \Rightarrow \Diamond far) \rightarrow$ Liveness

1. Background on Timed systems
    - Modeling Timed systems with Timed Automata
    - Classic Composition Operator for Timed Automata
    - Specifying Timed Properties with MITL
2. **Relations between components**
    - **Timed τ-Simulation**
    - **Divergence-sensitive and stability-respecting Timed τ-Simulation**
3. Experiments
    - The Tool VeSTA
    - Production Cell
    - CSMA/CD Protocol

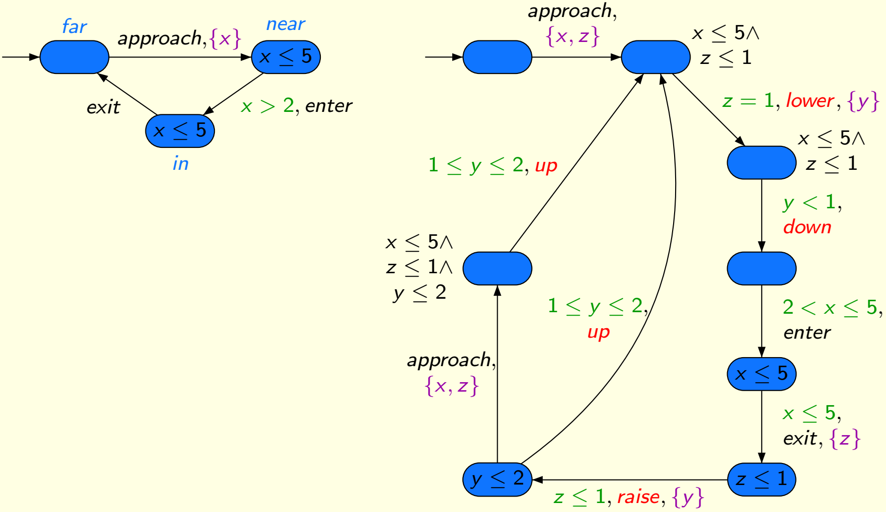
Which relation between $C||E$ and C preserving the local properties of C on $C||E$?



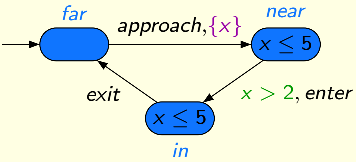
- The relation between C and $C||E$ is a timed τ-simulation, written $C||E \preceq C$, i.e., a simulation modulo the actions of E (τ-actions)

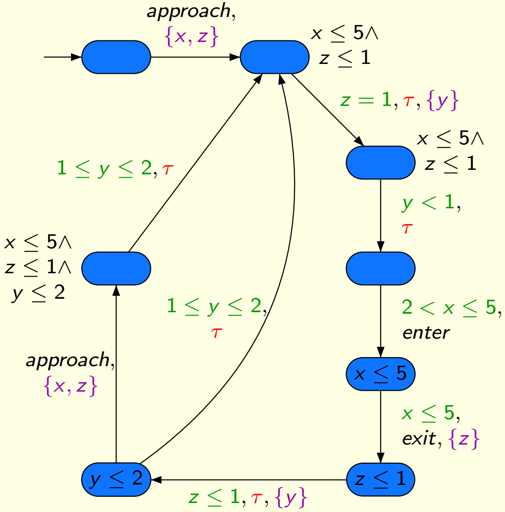


- The relation between C and $C||E$ is a timed τ-simulation, written $C||E \preceq C$, i.e., a simulation modulo the actions of E (τ-actions)



- The relation between C and $C||E$ is a timed τ-simulation, written $C||E \preceq C$, i.e., a simulation modulo the actions of E (τ-actions)





Theorem (Preservation of safety properties)

Let φ be a safety property. C and E are timed automata.

$$\textit{If } C \models \varphi \textit{ and } C||E \preceq C \textit{ then } C||E \models \varphi$$

Nice properties w.r.t. $||$

- *Composability*
  $C||E \preceq C$
- *Compatibility*
  $C \preceq C' \Rightarrow C||E \preceq C'||E$
- *Compositionality*
  $C \preceq C'$ *and* $D \preceq D' \Rightarrow C||D \preceq C'||D'$

Theorem (Preservation of safety properties)

Let φ be a safety property. C and E are timed automata.

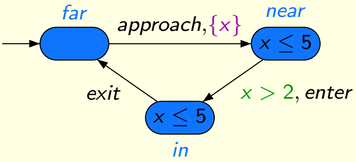
$$\text{If } C \models \varphi \text{ and } C||E \preceq C \text{ then } C||E \models \varphi$$

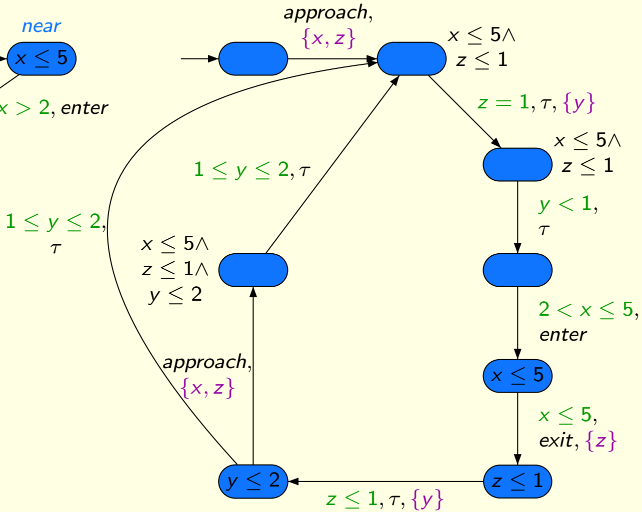
Nice properties w.r.t. $||$

- *Composability*
  $C||E \preceq C$
- *Compatibility*
  $C \preceq C' \Rightarrow C||E \preceq C'||E$
- *Compositionality*
  $C \preceq C' \text{ and } D \preceq D' \Rightarrow C||D \preceq C'||D'$

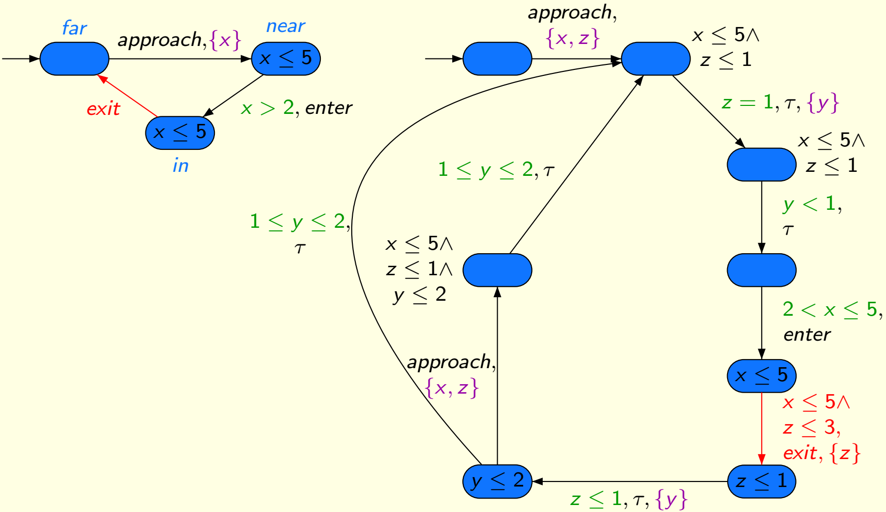
$\rightarrow$ Contribution of timed τ-simulation for incremental verification w.r.t. classic verification is immediate for safety properties.

- To preserve liveness properties, two more requirements
  1. No new deadlocks (stability-respecting)

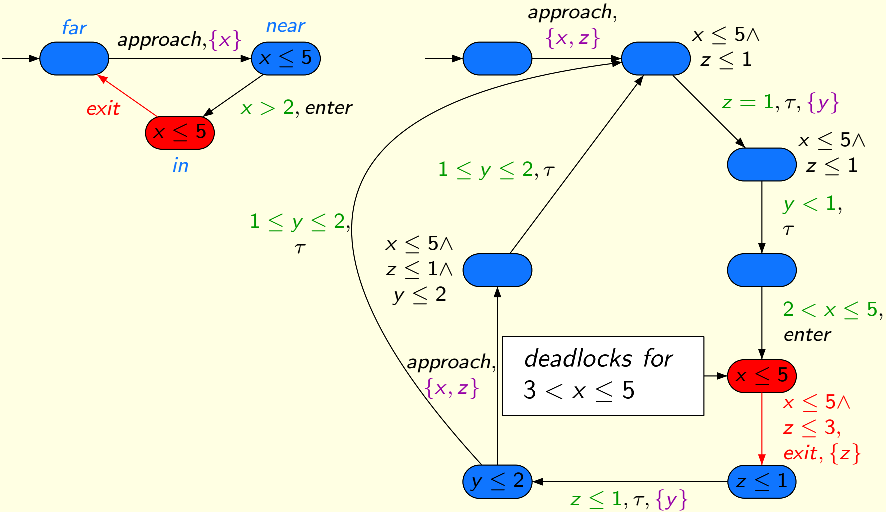




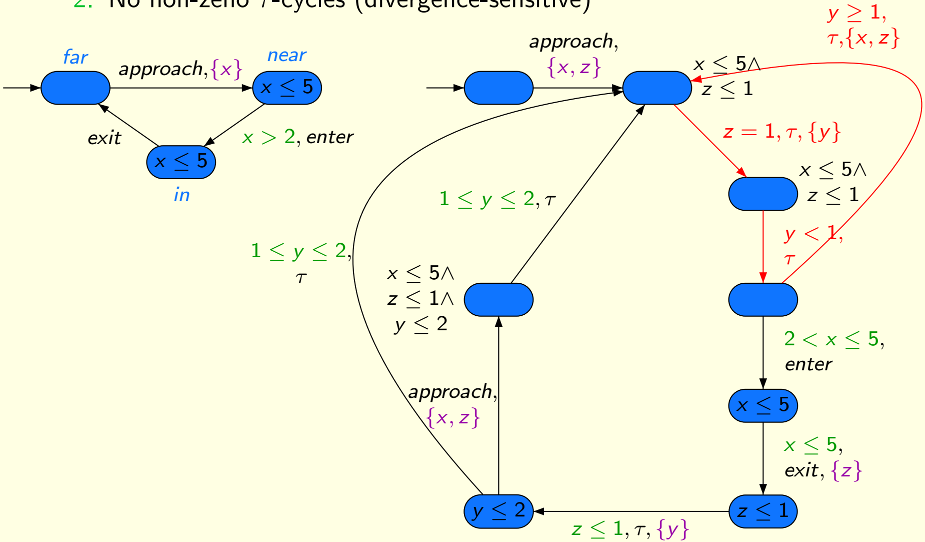
- To preserve liveness properties, two more requirements
  1. No new deadlocks (stability-respecting)



- To preserve liveness properties, two more requirements
  1. No new deadlocks (stability-respecting)



- To preserve liveness properties, two more requirements
  1. No new deadlocks (stability-respecting)
  2. No non-zeno τ-cycles (divergence-sensitive)



- The timed τ-simulation with these two requirements is called a Divergence-sensitive and Stability-respecting (DS) timed τ-simulation, written $C||E \preceq_{ds} C$.

Theorem (Preservation of MITL properties)

Let φ be a MITL property. C and E are timed automata.

$$\text{If } C \models \varphi \text{ and } C||E \preceq_{ds} C \text{ then } C||E \models \varphi.$$

- Composability and compatibility w.r.t. $||$ are not ensured for free.

$\longrightarrow$ Check algorithmically the DS timed τ-simulation
$\longrightarrow$ Compare with classic verification

1. Background on Timed systems
   - Modeling Timed systems with Timed Automata
   - Classic Composition Operator for Timed Automata
   - Specifying Timed Properties with MITL
2. Relations between components
   - Timed τ-Simulation
   - Divergence-sensitive and stability-respecting Timed τ-Simulation
3. **Experiments**
   - **The Tool Vesta**
   - **Production Cell**
   - **CSMA/CD Protocol**

- Verification of Simulations for Timed Automata,
- Checks the DS timed τ-simulation in the framework of integration of components, i.e., it checks

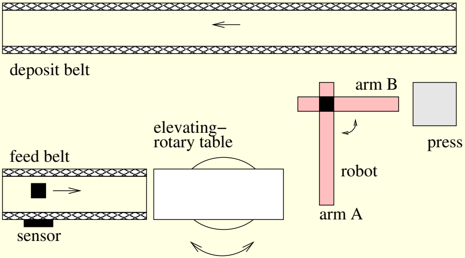
$$C||E \preceq_{ds} C$$

- VeSTA is available at

http://lifc.univ-fcomte.fr/ ∼ oudot/VeSTA

Presentation of the Production Cell Case Study

- Modeling: at least seven components (six devices + one or several pieces)



- Local properties to check
  - 7 local properties for the robot (P_1 to P_7), in particular 2 liveness and 3 bounded liveness.
  - 1 liveness local property for robot $||$ press (P_8).

Classic Method

Model-check all properties on the complete model

$$feed\ \ belt||sensor||table||robot||press||deposit\ \ belt||piece\ 1$$

Our Method

1. Model-check properties P_1 to P_7 on the robot,
2. Model-check property P_8 on robot || press,
3. Check preservation of P_1 to P_7 on robot || press, i.e.,
   $$robot||press \preceq_{ds} robot,$$
4. Check preservation of P_8 on the whole model, i.e.,
   $$complete\ \ model \preceq_{ds} robot||press.$$

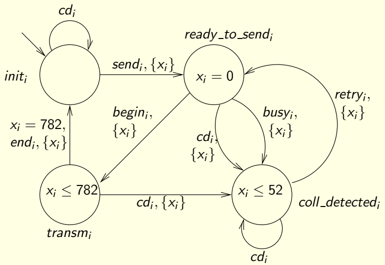
Detailed Results

- Computation times (seconds)

Property	Type	Classic Method	Local Verification	Preservation Checking
P_1	Safety	0.01	< 0.001	
P_2	Safety	0.01	< 0.001	
P_3	Liveness	0.98	< 0.001	
P_4	Liveness	15.79	0.04	0.05
P_5	Bounded Response	0.68	< 0.001	
P_6	Bounded Response	0.48	< 0.001	
P_7	Bounded Response	0.7	< 0.001	
P_8	Liveness	0.93	0.02	0.46
Total		19.58	0.06	0.51

Presentation of the CSMA/CD Protocol Case study

- Carrier Sense, Multiple Access with Collision Detection protocol
- Modeling: at least three components (a medium + 2 or more senders)
- Parameterized system (parameter: number of senders)



- The main property (P): *whatever the number of stations, if a collision occurs between two stations i and j, $i \neq j$, both detect it within 26 t.u.*

Classic Method

Model-check P on the complete model, with 2 senders, 3 senders, 4 senders...

Our method

- Model-check P on a model with 2 senders.
- Check that preservation is ensured when adding other senders.

Classic Method

Model-check P on the complete model, with 2 senders, 3 senders, 4 senders...

→ $\leq$ 6 senders: model-checking successful (from $<$0.001 seconds to $>$57 minutes),

→ $\geq$ 7 senders: verification can not be run to completion (waiting for ten hours).

Our method

- Model-check P on a model with 2 senders.
- Check that preservation is ensured when adding other senders.

Classic Method

Model-check P on the complete model, with 2 senders, 3 senders, 4 senders...

→ $\leq$ 6 senders: model-checking successful (from <0.001 seconds to >57 minutes),

→ $\geq$ 7 senders: verification can not be run to completion (waiting for ten hours).

Our method

- Model-check P on a model with 2 senders.
- Check that preservation is ensured when adding other senders.

→ Verification with 2 senders: <0.001 seconds

→ Preservation ensured thanks to simple arguments guaranteeing that DS timed τ-simulation holds.

- Preservation of safety / liveness MITL properties during integration of components with (DS) timed τ-simulation relations.

- Comparison with classic verification:
  - for safety properties, preservation is ensured for free
  - for liveness properties, first experiments results (verification time) seem encouraging.

- Study the contribution of timed τ-simulations for parametrized systems, e.g., networks of automata (as CSMA/CD protocol),
- Study other composition operators, which would guarantee deadlock and τ-livelock-freedom during integration of components (J. Sifakis)
- How to guide a decomposition into components to obtain their compatibility with the (DS) timed τ-simulation,
- How to reuse a component so that its integration in an application is compatible with the DS timed τ-simulation.

- Study the contribution of timed τ-simulations for parametrized systems, e.g., networks of automata (as CSMA/CD protocol),
- Study other composition operators, which would guarantee deadlock and τ-livelock-freedom during integration of components (J. Sifakis)
- How to guide a decomposition into components to obtain their compatibility with the (DS) timed τ-simulation,
- How to reuse a component so that its integration in an application is compatible with the DS timed τ-simulation.

- Questions ?

Experiments in the use of τ-simulations for the components-verification of real-time systems

F. Bellegarde, J. Julliand, H. Mountassir and **E. Oudot**

LIFC, University of Franche-Comté, France

11th November 2006

- SAVCBS'06 -
5^{th} International Workshop on Specification And Verification of Component-Based Systems
Portland, Oregon, USA