Interval and Point-Based Approaches to Hybrid System Verification

Arjun Kapur
Stanford University

August 25, 1997

Advisor: Zohar Manna

Overview

Goal:
Methodologies for proving properties about hybrid systems

Each methodology consists of

Semantics: Model for hybrid system behavior

Specification Language: Logic for specifying properties of hybrid systems

Verification: Approach for proving properties

What are Hybrid Systems?

Hybrid System:

Continuous components
+
Discrete components

Continuous components:

- Variables governed by physical laws
- Variables range over real numbers

Example: Temperature, Air Pressure

Discrete components:

- Variables ranging over discrete set
- Variables range over integers

Example: Semaphores, Switches

Why Verify Hybrid Systems?

Hybrid systems occur everywhere:

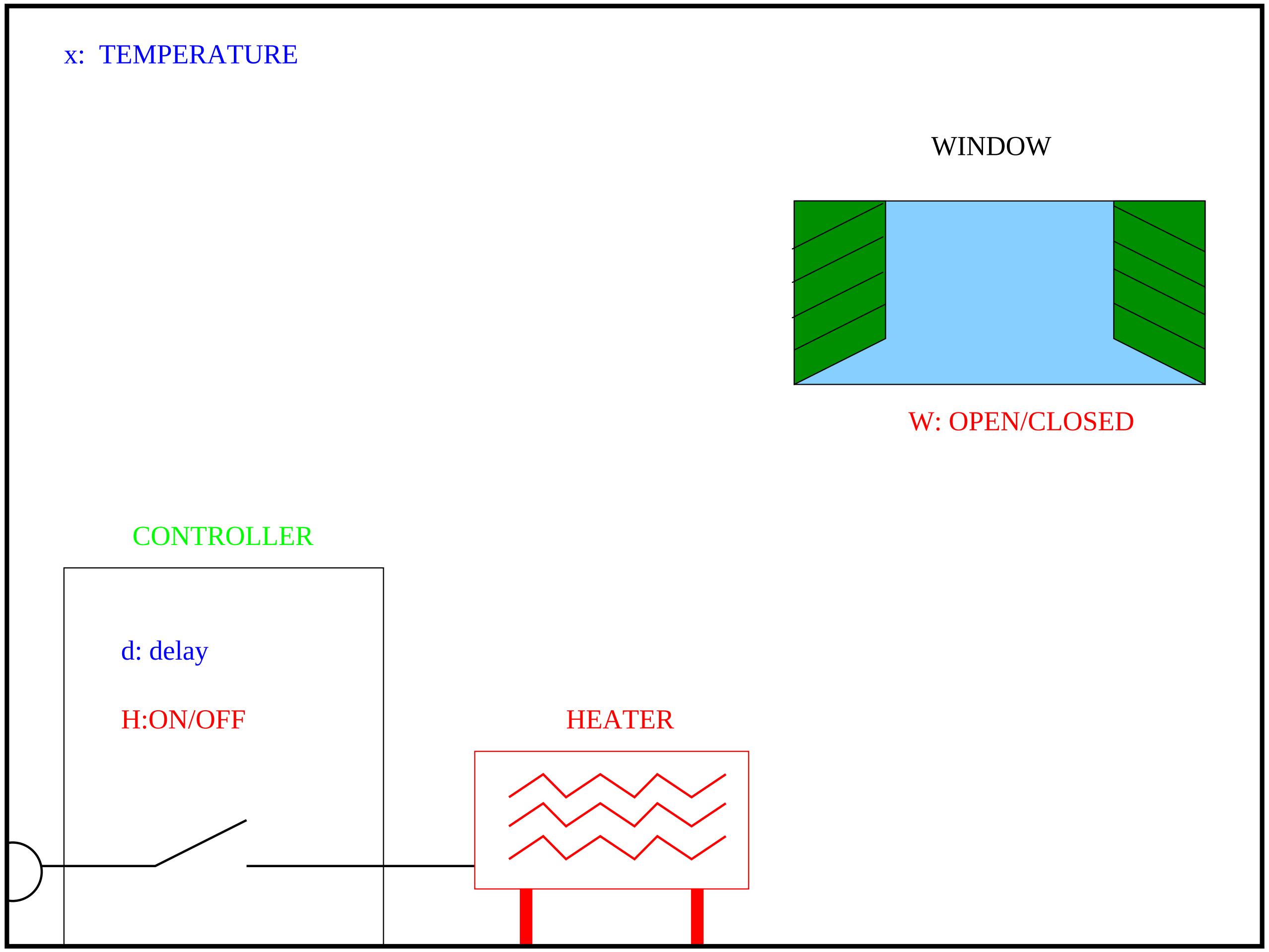
- Airplane Navigation Systems
- Nuclear Power Plants
- Missile Guidance Systems
- Industrial Control Systems

Safety-critical applications. Errors mean:

- Loss of life
- Loss of property

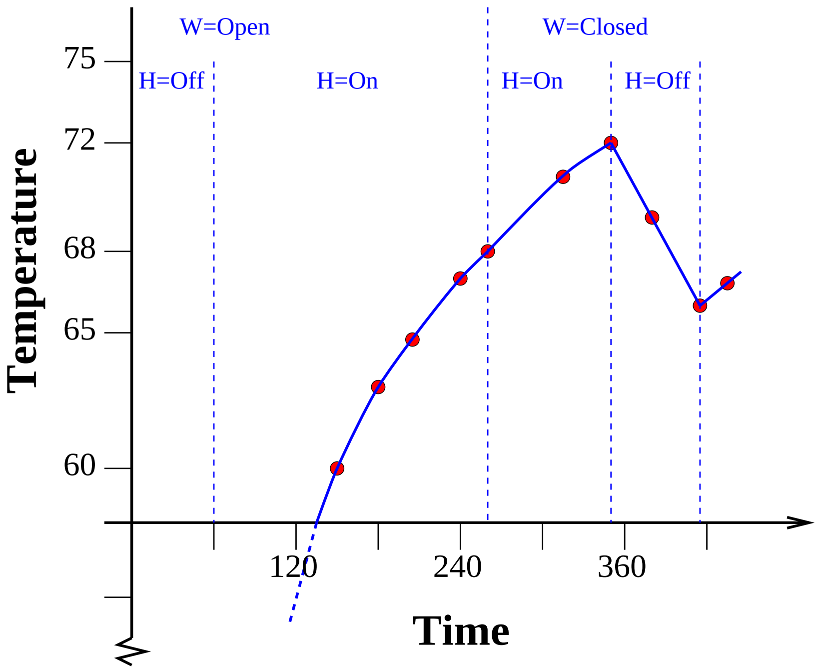
Errors DO occur:

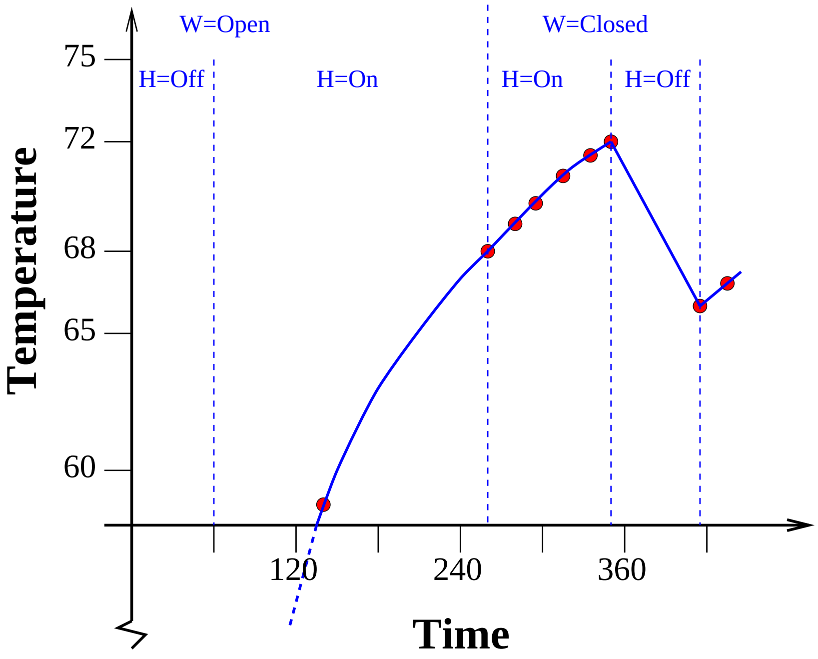
- F-18 crash (1980)
- Ariane 5 launcher (1996)


x: TEMPERATURE
WINDOW
W: OPEN/CLOSED
CONTROLLER
d: delay
H:ON/OFF
HEATER

Sampling Runs

• Sampled Points

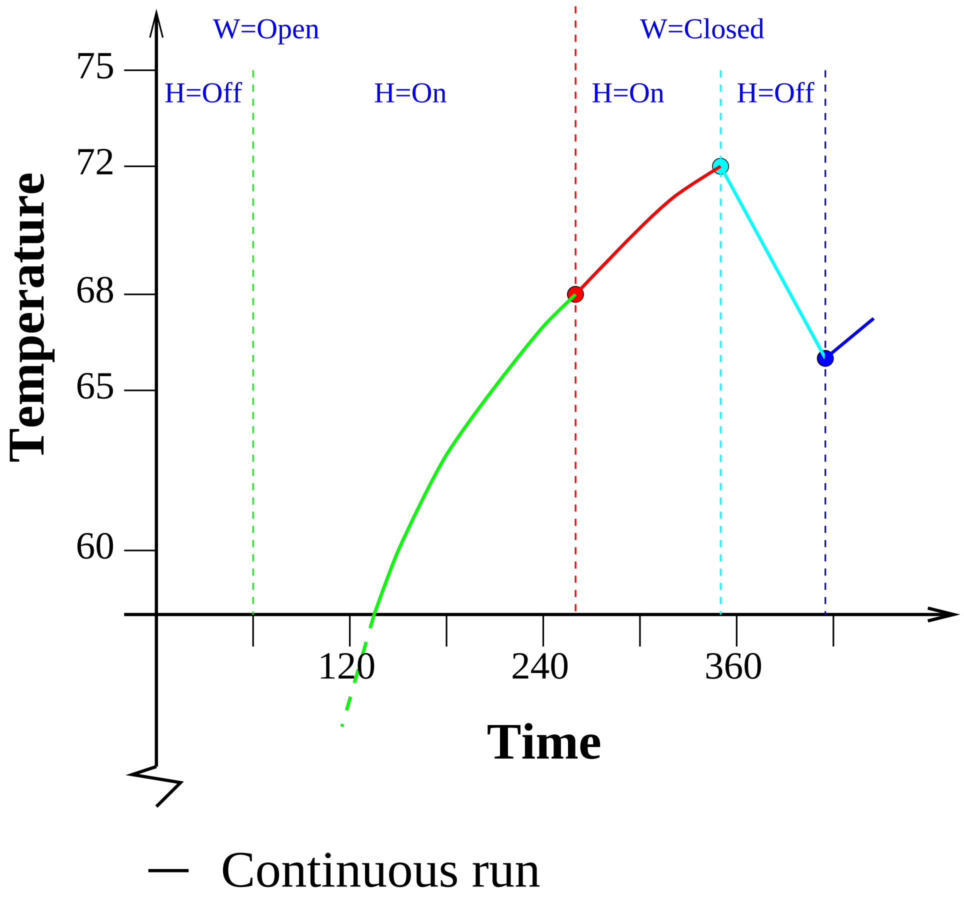




Point-Based Property:

Eventually, temperature is between 65^{o}F and 75^{o}F.

Continuous Runs



Interval-Based Property:
In all intervals of less than 10 seconds, the temperature does not change by more than 5 degrees.

Contributions

Semantics

- Continuous interval semantics
- Super-dense semantics

Specification Languages

- Hybrid Temporal Logic (HTL)
  - Continuous-time logic
  - Decision procedures for fragments of HTL
  - Specification of interval-based properties
- Linear-Time Temporal Logic
  - Extension to a super-dense semantics

Verification

- Diagrams
  - Combine deductive and algorithmic techniques
  - Complete for linear-time temporal logic
- HTL-Verification
  - Rules for proving both point-based and interval-based properties

Outline

	Properties	
Semantics	**Point-based**	**Interval-based**
Sampling	1. Hybrid Diagrams	Stuttering Automata
Continuous	2. HTL Verification Hybrid Diagrams	2. HTL Verification

Hybrid Diagrams

Joint work with de Alfaro and Manna

- Semantics
  Sampling Semantics
- Specification Language
  Linear-Time Temporal Logic
- System Model and Verification Method
  Hybrid Diagrams

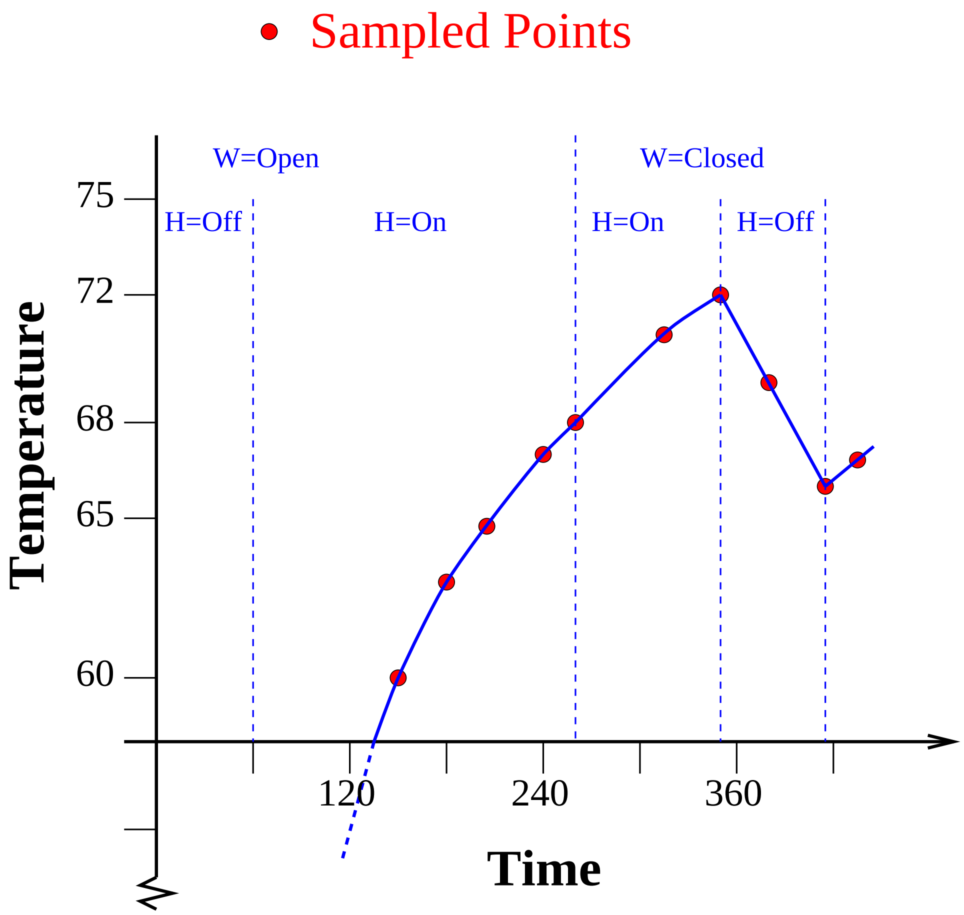
Advantages:

- Visual representation of the system and its properties
- Guidance for constructing proofs
- Incremental proof construction
- Combines deductive and algorithmic techniques
- Complete for proving quantifier-free LTL specifications

Sampling Semantics

- Behavior of system: infinite sequences of snapshots
- Each snapshot records the value of all the variables

Example:



Technical Notes

- Time diverges and is nondecreasing
- Two snapshots may have the same value of T
- $T = 0$ is always sampled

Linear-time Temporal Logic

Formulas φ of **Linear-time Temporal Logic** (LTL):

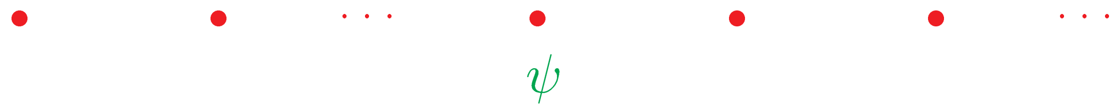
$$\varphi \;:=\; \psi \mid \neg\varphi \mid \varphi_1 \vee \varphi_2 \mid \Box\,\varphi \mid \varphi_1 \mathcal{U} \varphi_2 \mid \forall x \,.\, \varphi$$

where $x \in V$ and ψ is a first-order formula.

σ satisfies the LTL formula φ iff $(\sigma, 0) \models \varphi$

$(\sigma, i) \models \varphi$:

For a first-order formula ψ, $(\sigma, i) \models \psi$ iff ψ evaluates to *true* at state (σ, i).

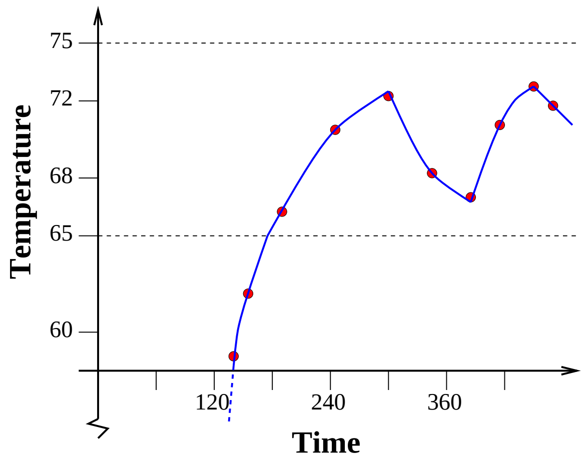


$(\sigma, i) \models \Box\,\varphi$ iff $\forall j \geq i$, $(\sigma, j) \models \varphi$.

$(\sigma, i) \models \varphi_1 \mathcal{U} \varphi_2$ iff $\exists k \geq i$ such that $(\sigma, k) \models \varphi_2$ and $\forall i \leq j < k$, $(\sigma, j) \models \varphi_1$.

System RH Requirements

Specification:



Eventually, $x \in [65, 75]$
Once $x \in [65, 75]$, then
forever $x \in [65, 75]$

Abbreviation: $\Diamond \varphi = \neg \Box \neg \varphi$

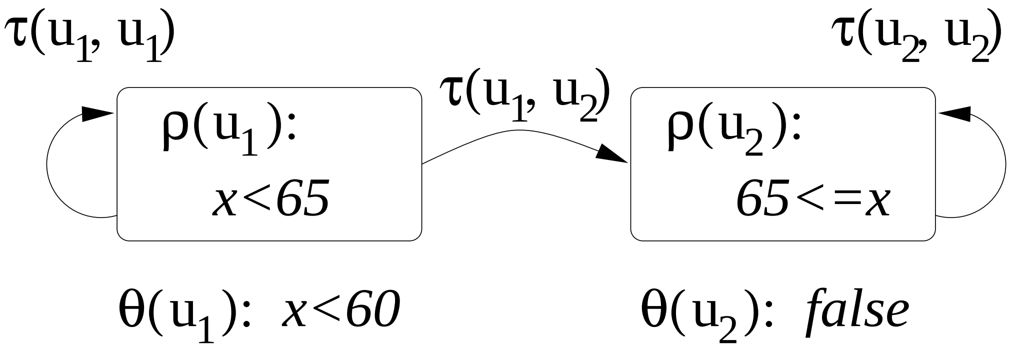
$$\bullet \quad \bullet \quad \cdots \quad \underset{\varphi}{\bullet} \quad \bullet \quad \bullet \quad \cdots$$

Properties

- $\Diamond(65 \leq x \leq 75)$
- $\Box((65 \leq x \leq 75) \rightarrow \Box(65 \leq x \leq 75))$

$$\bullet \quad \bullet \quad \cdots \quad \underset{65 \leq x \leq 75}{\bullet} \quad \underbrace{\bullet \quad \bullet \quad \cdots}_{65 \leq x \leq 75}$$

Hybrid Diagrams



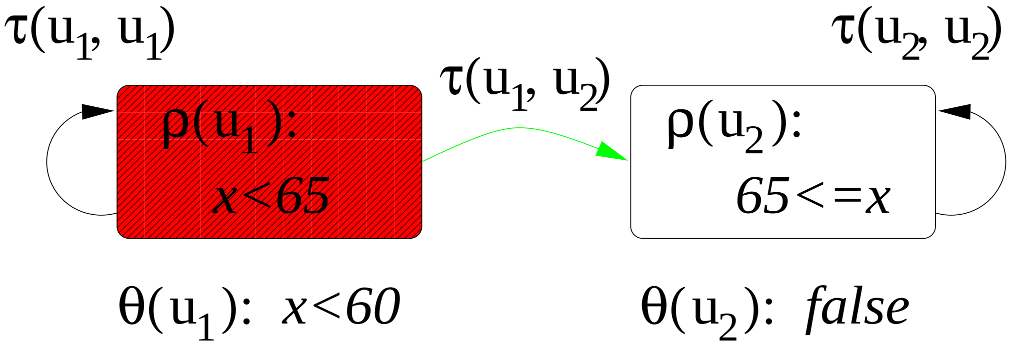
- θ: describes initial states
- ρ: properties proven for vertex
- τ: discrete changes or continuous changes
  Assertions over $\mathcal{V}$, $\mathcal{V}'$, and Δ

Continuous changes: $tick_a[\Delta]$

- Changes in system when time $\Delta \geq 0$ elapses
- Formally,

$$En_a \qquad \text{enabling condition}$$
$$\wedge$$
$$\bigwedge_{x \in \mathcal{V}} (x' = F_a^x[\Delta]) \qquad \text{evolution terms}$$
$$\wedge$$
$$\forall t \in [0, \Delta) \,.\, \Pi[F_a^x[t]/x]_{x \in \mathcal{V}} \qquad \text{urgency}$$

Hybrid Diagrams: Progress Constraints



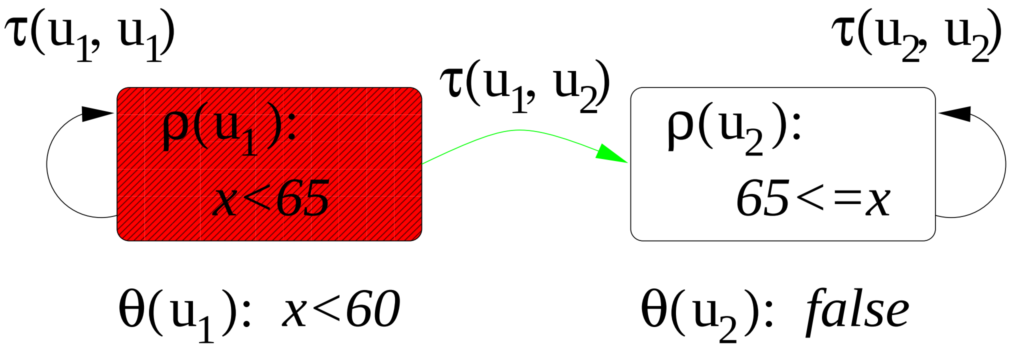
Progress Constraints (R, G) where

R	Request	set of vertices
G	Grant	set of edges

$\mathcal{J}$: Can not visit R continuously without taking G

$\mathcal{C}$: Can not visit R infinitely often without taking G

Diagrams: Sampling Semantics



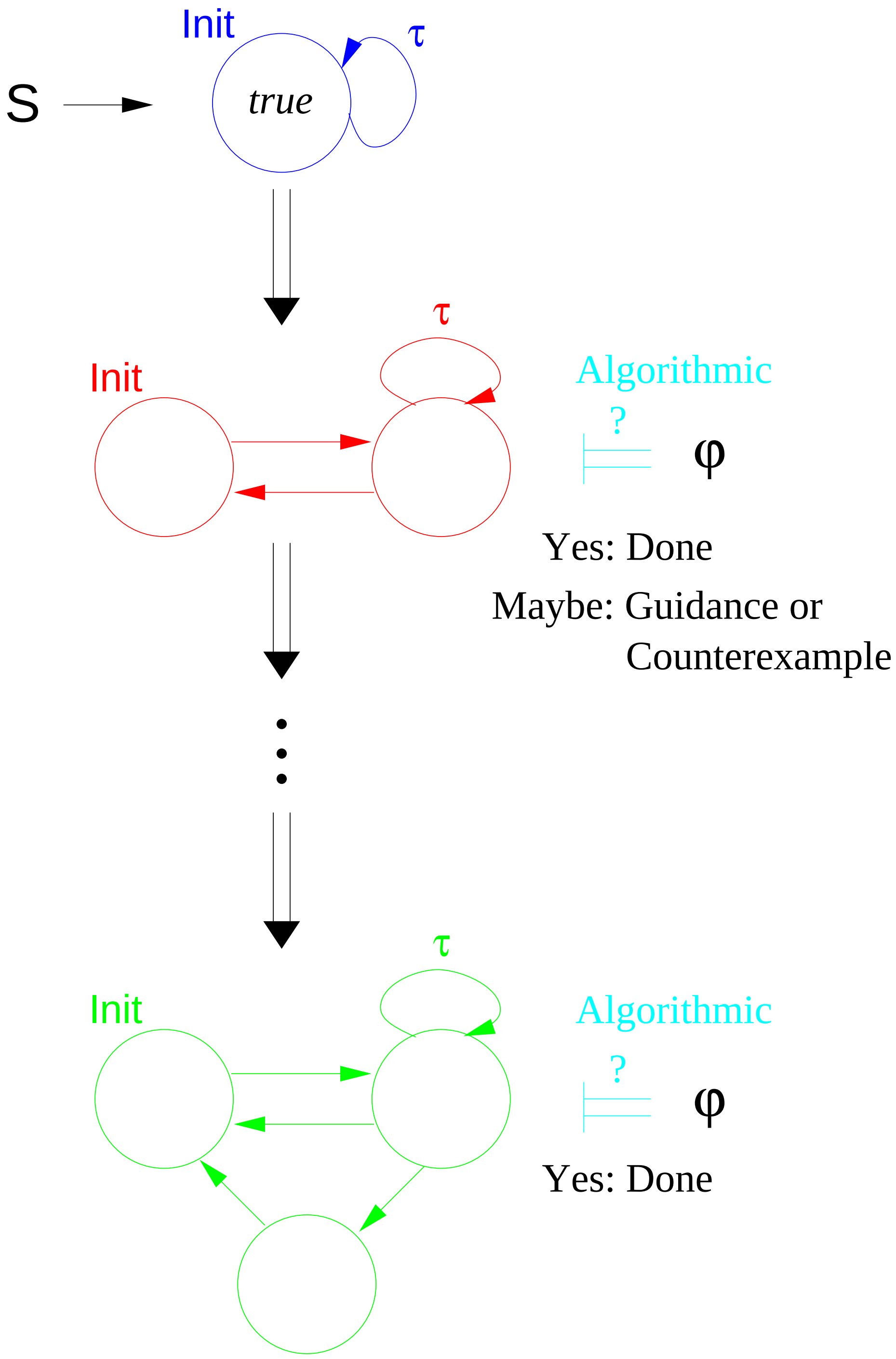
A **run** of a diagram is an infinite sequence of locations $(v_0, s_0), (v_1, s_1), (v_2, s_2), \ldots,$ satisfying:

- Initiality: $s_0 \models \theta(v_0)$
- Vertex labels: $s_i \models \rho(v_i)$ $\quad (\forall i)$
- Edge labels: $(s_i, s_{i+1}) \models \exists \Delta \,.\, \tau(v_i, v_{i+1})$ $\quad (\forall i)$
- Time divergence: $\forall t, \exists i \;\; s_i(T) \geq t$
- Progress Constraints: hold

Run:	$(v_0, s_0),$	$(v_1, s_1),$	$(v_2, s_2),$	$\ldots$
Computation:	$s_0,$	$s_1,$	$s_2,$	$\ldots$
$\mathcal{L}(A)$:	computations of A			

Diagram Approach Overview

Prove: $S \models \varphi$



$$\mathcal{L}(S) \subseteq \mathcal{L}(D_1) \subseteq \ldots \subseteq \mathcal{L}(D_n) \subseteq \mathcal{L}(\varphi)$$

Algorithmic Check

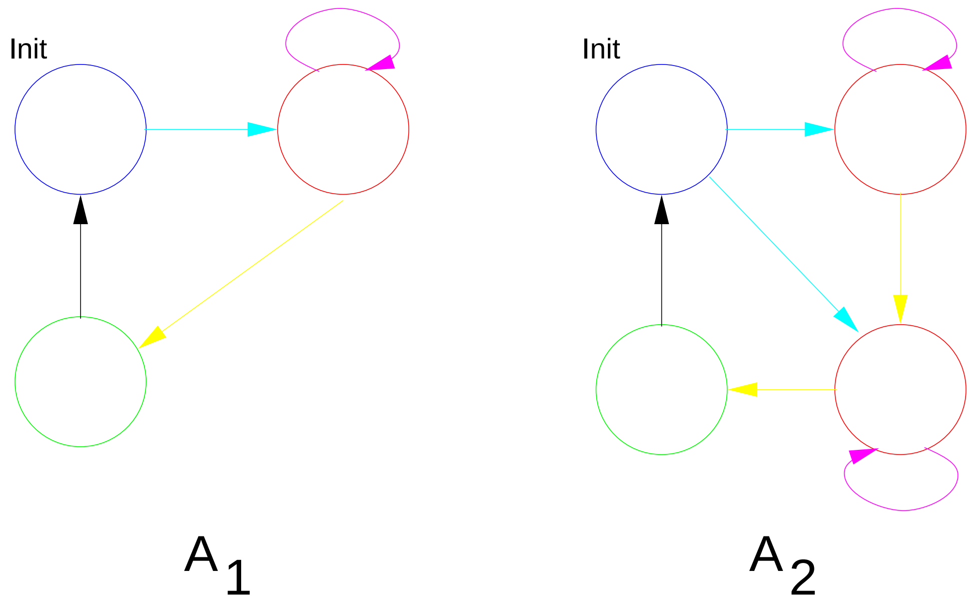
Algorithmic Check:

- Construction of Streett Automaton $N_{\neg\varphi}$ for $\neg\varphi$
- Construction of $D_n \times N_{\neg\varphi}$
- Check $\mathcal{L}(D_n \times N_{\neg\varphi}) = \emptyset$

Possible Results:

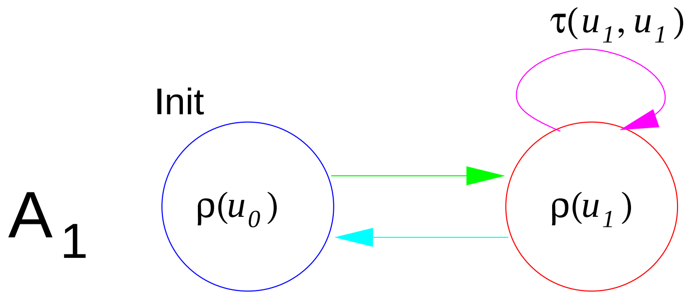
- $\mathcal{L}(D_n \times N_{\neg\varphi}) \neq \emptyset$: φ does not hold
- Model checker has a path that may be a counterexample
- Model checker does not terminate

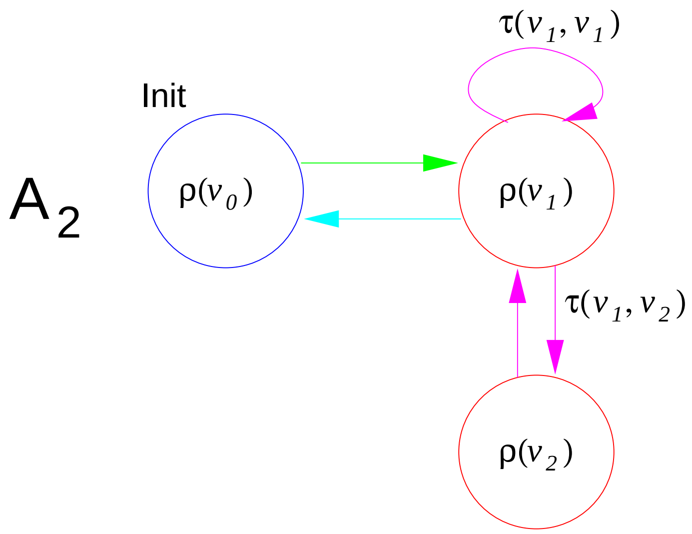
Simulation Rule



- Initial of $A_1 \subseteq$ Inital of A_2
- Every A_1 transition has a corresponding A_2 transition
- Every progress constraint of A_2 is justified by some progress constraint of A_1

Simulation Rule: Example





$\widehat{\tau}(u,v) \overset{\text{def}}{=} \rho(u) \wedge \tau(u,v) \wedge \rho'(v)$ consecution

Every A_1 transition has an A_2 transition

- $(\widehat{\tau}(u_1,u_1) \ \wedge \ \rho(v_1)) \rightarrow (\widehat{\tau}(v_1,v_1) \ \vee \ \widehat{\tau}(v_1,v_2))$

- In general, $\mu : V \mapsto 2^V$, $\forall u, u' \in V_1$, $\forall v \in \mu(u)$,

$$(\widehat{\tau}(u,u') \ \wedge \ \rho(v)) \rightarrow \bigvee_{v' \in \mu(u')} \widehat{\tau}(v,v')$$

Verifying RH: Simulation

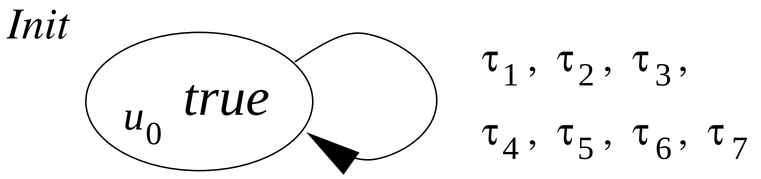
Discrete transitions

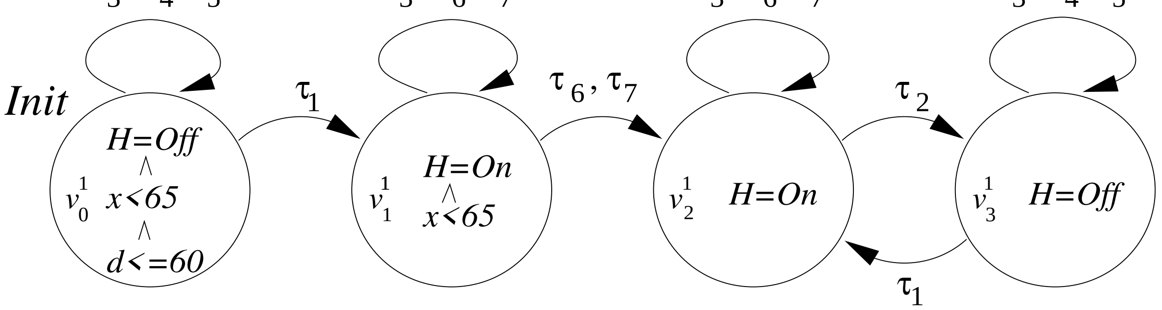
τ_1: Heater turns *On*
τ_2: Heater turns *Off*
τ_3: Window is *Opened*/*Closed*

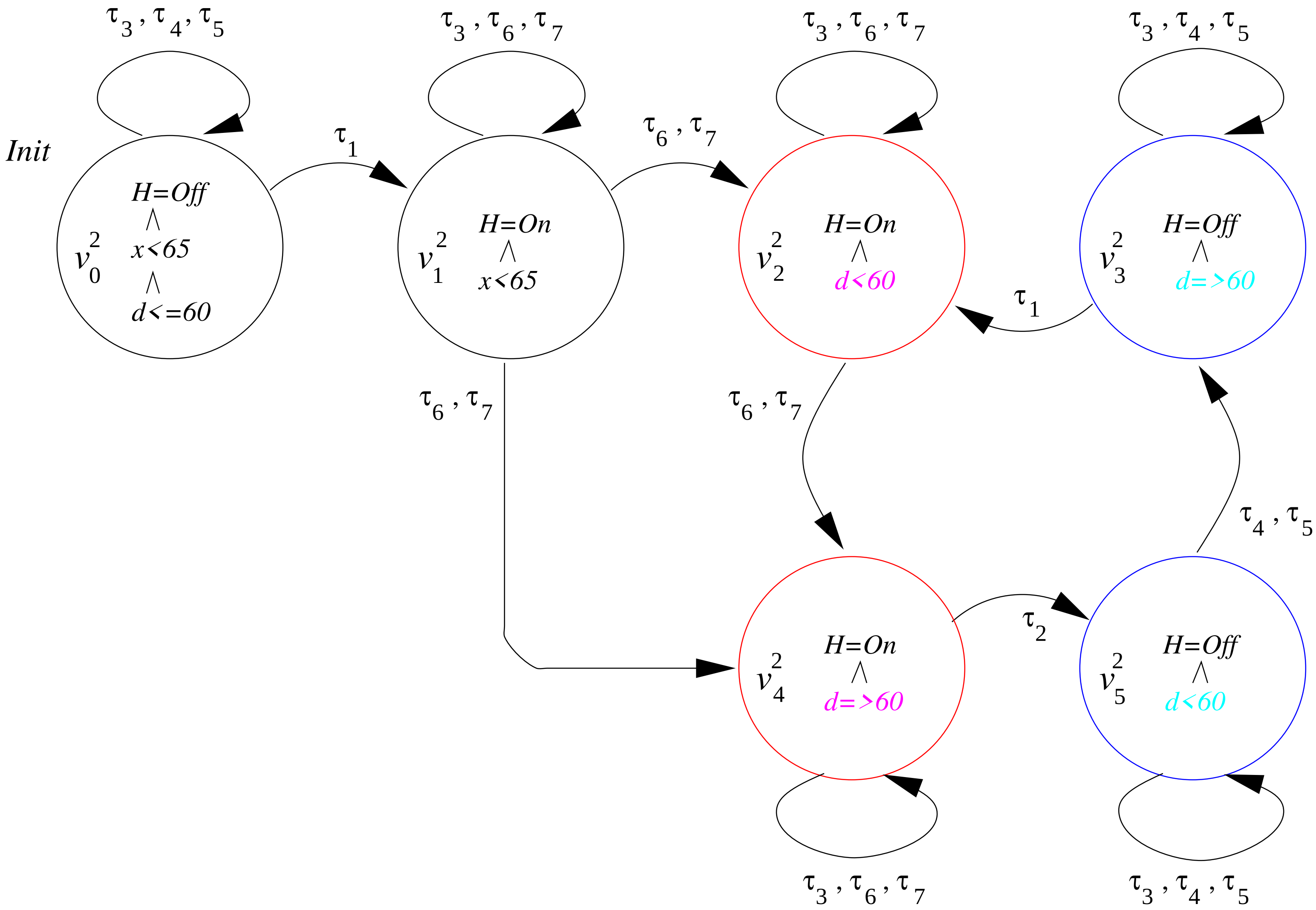
Tick transitions

	Window	
Heater	*Closed*	*Open*
Off	τ_4	τ_5
On	τ_6	τ_7

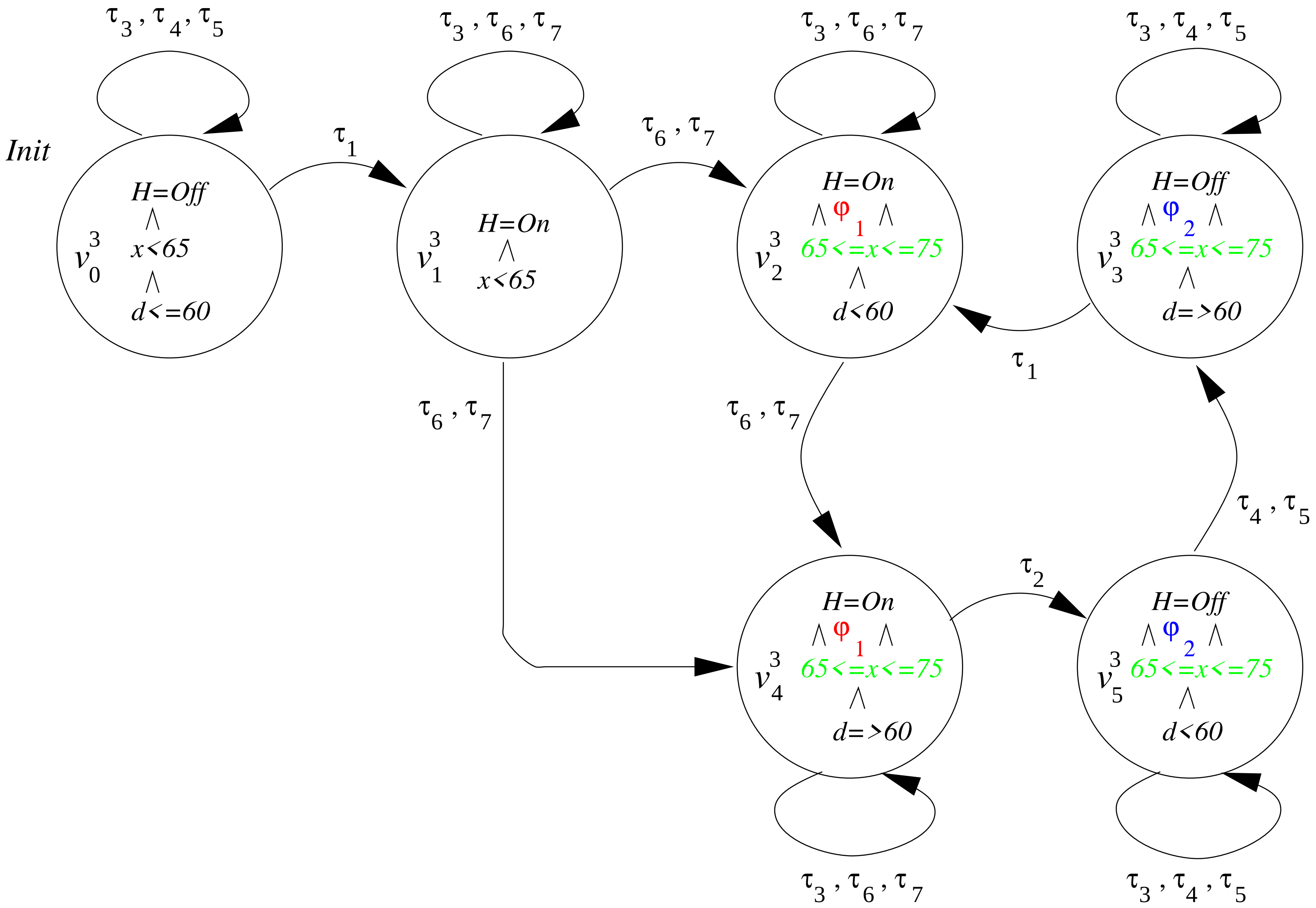
$$\tau_4 : \begin{cases} H = \mathit{Off} \ \wedge \ W = \mathit{Closed} \wedge \\ x' = 60 + e^{-\Delta/105}(x - 60) \wedge \\ T' = T + \Delta \wedge \\ d' = d + \Delta \ \wedge \ \text{urgency} \end{cases}$$







Init
τ_3, τ_4, τ_5
v^2_0
$H{=}Off \wedge x{<}65 \wedge d{<=}60$
τ_1
τ_3, τ_6, τ_7
v^2_1
$H{=}On \wedge x{<}65$
τ_6, τ_7
τ_3, τ_6, τ_7
v^2_2
$H{=}On \wedge d{<}60$
τ_3, τ_4, τ_5
v^2_3
$H{=}Off \wedge d{=>}60$
τ_1
τ_6, τ_7
τ_6, τ_7
τ_4, τ_5
v^2_4
$H{=}On \wedge d{=>}60$
τ_2
v^2_5
$H{=}Off \wedge d{<}60$
τ_3, τ_6, τ_7
τ_3, τ_4, τ_5



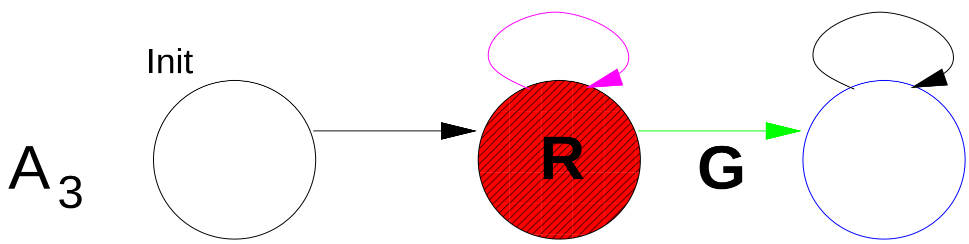
Init
τ3 , τ4, τ5
τ1
τ3 , τ6 , τ7
τ6 , τ7
τ3 , τ6 , τ7
τ3 , τ4 , τ5
H=Off ∧ x<65 ∧ d<=60
v0 3
H=On ∧ x<65
v1 3
H=On ∧ φ1 ∧ 65<=x<=75 ∧ d<60
v2 3
H=Off ∧ φ2 ∧ 65<=x<=75 ∧ d=>60
v3 3
τ1
τ6 , τ7
τ6 , τ7
τ4 , τ5
τ2
H=On ∧ φ1 ∧ 65<=x<=75 ∧ d=>60
v4 3
H=Off ∧ φ2 ∧ 65<=x<=75 ∧ d<60
v5 3
τ3 , τ6 , τ7
τ3 , τ4 , τ5

Adding Constraints

To add a constraint (R, G)**:**

- First, prove all runs of diagram satisfy constraint
- Then, add constraint to justice or compassion set

Thus, language containment is preserved



Measuring Progress

- δ: Ranking function

$$\begin{array}{ccc} \text{locations} & \mapsto & \mathcal{D} \\ (v, s) & & \text{well-founded domain} \end{array}$$

- γ: Delay function

$$\begin{array}{ccc} \text{locations} & \mapsto & \mathcal{R}^+ \\ (v, s) & & \end{array}$$

- Represent functions using assertions $\delta(u)$ and $\gamma(u)$

Idea:

- (δ, γ): how far are we from achieving Grant G
- δ can only decrease
- γ measures the maximum time before δ decreases

Justice Rule

To add (R, G) to Justice Set $\mathcal{J}$

- label every v with $\delta(v)$, $\gamma(v)$

- $\forall (u, v) \notin G$, measure (δ, γ) decreases

$$\widehat{\tau}(u, v) \to \left(\begin{array}{c} \delta(u) > \delta'(v) \\ \vee \\ \delta(u) = \delta'(v) \ \wedge \ \gamma(u) \geq \gamma'(v) + \Delta \end{array} \right)$$

Verifying RH: Progress

Add Justice Constraint

$$\begin{array}{cccc} (& R & , & G &) \\ (& \{v_0, v_1\} & , & \{(v_1, v_2), (v_1, v_4)\} &) \end{array}$$

- A run of A_3 can not stay forever in v_0 or v_1, and must eventually proceed to v_2 or v_4.

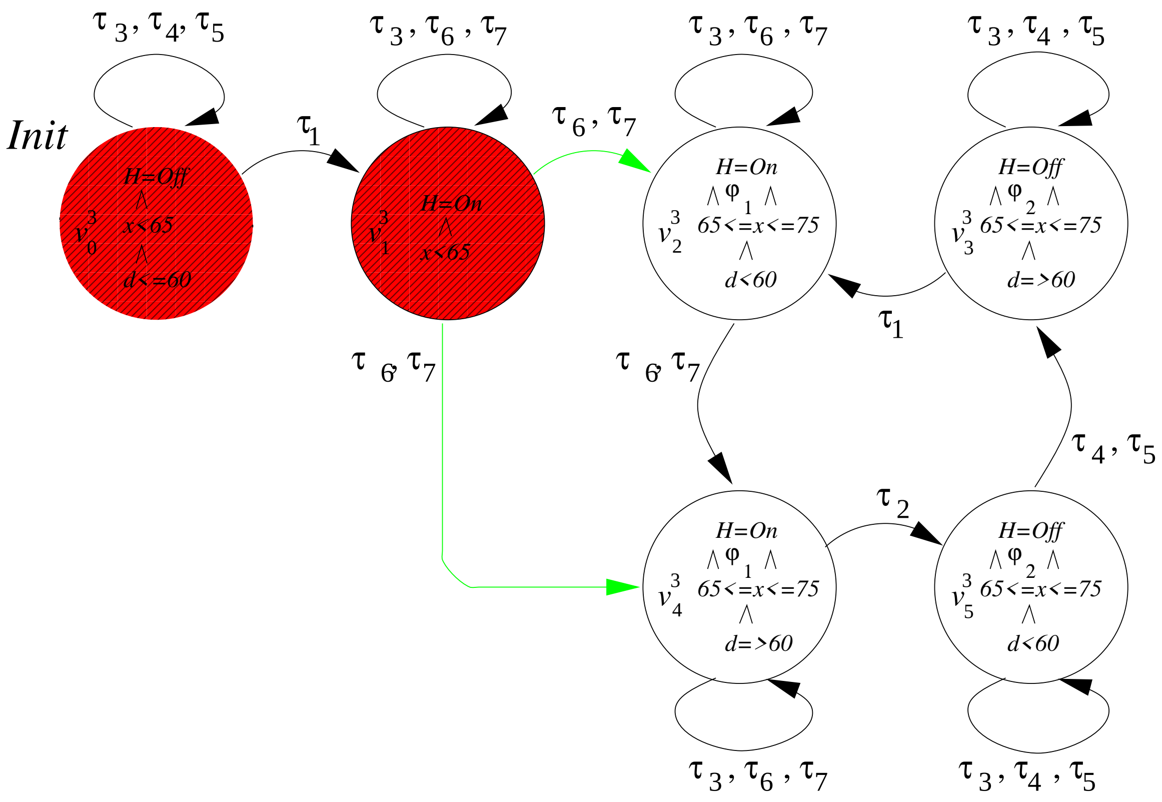
- Ranking functions

$$\begin{aligned} \delta(v_0) &= 1 \\ \delta(v_i) &= 0 \quad (i \in [1..5]) \end{aligned}$$

- Delay functions

$$\begin{aligned} \gamma(v_0) &= 60 - y \\ \gamma(v_1) &= \mathit{if}\ \ x \leq 60 \\ &\qquad\quad \mathit{then}\ \ 175 + 105 \cdot \ln((75 - x)/49) \\ &\qquad\quad \mathit{else}\ \ 150 + 70 \cdot \ln((70 - x)/10) \\ \gamma(v_j) &= 0 \quad (j \in [2..5]) \end{aligned}$$

Verifying RH: Progress

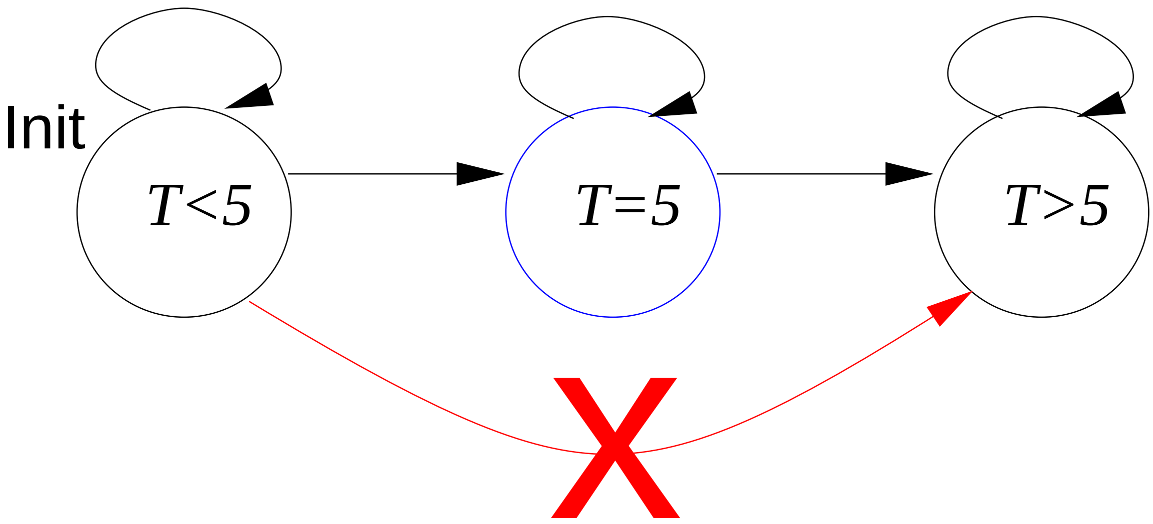


From now on algorithmic

- $\Diamond(65 \leq x \leq 75)$
- $\Box((65 \leq x \leq 75) \rightarrow \Box(65 \leq x \leq 75))$

Continuous Hybrid Diagrams

- Account for intermediate points
- New rules to strengthen and eliminate edges that do not correspond to continuous runs



- Possible to prove properties not valid in the sampling semantics

$$\Diamond(T = 5)$$

Deductive Approach

Joint work with Henzinger, Manna, Pnueli

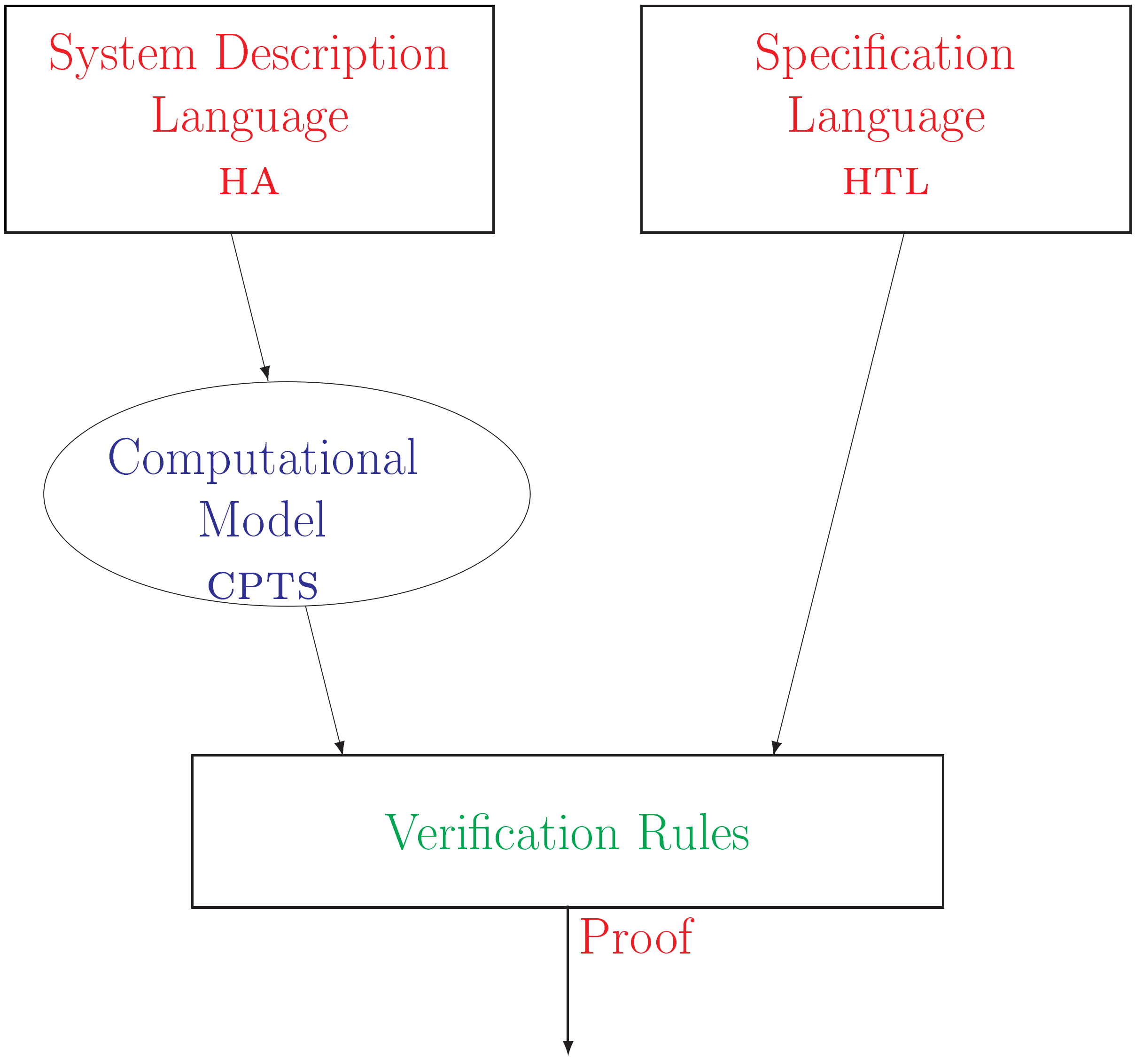
- Semantics
  Continuous Interval Semantics

- Specification Language
  Hybrid Temporal Logic

- System Model
  Concrete Phase Transition Systems

- Verification Method
  Proof Rules

Overview of Deductive Approach



Idea:

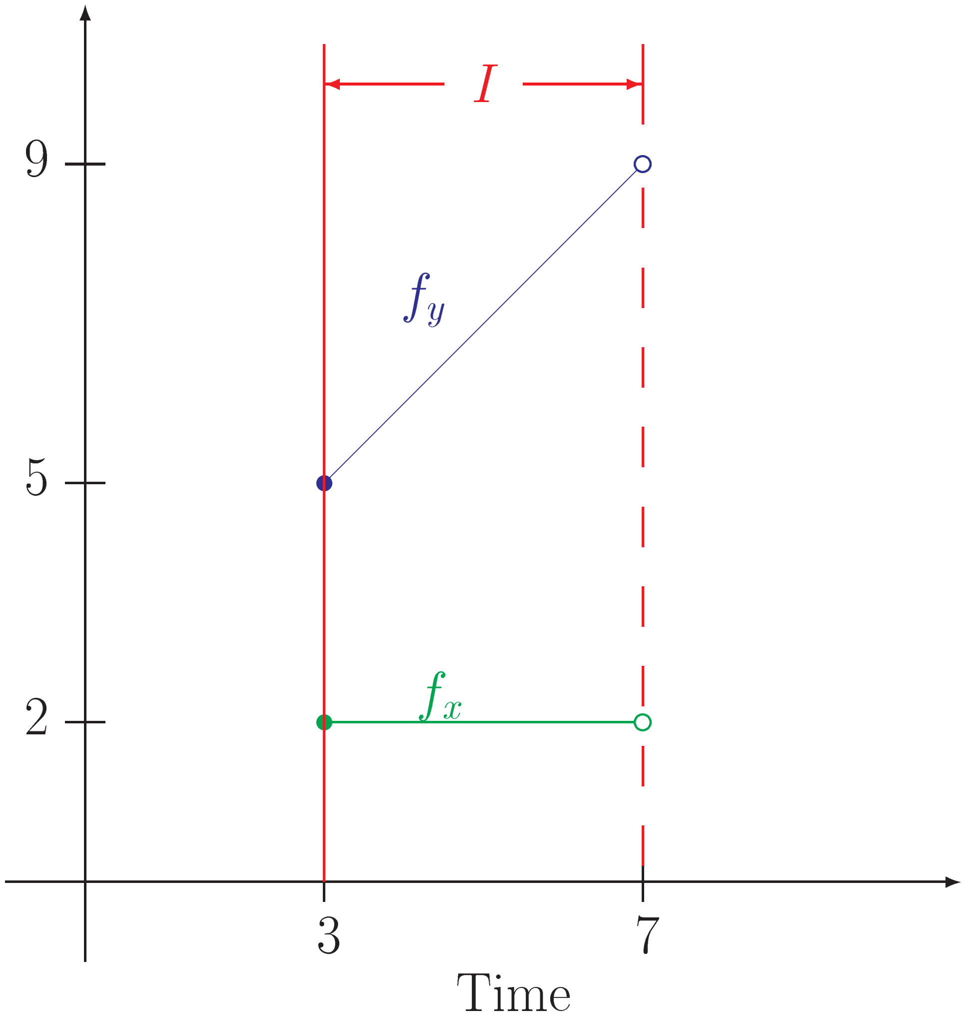
$$\frac{\begin{array}{l}\text{initial states all satisfy } \varphi \\ \text{all transitions preserve } \varphi\end{array}}{\varphi \text{ always holds}}$$

Phases

Phase $P = \langle I, f \rangle$ over V consists of:

- $I = [a, b)$
- type-consistent family $f = \{f_x \mid x \in V\}$ of functions $f_x \colon I \to \mathcal{R}$, where each function
  - assigns to each point $t \in I$ a value for the variable $x \in V$.
  - piecewise smooth in I

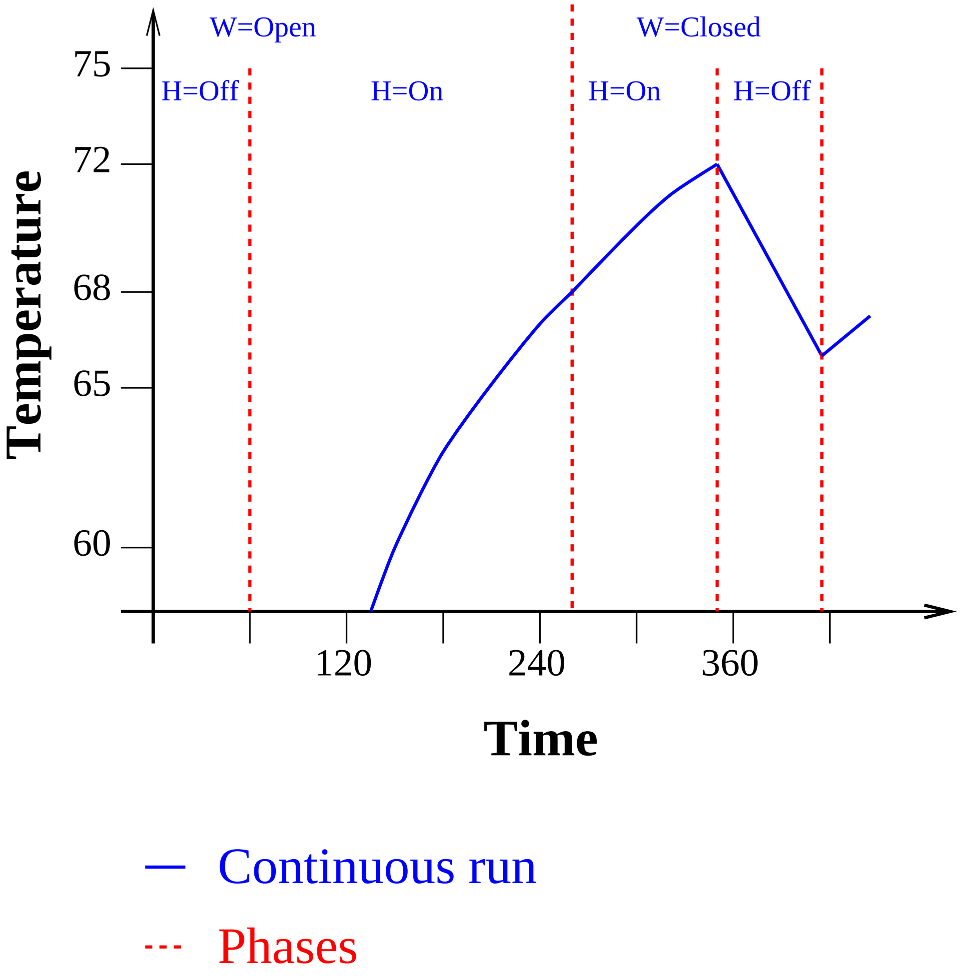
Example: $V = \{x, y\}$ $\qquad$ $P = \langle [3, 7), \{f_x, f_y\} \rangle$



Continuous Interval Semantics

Runs: **Phases** $P = \langle \mathcal{R}^+, f \rangle$

Run Fragment of RH**:** $P = \langle [0, 262.6), f \rangle$



Hybrid Temporal Logic: Syntax

Notation: For a variable $x \in V$

- $\overleftarrow{x}$: limit from the right
- $\overrightarrow{x}$: limit from the left
- $\dot{\overleftarrow{x}}$: right derivative of x (with respect to time)
- $\dot{\overrightarrow{x}}$: left derivative of x (with respect to time)

Local formula: formula over x, $\overleftarrow{x}$, $\overrightarrow{x}$, $\dot{\overleftarrow{x}}$, and $\dot{\overrightarrow{x}}$ (for all $x \in V$), and function and predicate symbols from a language $\mathcal{L}$.

Formulas φ of **Hybrid Temporal Logic** (HTL):

$$\varphi \ := \ \psi \mid \mathbf{fin} \mid \neg\varphi \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 ; \varphi_2 \mid \forall x \, . \, \varphi$$

where $x \in V$ and ψ is an atomic local formula.

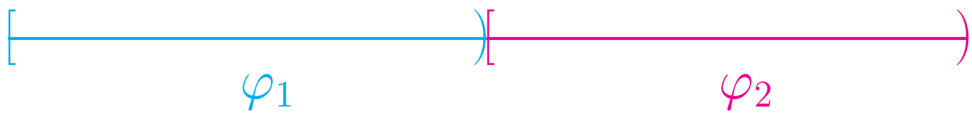
Hybrid Temporal Logic: Semantics

Phase $P = \langle [a, b), f \rangle$ **satisfies** φ ($P \models \varphi$):

- If ψ does not contain left limits or left derivatives, then $P \models \psi$ iff ψ evaluates to *true*

- If ψ contains left limits or left derivatives, then $P \models \psi$ iff $b < \infty$ and ψ evaluates to *true*

- $P \models \varphi_1; \varphi_2$ iff there are two phases, P_1 and P_2, that partition P such that $P_1 \models \varphi_1$ and $P_2 \models \varphi_2$.



- etc.

Hybrid Temporal Logic: Abbreviations

Leftmost subphase:

$$\lhd\varphi \quad \equiv \quad \varphi \lor (\varphi; \mathit{true})$$

Rightmost subphase:

$$\rhd\varphi \quad \equiv \quad \varphi \lor (\mathit{true}; \varphi)$$

Some subphase:

$$\Diamond\varphi \quad \equiv \quad (\lhd\varphi) \lor (\rhd\varphi) \lor (\mathit{true}; \varphi; \mathit{true})$$

All subphases:

$$\Box\varphi \quad \equiv \quad \neg\Diamond\neg\varphi$$

All finite subphases:

$$\Box_f\varphi \quad \equiv \quad \Box(\mathbf{fin} \rightarrow \varphi)$$

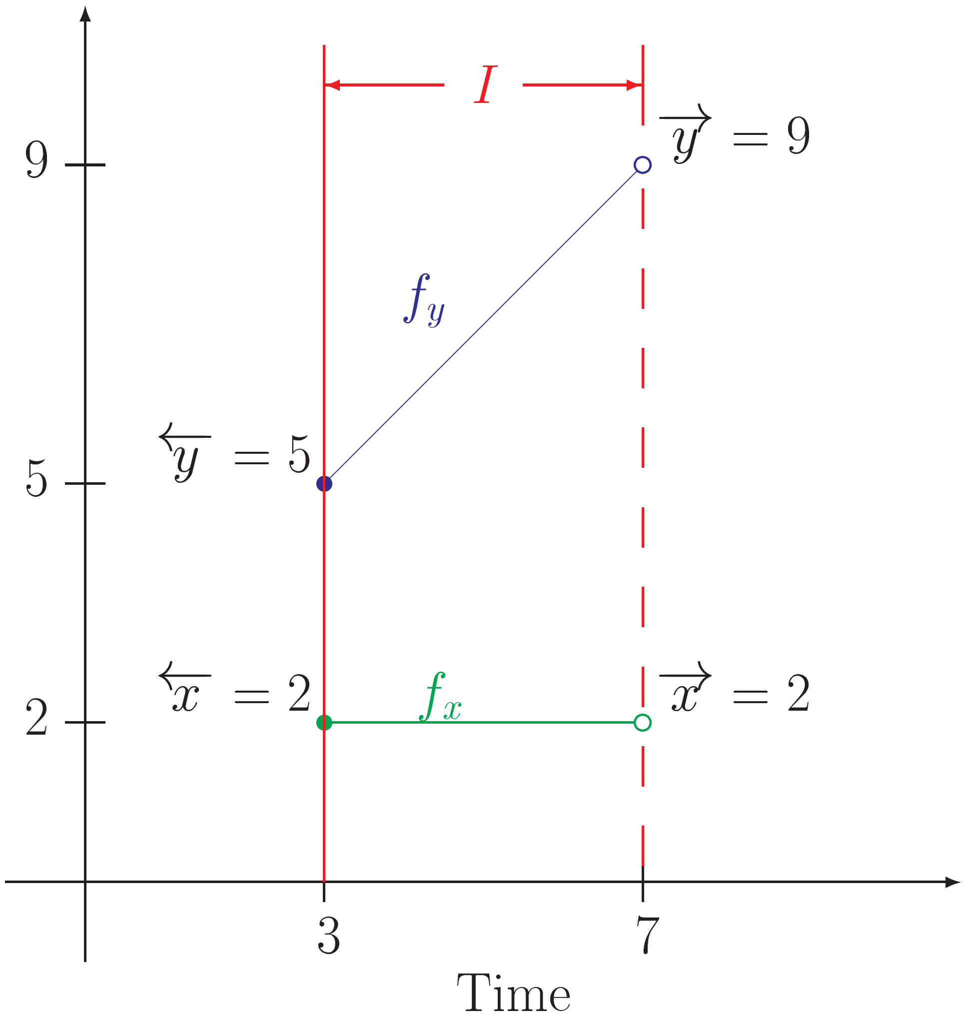
Temporal entailment:

$$\varphi \Rightarrow\!\succ_f \psi \quad \equiv \quad \Box_f(\varphi \rightarrow \psi)$$

Example

$$V = \{x, y\} \qquad P = \langle [3, 7), \{f_x, f_y\} \rangle$$

$$P \models \quad \begin{array}{ll} \overrightarrow{y} = 9 & \Box(\overleftarrow{x} = 2) \\ \overleftarrow{y} = 5 & \Box(\overrightarrow{y} > \overleftarrow{y}) \\ \dot{\overleftarrow{y}} = 1 & \overleftarrow{y} = 5; \overleftarrow{y} = 7 \end{array}$$



Phase Invariants

Phase invariants: a first-order formula $\rho_\phi(V, \dot{V})$ in the two variable tuples V and $\dot{V}$

Use: proving properties

Phase $P = \langle I, f \rangle$ over V is a ϕ-phase if

$$P \models \textbf{continuous} \wedge \Box(\rho_\phi(V, \dot{V}))$$

continuous: HTL formula stating all variables are "continuous" over the phase P.

Example:

$$\rho_\phi(V, \dot{V}):\ 5 \leq y < 9 \ \wedge\ \dot{y} = 1$$

characterizes all phases in which y steadily increases at a rate of 1 and always remains within the interval $[5, 9)$.

Concrete Phase Transition System

Definition: A CPTS $\mathcal{S} = (V, \Phi, \Theta, \mathcal{T})$ consists of:

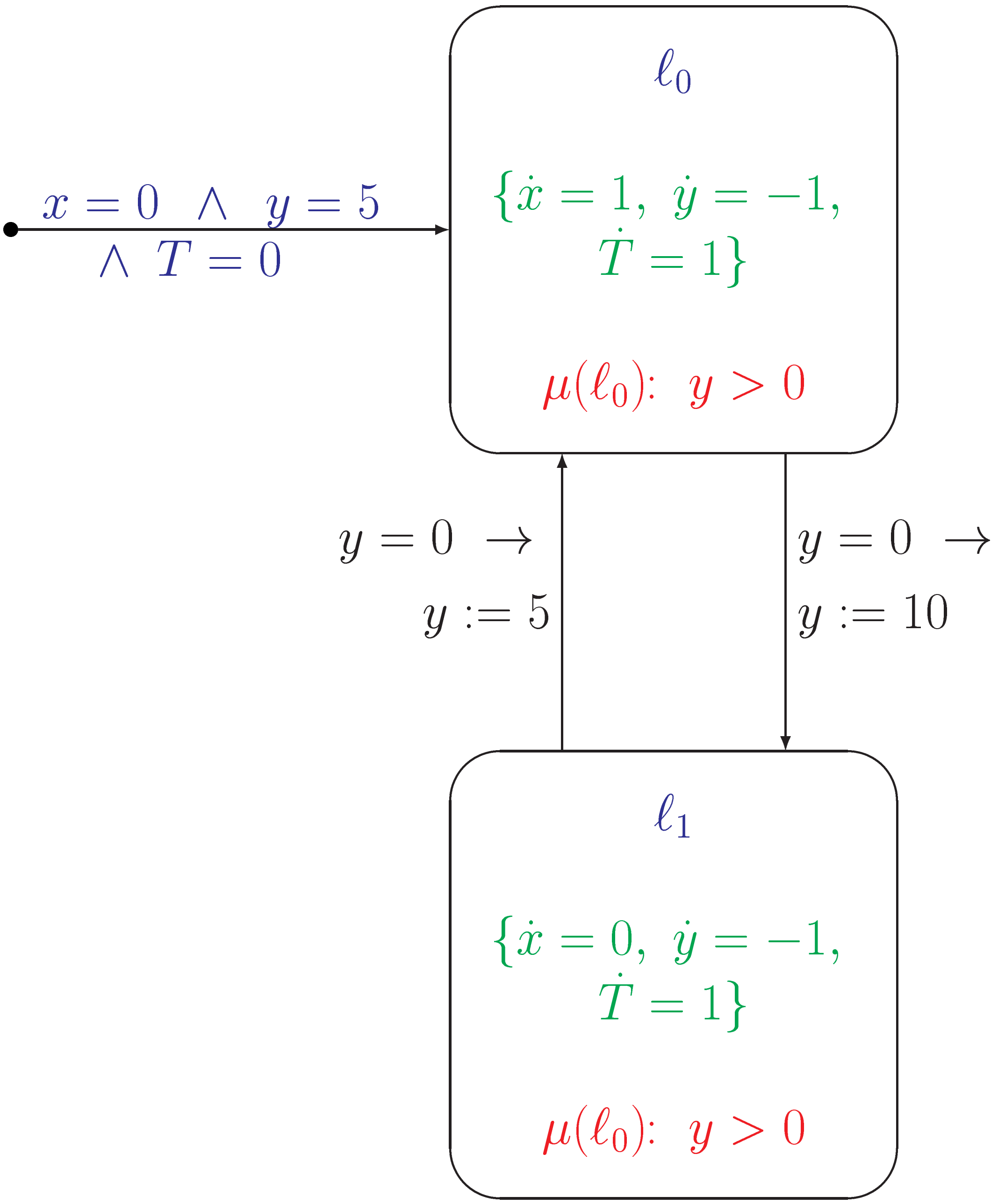
- State variables: V

- Phase invariants: Φ over V
  each phase invariant $\phi \in \Phi$ is presented by an assertion of the form $\rho_\phi(V, \dot{V})$

- Initial condition: Θ, state formula over V

- Discrete transitions: $\mathcal{T}$
  each transition $\tau \in \mathcal{T}$ is associated with an assertion $\rho_\tau(V, V')$, relating values at the right-end of a phase to the values at the left-end of a successor phase.

Trivial Example



Rule P-INV

Given: CPTS $\mathcal{S} = (V, \Phi, \Theta, \mathcal{T})$, first-order formula $\psi(V)$
φ: strengthened version of ψ
χ: weaker version of φ

PI1.	$\Theta \rightarrow \varphi(V)$	
PI2.	$\varphi(V) \rightarrow \psi(V)$	
PI3.	$\rho_\tau(V, V') \wedge \chi(V) \rightarrow \varphi(V')$	$\forall \tau \in \mathcal{T}$
PI4.	$\rho_\phi(V, \dot{V}) \wedge \chi(V) \rightarrow \varphi(V)$	$\forall \phi \in \Phi$
PI5.	**continuous** $\wedge \, \Box \, \rho_\phi(V, \dot{V}) \wedge \varphi(V) \Rightarrow\!\succ_f \chi(\overrightarrow{V})$	$\forall \phi \in \Phi$

$$\Box\, \psi(V)$$

Illustration of P-INV

$$\begin{array}{llll} \varphi_1: & at\_\ell_0 \ \rightarrow \ 0 < y \leq 5 & \chi_1: & at\_\ell_0 \ \rightarrow \ 0 \leq y \leq 5 \\ \varphi_2: & at\_\ell_1 \ \rightarrow \ 0 < y \leq 10 & \chi_2: & at\_\ell_1 \ \rightarrow \ 0 \leq y \leq 10 \end{array}$$

To prove $\square\, \varphi_1$

PI1. $y = 5 \ \rightarrow \ 0 < y \leq 5$

PI3. $y = 0 \ \wedge \ y' = 5 \ \wedge \ \chi_1(V) \ \rightarrow \ \varphi_1(V')$

PI4. $at\_\ell_0 \ \wedge \ y > 0 \ \wedge \ \ldots \ \wedge \ \chi_1(V) \ \rightarrow \ \varphi_1(V)$

PI5. $\textbf{continuous} \ \wedge \ \square(\dot{y} = -1 \ \wedge \ y > 0) \ \wedge \ \varphi_1(V) \quad \Rightarrow\!\succ_f \quad \chi_1(\overrightarrow{V})$

Rule I-INV

Given: CPTS $\mathcal{S} = (V, \Phi, \Theta, \mathcal{T})$, formula $\psi(\overleftarrow{V}, \overrightarrow{V})$
φ: strengthed version of ψ

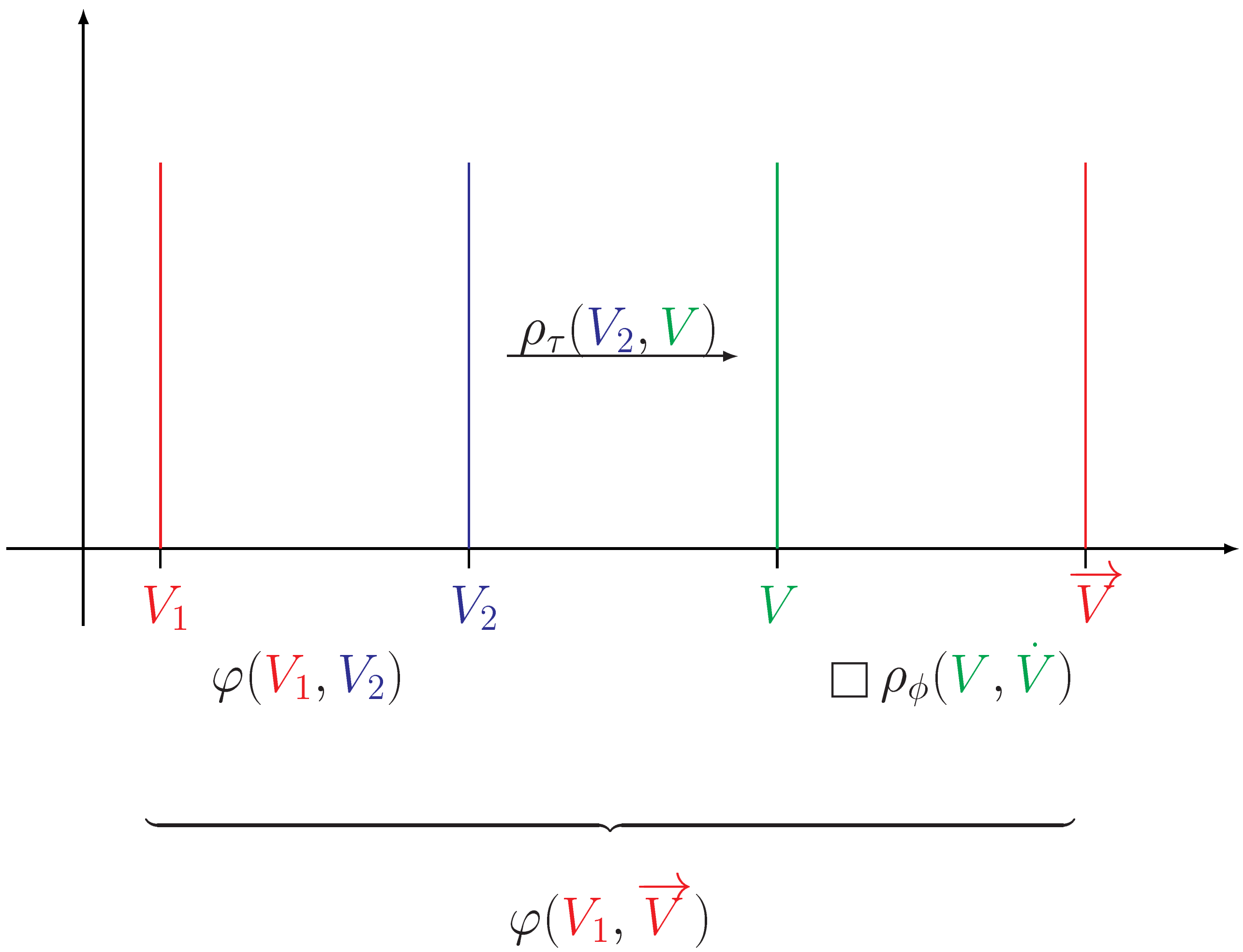
$$
\begin{array}{ll}
\text{II1.} & \varphi(V, V') \;\rightarrow\; \psi(V, V') \\[1em]
\text{II2.} & \textbf{continuous} \wedge \Box\, \rho_\phi(V, \dot{V}) \Rightarrow\!\!\succ_f \varphi(\overleftarrow{V}, \overrightarrow{V}) \quad \forall \phi \in \Phi \\[1em]
\text{II3.} & \begin{array}{lll} \varphi(V_1, V_2) \wedge \rho_\tau(V_2, V) \wedge \textbf{continuous} & & \forall \tau \in \mathcal{T} \\ \wedge\, \Box\, \rho_\phi(V, \dot{V}) \quad \Rightarrow\!\!\succ_f \quad \varphi(V_1, \overrightarrow{V}) & & \forall \phi \in \Phi \end{array} \\
\hline
 & \Box_f\, \psi(\overleftarrow{V}, \overrightarrow{V})
\end{array}
$$

Illustration of I-INV

Premise II3:

$$\begin{array}{ll} \varphi(V_1, V_2) \ \wedge \ \rho_\tau(V_2, V) \ \wedge \ \textbf{continuous} & \forall \tau \in \mathcal{T} \\ \wedge \ \Box\, \rho_\phi(V, \dot{V}) \quad \Rightarrow\!\!\succ_f \quad \varphi(V_1, \overrightarrow{V}) & \forall \phi \in \Phi \end{array}$$

Key Idea:



Related Work: Semantics

- Sampling Semantics
  - MP: Models for Reactivity
  - HMP: Timed Transition Systems
  - AH: Real-Time Logics
  - Heitmeyer, Lynch: Generalized Railroad Crossing
- Super-dense Semantics
  - MMP: From Timed to Hybrid Systems
  - Bouajjani, Lakhnech: Logic vs. Automata

Related Work: Specification Languages

- Real-Time Logics
  - Integrator Computation Tree Logic: HyTech
  - Duration Temporal Logic: Bouajjani, Echahed, Sifakis
  - Duration Interval Logic: Bouajjani, Lakhnech, Robbana
  - Hybrid Automata Temporal Logic: Bouajjani, Lakhnech
- Duration Calculus: Chaochen, Hoare, Ravn, others
- Temporal Logic of Actions: Lamport
- Interval Logics with chop: Moszkowski, Rosner and Pnueli
- Hybrid CC: Gupta, Jagadeesan, Saraswat

Related Work: Verification

- Deductive Approaches
  - Temporal Logic of Actions: Lamport
  - Timed I/O Automata: Lynch and Vaandrager
  - Transition Systems: Manna and Pnueli, Nicollin, Sifakis, et al
  - PVS: Shankar, Skakkebæk
- Algorithmic Approaches
  - HyTech: Henzinger, Ho, others
  - Integration Graphs: Kesten, Pnueli, Sifakis, Yovine
- Diagrammatic Approaches: Reactive Systems
  - Fairness Diagrams: de Alfaro and Manna
  - Verification Diagrams: Browne, Manna, Sipma
  - Deductive Model Checking: Manna, Sipma, Uribe