# Linear Hybrid Automata for Analyzing Hybrid Systems

Akshay Rajhans

ECE Qualifying Exam Presentation, Spring 2010

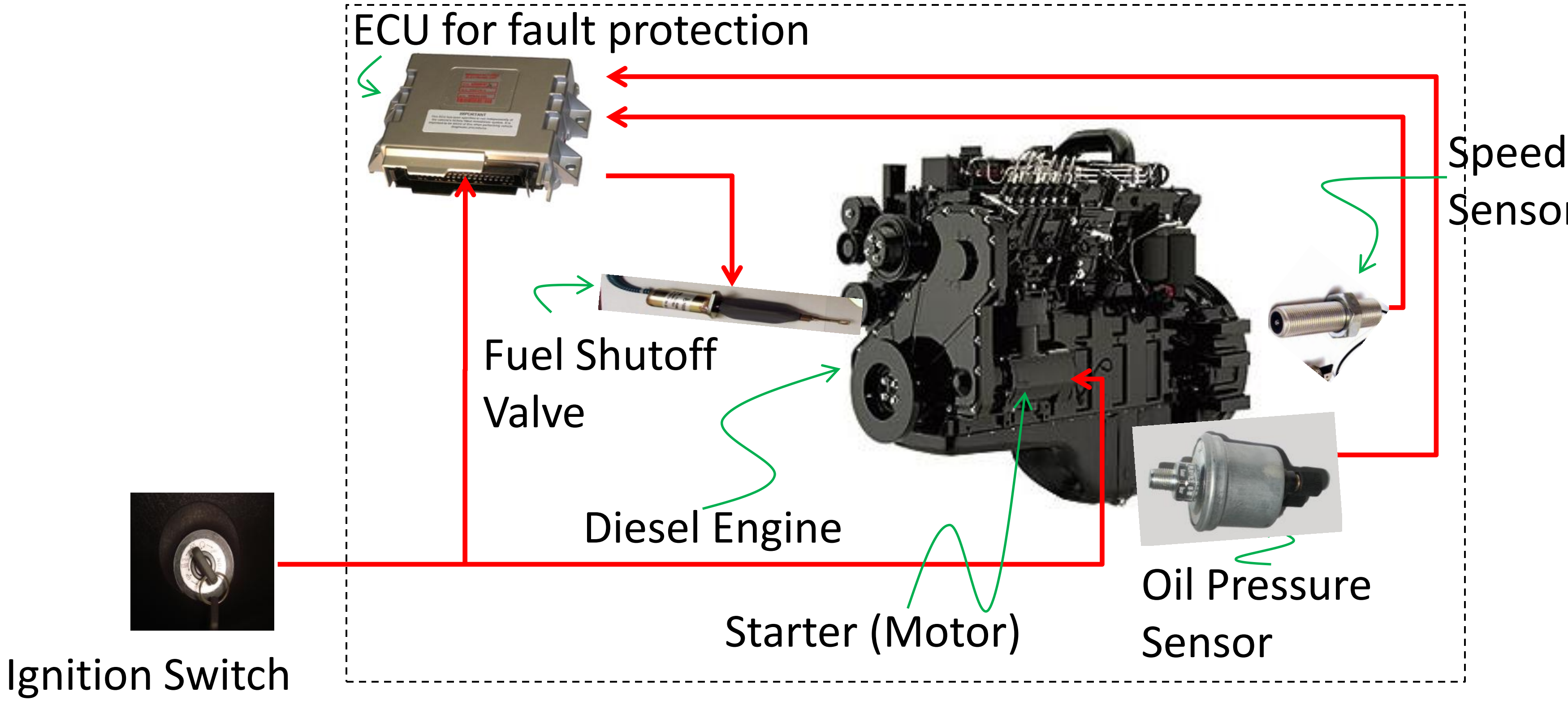
Apr 20, 2010

# Outline

- **Hybrid systems**
  - Hybrid automata, linear hybrid automata
- **Checking Conformance to Specification**
  - Simulation relations, computing them
- **Checking Conformance Compositionally**
  - Assume-guarantee reasoning
- **Future Research Directions**
- **Summary**

# A motivating example

- Verifying specifications “under all cases”



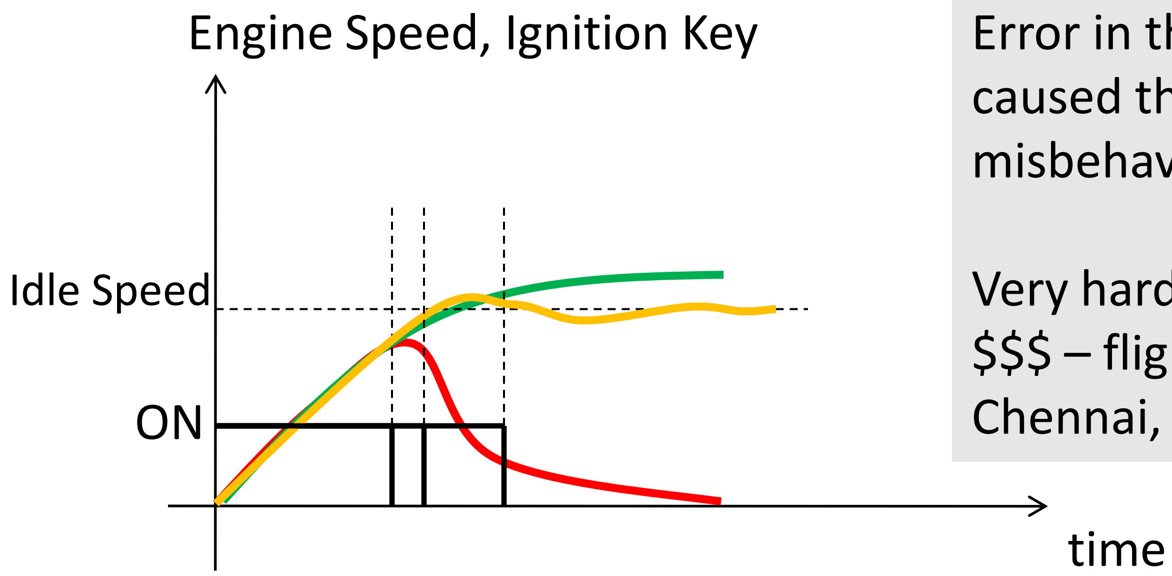

**Specification**: “If engine is running and oil pressure drops below x psi, the engine is shut off.”

**What we observed: Not always!**

# Diagnosis

Starting logic in ECU:

"If ignition key pressed & engine speed > idle, engine started"

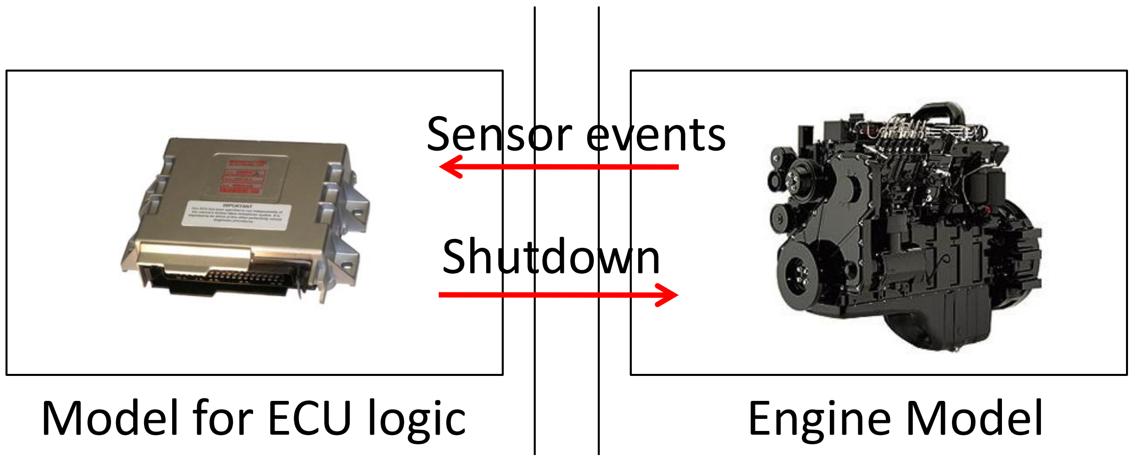



Error in the starting logic caused the stopping logic to misbehave.

Very hard to debug. > 2 months.
$$$ – flights between Pune, Chennai, Bangalore; man-hours

# If we had…

A way to model our (software & physical) system…



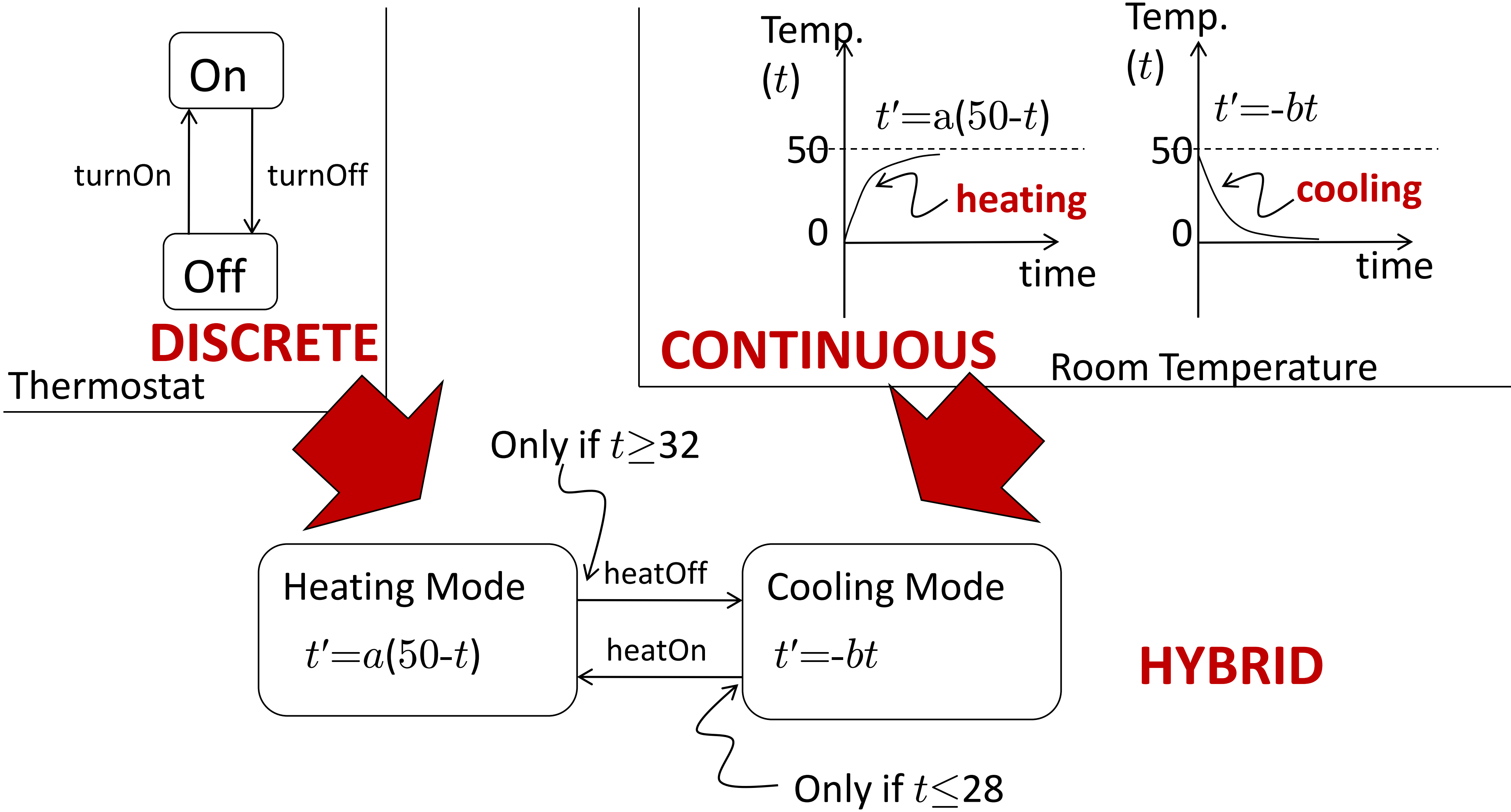

and…

A method and/or a tool to verify the specification never gets violated under ANY conditions …

# Outline

- **Hybrid systems**
  - Hybrid automata, linear hybrid automata
- **Checking Conformance to Specification**
  - Simulation relations, computing them
- **Checking Conformance Compositionally**
  - Assume-guarantee reasoning
- **Future Research Directions**
- **Summary**

# Hybrid dynamics: a simple example

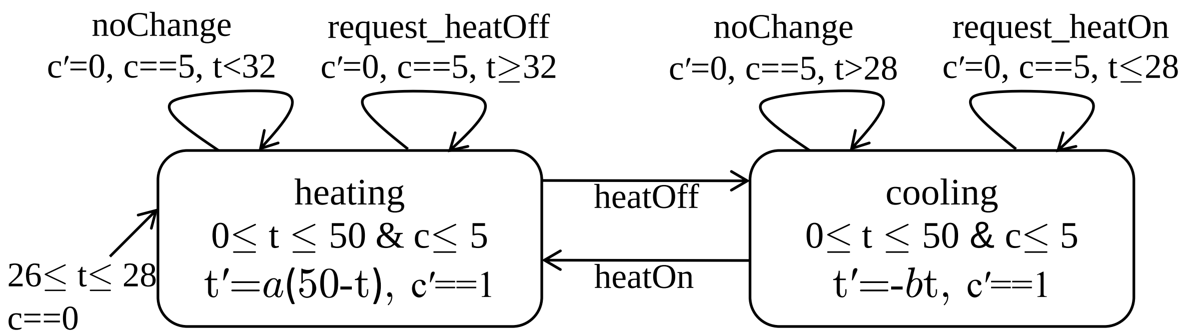



“Temperature Control System”

# Hybrid automata (HA)

**HA:** tuple $(Loc, Var, Lab, Tran, Act, Inv, Init)$

**Example:**




**Thermostat System**

-Sample temperature every 5 sec.

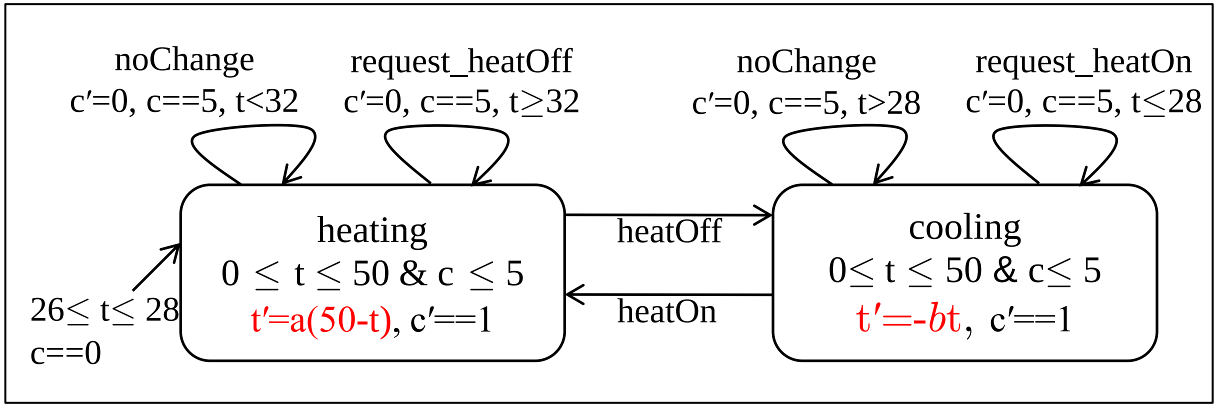
-Set-point 30 deg, hysteresis $\pm$ 2 deg

# Linear Hybrid automata (LHA)

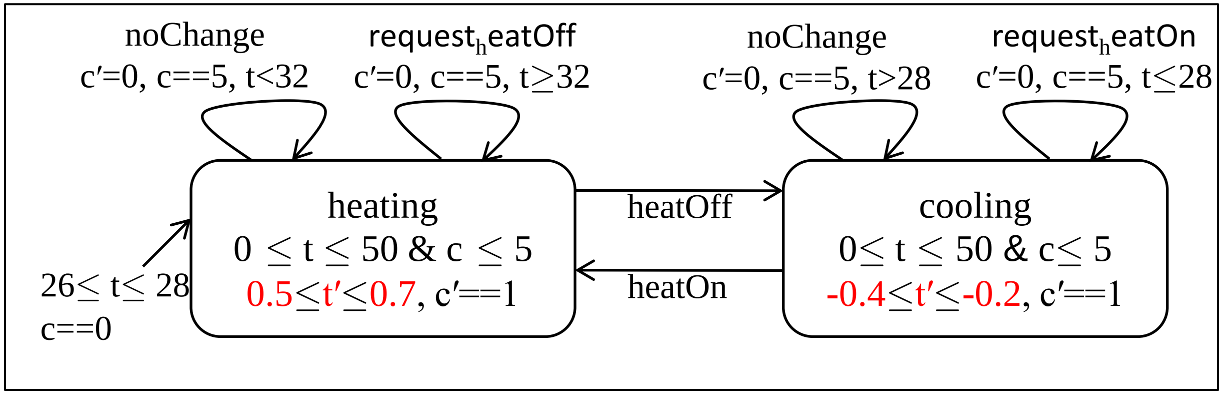
**LHA:** $Act, Inv, Init$ and continuous part of $Tran$ given by linear formulas $Ax$ {≤or<} $b$

**Example:**

**HA**

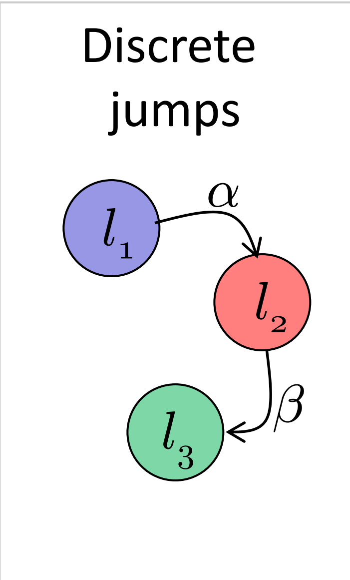


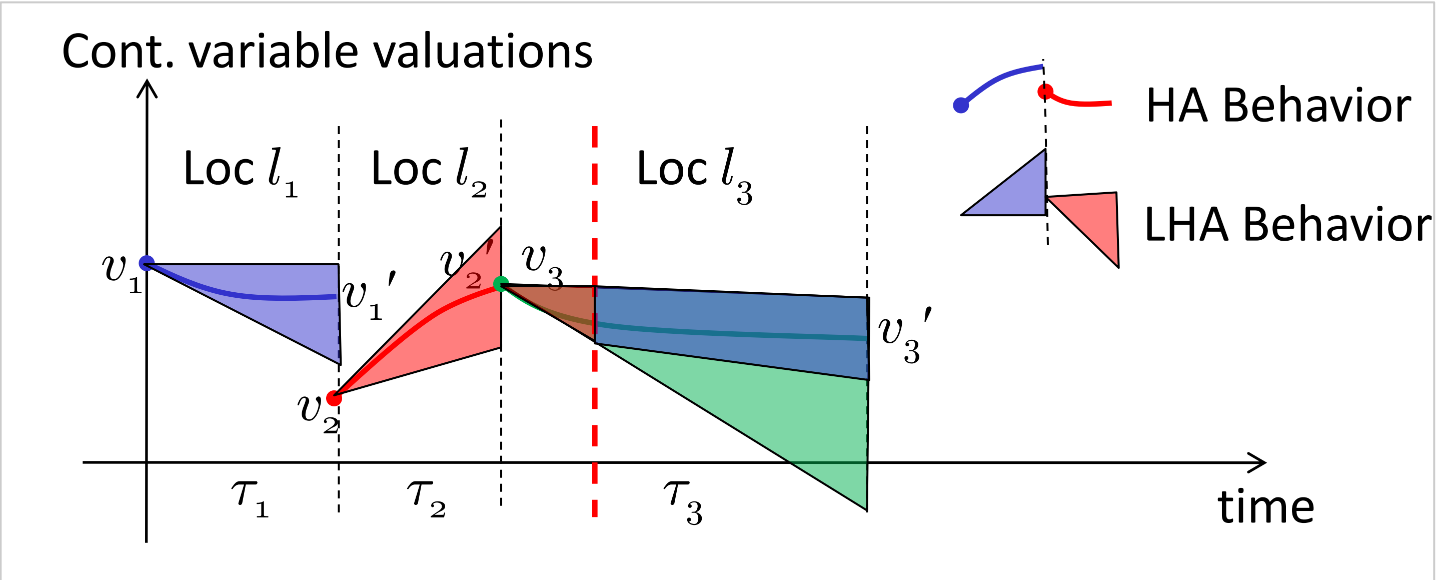

**LHA**



# Behaviors of hybrid systems

- Continuous evolutions and discrete jumps

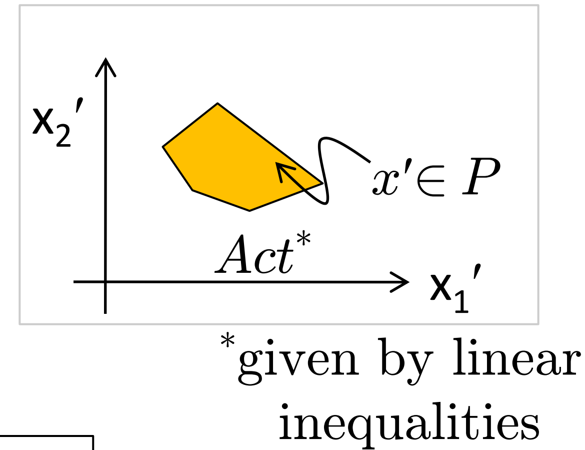


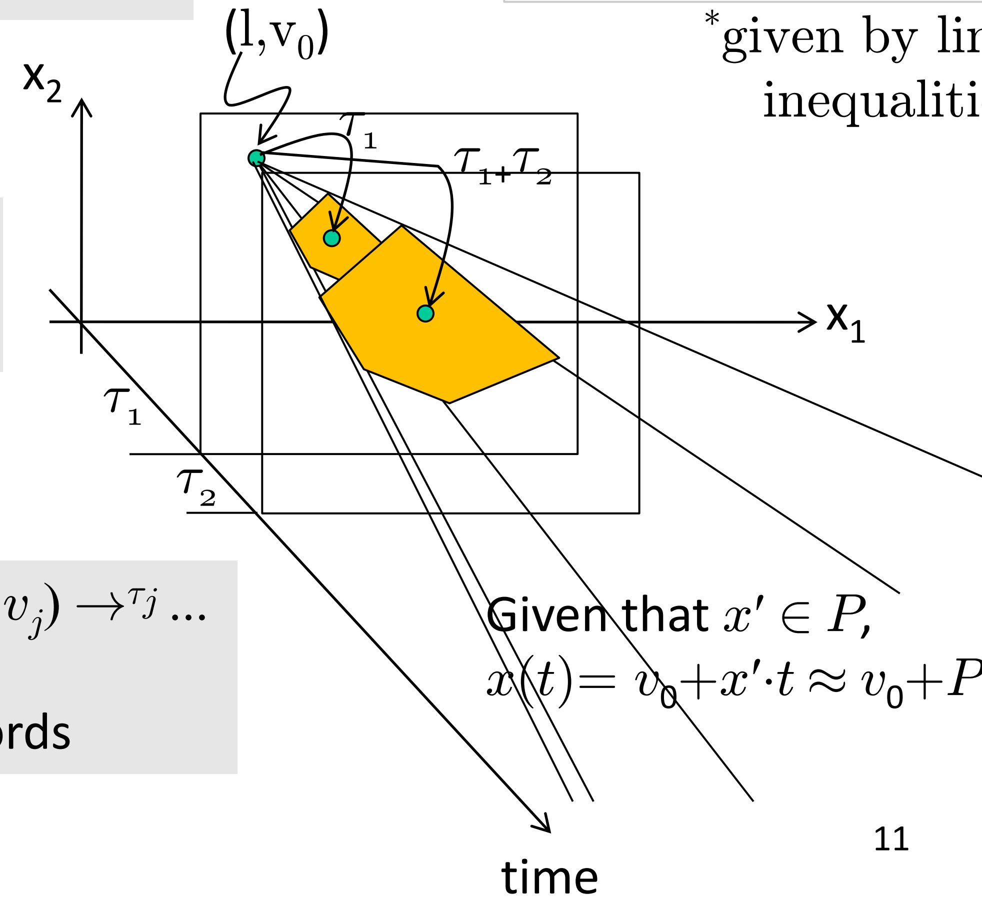

- LHA can approximate complex hybrid dynamics arbitrarily well. **[Hen+98]**

# Timed transitions of LHA

LHA states are pairs $(l,v)$, where
- $l \in Loc$ and
- $v$ = instantaneous valuation of $Var$




Time evolution in a location is a polyhedral computation




***Timed* trace**: $(l_i,v_i) \rightarrow^{\tau_i} (l_i,v_i') \rightarrow^{\alpha} (l_j,v_j) \rightarrow^{\tau_j} \ldots$
***Timed* word**: $\tau_i \; \alpha \; \tau_j \ldots$
***Timed* language**: set of all timed words

# Outline

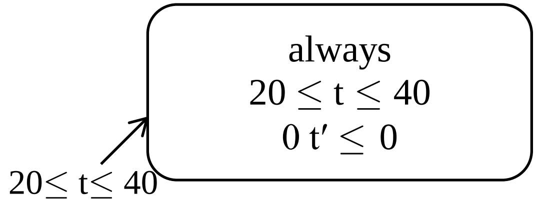
- ✓ **Hybrid systems**
  - Hybrid automata, linear hybrid automata
- ➢ **Checking Conformance to Specification**
  - Simulation relations, computing them
- **Checking Conformance Compositionally**
  - Assume-guarantee reasoning
- **Future Research Directions**
- **Summary**

# Checking Conformance using LHA

## In words

“Under any condition, the room temperature never goes below 20 C or above 40 C.”

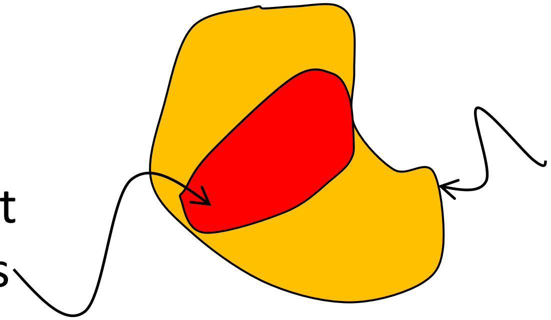
## Modeled as an LHA




**Specifications**

## Verification problem (in abstract sense)

Set of thermostat system behaviors

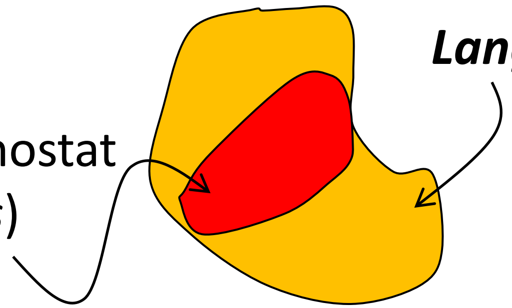


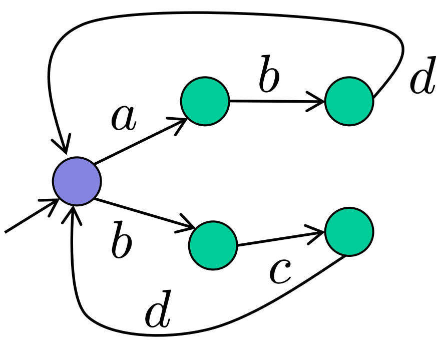
Set of behaviors of (allowed by) the specification

# Checking Conformance – Transition systems

## Verification problem

*Language* = Set of thermostat system behaviors (*words*)




In case of loops:
- words infinite, e.g., `abdbcd`...
- language with infinite words

## Language inclusion

**Brute force method impossible** for infinite words and/or languages

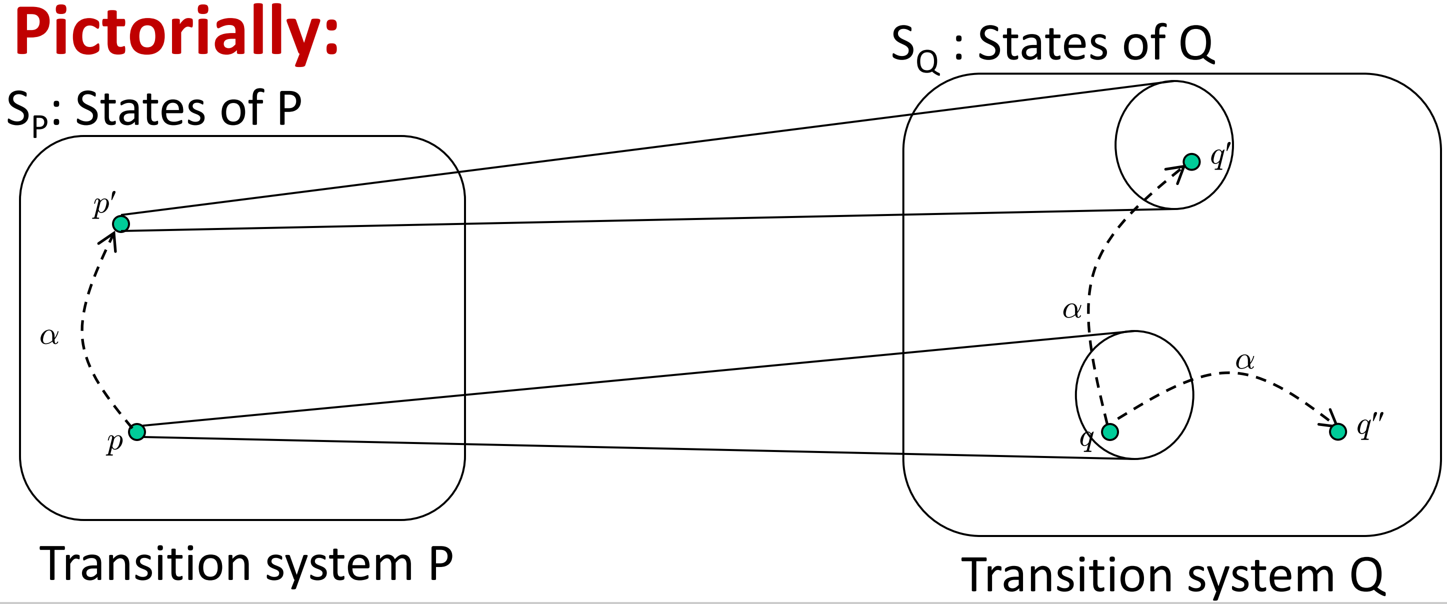
# Simulation Relations – Transition systems

- **Local** condition to guarantee language inclusion
- A relation $\preceq \subseteq S_P \times S_Q$ is a **simulation relation** iff

  $\forall$ (p,q)$\in \preceq$, also written as p$\preceq$q and $\forall\alpha\in$set of labels $\Sigma$,

  if $p \rightarrow^{\alpha} p'$ then $\exists\, q' \in S_Q$ s.t. $q \rightarrow^{\alpha} q' \wedge p' \preceq q'$.

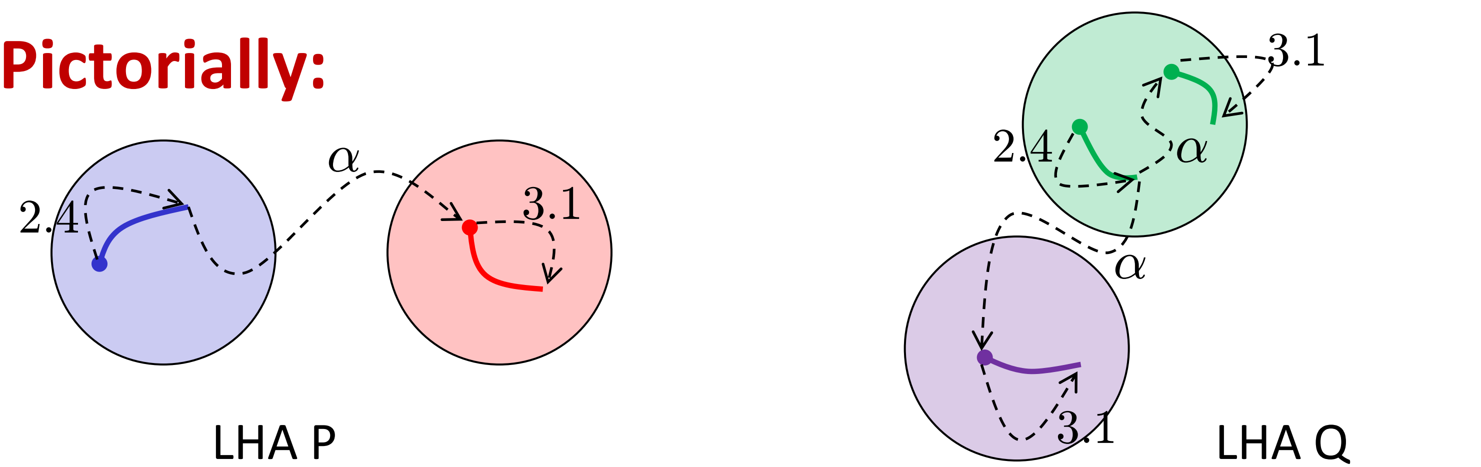
- **Pictorially:**



# Simulation Relations – LHA

- A relation $\preceq \subseteq S_P \times S_Q$ is a **simulation relation** iff
  $\forall\ (k,u) \preceq (l,v)$ and $\forall \alpha \in$ set of labels $Lab \cup R^+$,
  if $(k,u) \rightarrow^{\alpha} (k',u')$ then $\exists\ (l',v') \in S_Q$ s.t. $(l,v) \rightarrow^{\alpha} (l',v') \wedge (k',u') \preceq (l',v')$.
- **States are pairs, transitions are timed or discrete**

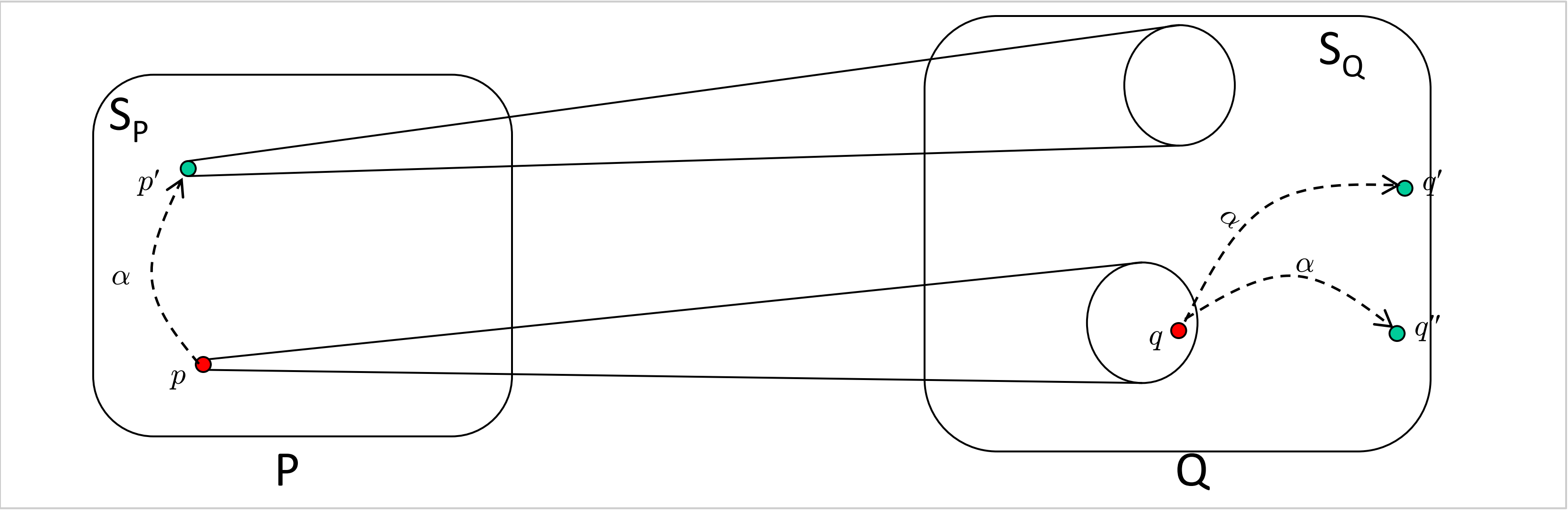
- **Pictorially:**



# Computing simulation relations

## Fixed-point algorithm: (Idea)

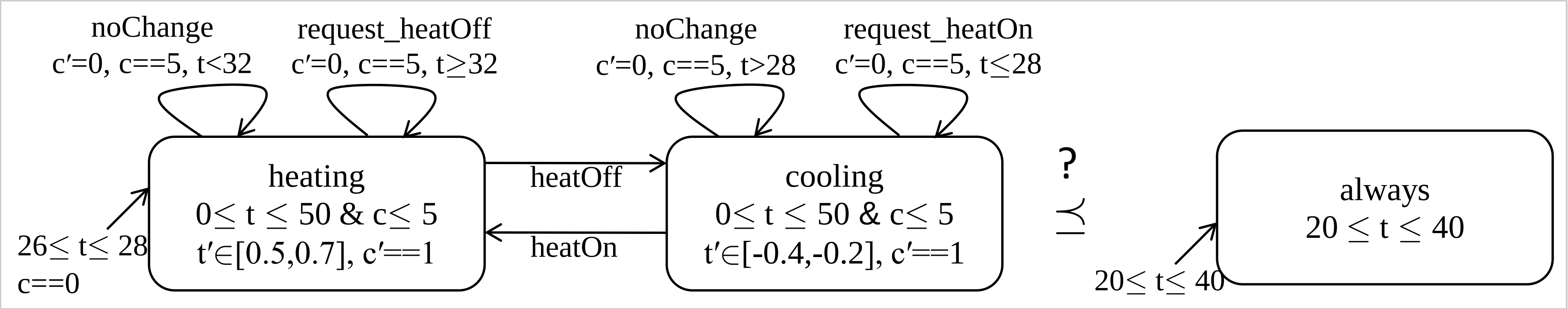
1. Starting with the first guess $\preceq_0 = S_P \times S_Q$
2. Refine the guess $\preceq_{i+1} := \preceq_i \backslash B_i$ by subtracting bad states
3. Stop when no more bad states are left (i.e. when a ***fixed-point*** is reached) **i.e.** $\preceq_{i+1} == \preceq_i$.



# PHAVer

- **P**olyhedral **H**ybrid **A**utomaton **Ver**ifier [Fre08]
  - Can compute simulation relations for LHA using the fixed-point algorithm
- For thermostat example, we ask:

```
is_sim(thermostat,spec)?
```




PHAVer: `No.`

# Outline

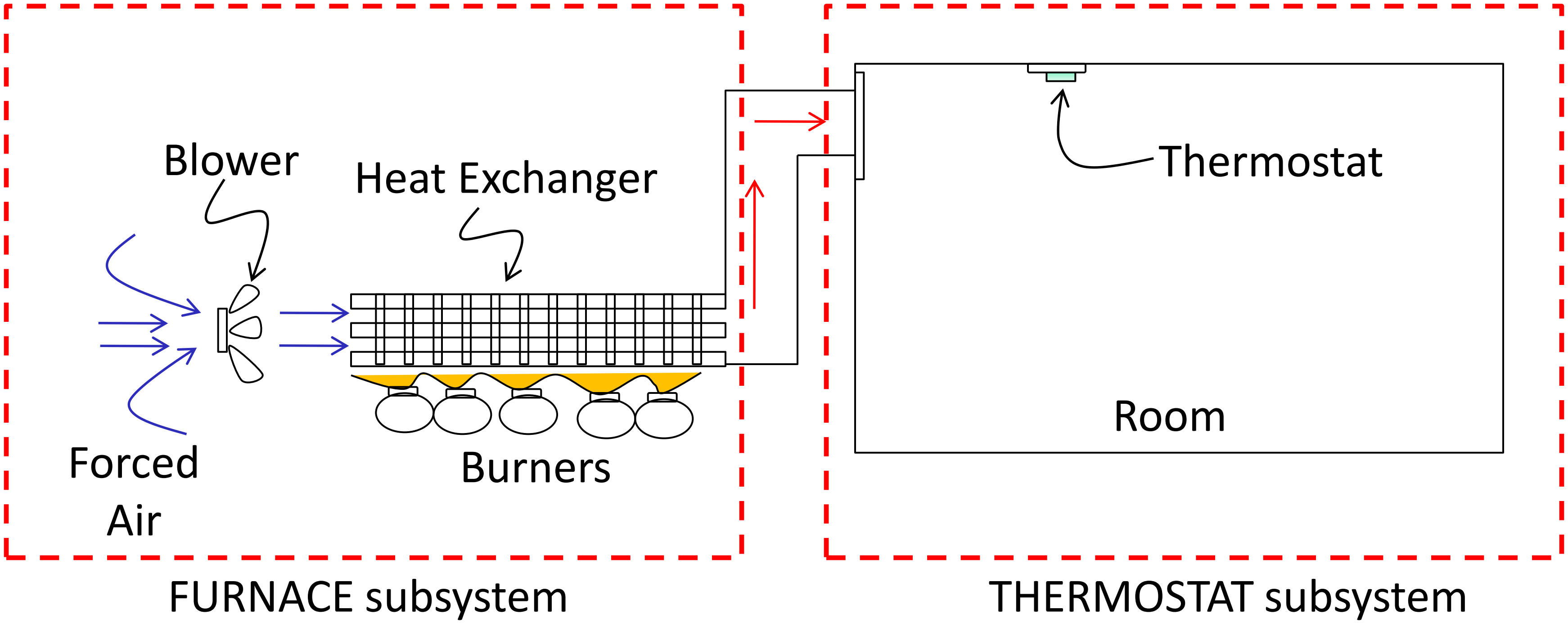
- ✓ **Hybrid systems**
  - Hybrid automata, linear hybrid automata
- ✓ **Checking Conformance to Specification**
  - Simulation relations, computing them
- ➢ **Checking Conformance Compositionally**
  - Assume-guarantee reasoning
- **Future Research Directions**
- **Summary**

# Modular modeling and reasoning

- **Parallel composition** ('||') **operation:**
  - Events with matching labels are synchronous.
  - Continuous variables are disjoint.
  - $\mathrm{Sys} = \mathrm{Subsys}_1 || \mathrm{Subsys}_2 || \ldots$
- **Often:** Modeling and analysis of such a $\mathrm{Sys}$ too expensive.
- **Need:** Ability to deduce whether $\mathrm{Sys} \preceq Q$ without having to construct $\mathrm{Sys}$ explicitly.

# Example

- sys = thermostat || furnace




- Burners heats unevenly, $n$ monitoring sensors

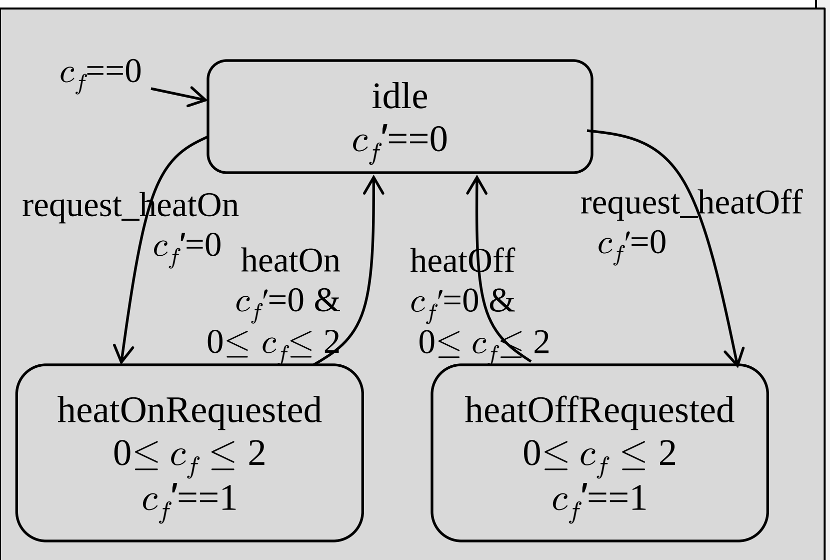
# Assume-guarantee (AG) reasoning

- **Non-circular reasoning** [Fre04]

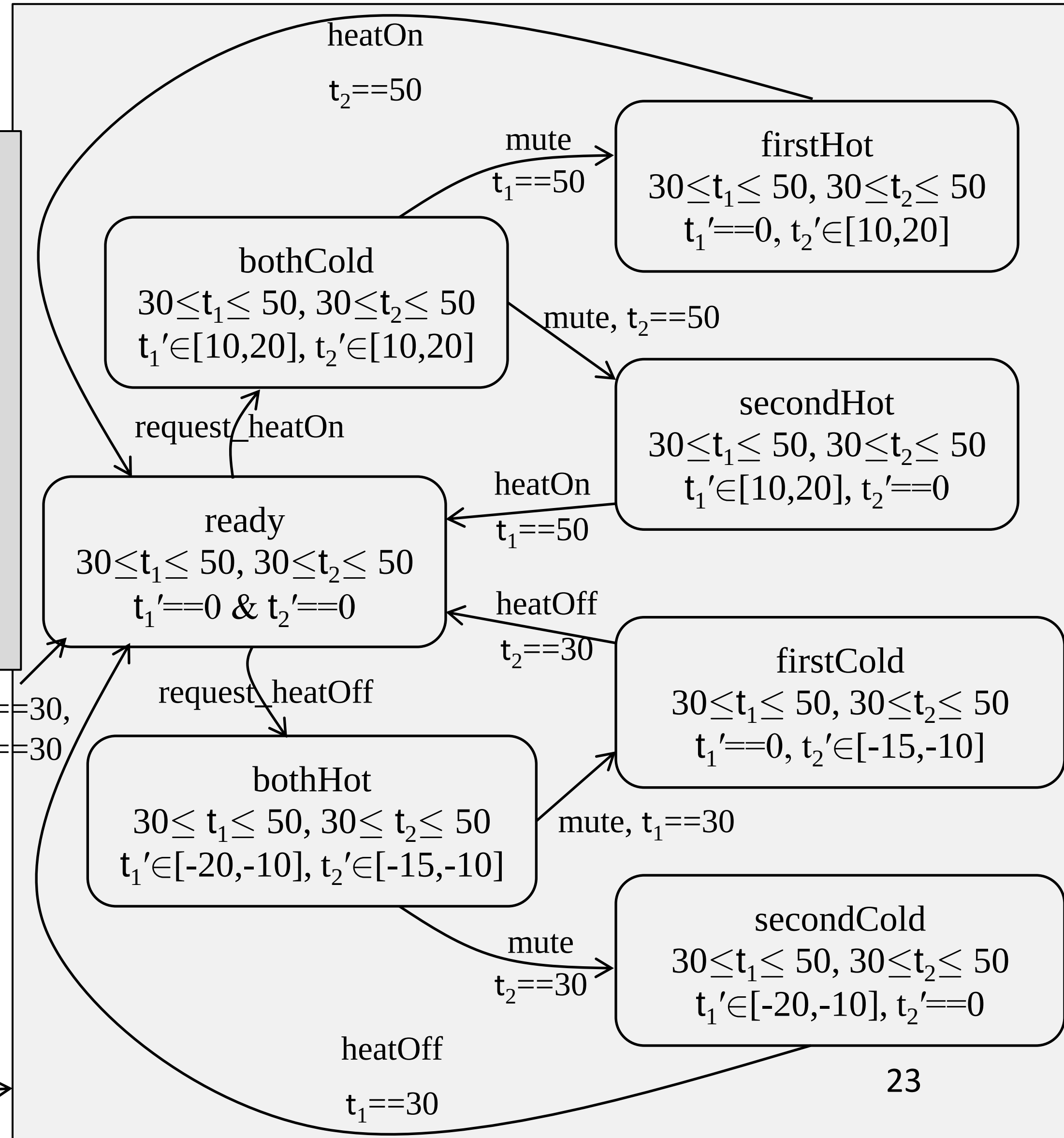
$$\frac{\begin{array}{rcl} P_1 \| A & \preceq & Q \\ P_2 & \preceq & A \end{array}}{\begin{array}{rcl} P_1 \| P_2 & \preceq & Q \end{array}}$$

- **Main Idea:** Complex task deduced from simpler subtasks
- **Profitable** if $A$ is simpler than $P_2$, i.e. $P_1 || A$ simpler than $P_1 || P_2$

# Example (Furnace vs Assumption)




$$\begin{array}{ccc} P_1 \,||\, \boxed{A} & \preceq & Q \\ P_2 & \preceq & A \\ \hline P_1 \,||\, \boxed{P_2} & \preceq & Q \end{array}$$



# Experimental results

## Comparison: Computation time AG/non-AG

| #sensors in the furnace model | **Non-AG Method** $P_1 \| P_2 \preceq Q$ | | **AG Method** $P_1 \| A \preceq \textit{ and } Q \| P_2 \preceq A$ | |
|---|---|---|---|---|
| | # State variables | Time (s) | # State variables | Time (s) |
| 1 | 4 | 2.10 | 4 *and* 2 | 1.10 |
| 2 | 5 | 121.45 | 4 *and* 3 | 2.85 |
| 3 | 6 | $\infty$* | 4 *and* 4 | 23.51 |
| 4 | 7 | $\infty$* | 4 *and* 5 | 272.67 |

Legend: $P_1$: Thermostat, $P_2$: Furnace, $A$: Furnace Assumption, $Q$: Specification

*: did not terminate in one day (24 hrs)

# Experimental results

- Simulation relation computation expensive
- Heuristic: Look at only reachable states

| #sensors in the furnace model | **Non-AG Method** $P_1 \| \| P_2 \preceq Q$ | | | **AG Method** $P_1 \| \| A \preceq Q$ *and* $P_2 \preceq A$ | | |
|---|---|---|---|---|---|---|
| | # State variables | **Time (s)** | | # State variables | **Time (s)** | |
| | | Full | Reach | | Full | Reach |
| 1 | 4 | 2.10 | 0.10 | 4 *and* 2 | 1.10 | 0.20 |
| 2 | 5 | 121.45 | 0.40 | 4 *and* 3 | 2.85 | 0.30 |
| 3 | 6 | $\infty$* | 1.40 | 4 *and* 4 | 23.51 | 2.80 |
| 4 | 7 | $\infty$* | 23.71 | 4 *and* 5 | 272.67 | 96.02 |

Legend: $P_1$: Thermostat, $P_2$: Furnace, $A$: Furnace Assumption, $Q$: Specification

# Outline

- ✓ **Hybrid systems**
  - Hybrid automata, linear hybrid automata
- ✓ **Checking Conformance to Specification**
  - Simulation relations, computing them
- ✓ **Checking Conformance Compositionally**
  - Assume-guarantee reasoning
- ➢ **Future Research Directions**
- **Summary**

# Future Research Directions – I

- Explicit simulation relation computation using fixed-point approach is **expensive**.

  $\preceq_{n+1} := \preceq_n \setminus B_n$ i.e. set difference $\preceq_n \cap \neg B_n$

- **Alternative to explore:**
  - Developing necessary and sufficient conditions for existence of simulation relations
  - Ex.: Approximate simulation for transition systems with observations in metric spaces [GirardPappas03]

# Future Research Directions – II

- For AG, significant **human effort** is needed in coming up with good assumptions $A$.

$$\frac{\begin{array}{rcl} P_1 || A & \preceq & Q \\ P_2 & \preceq & A \end{array}}{\begin{array}{rcl} P_1 || P_2 & \preceq & Q \end{array}}$$

- **Alternatives to explore:**
  - (Semi-)automating the assumption generation process by doing it iteratively
  - Ex.: Parameter synthesis for LHA [FrehseJhaKrogh08]

# Summary

- ✓ Simulation relations for conformance check
- ✓ AG for compositional reasoning

- ➢ Explicitly generating simulation relations computationally expensive
  - Opportunity for further research
- ➢ Assumption generation needs human effort
  - Opportunity for further research

# References

**[Hen+98] Algorithmic Analysis of Nonlinear Hybrid Systems,** Thomas Henzinger, Pei-Hsin Ho, Howard Wong-Toi

**[Fre08] PHAVer: Algorithmic Verification of Hybrid Systems Past HyTech,** Goran Frehse

**[Fre04] Assume-guarantee Reasoning of Hybrid Systems with Discrete Interaction using Simulation Relations,** Goran Frehse

# Linear Hybrid Automata (LHA) for Analyzing Hybrid Systems (HS)

Akshay Rajhans

ECE Qualifying Exam Presentation, Spring 2010

Apr 20, 2010