

Synthesis of Linear Hybrid Automata from Experimental Data

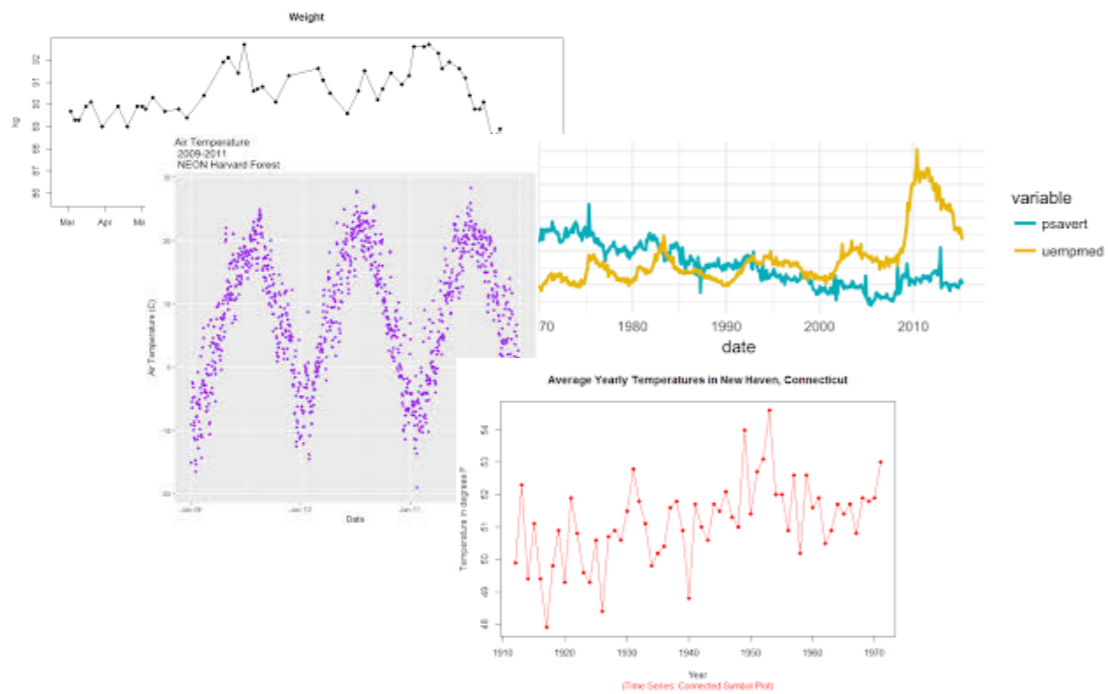
Miriam García Soto

Co-authored work with Thomas A. Henzinger, Christian Schilling and Luka Zeleznik

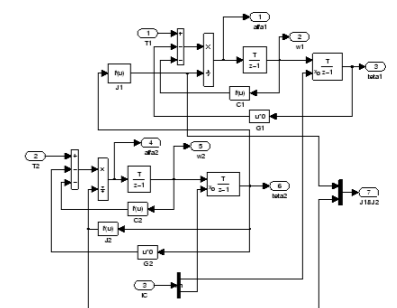
IST Austria - 2019

Motivation

Main goal of many sciences is to create a model from a real system



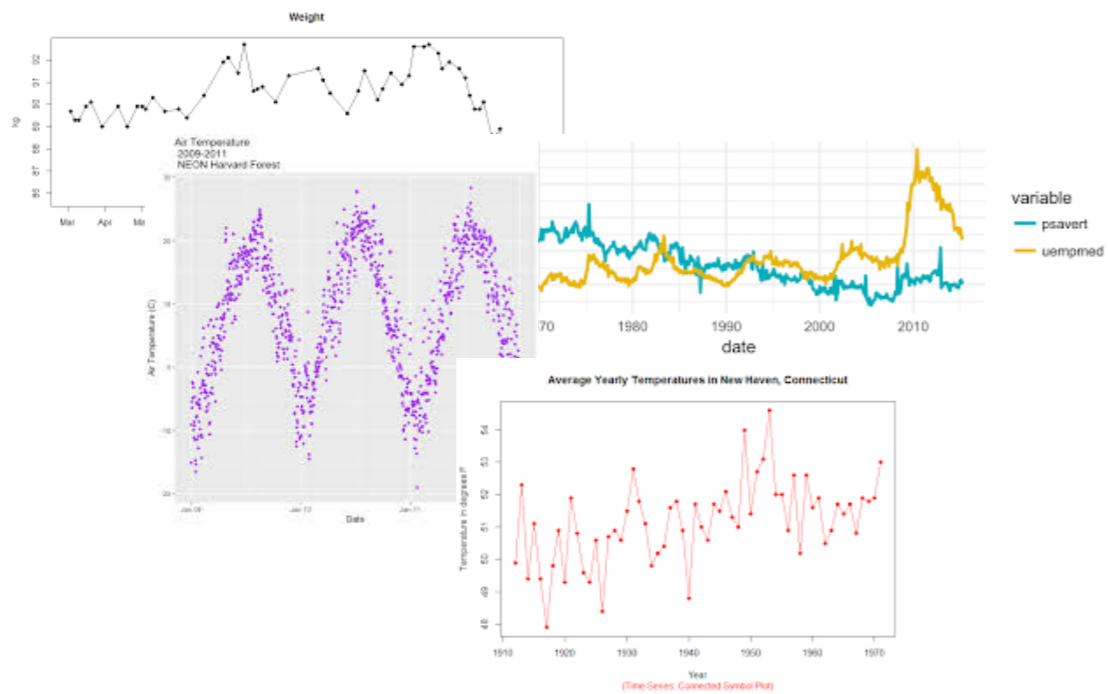
Experimental data



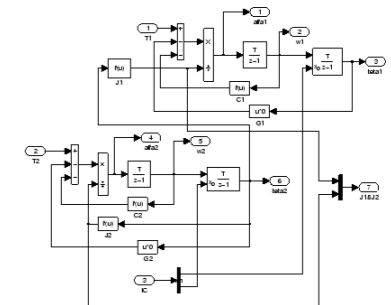
Model

Motivation

Main goal of many sciences is to create a model from a real system



Experimental data

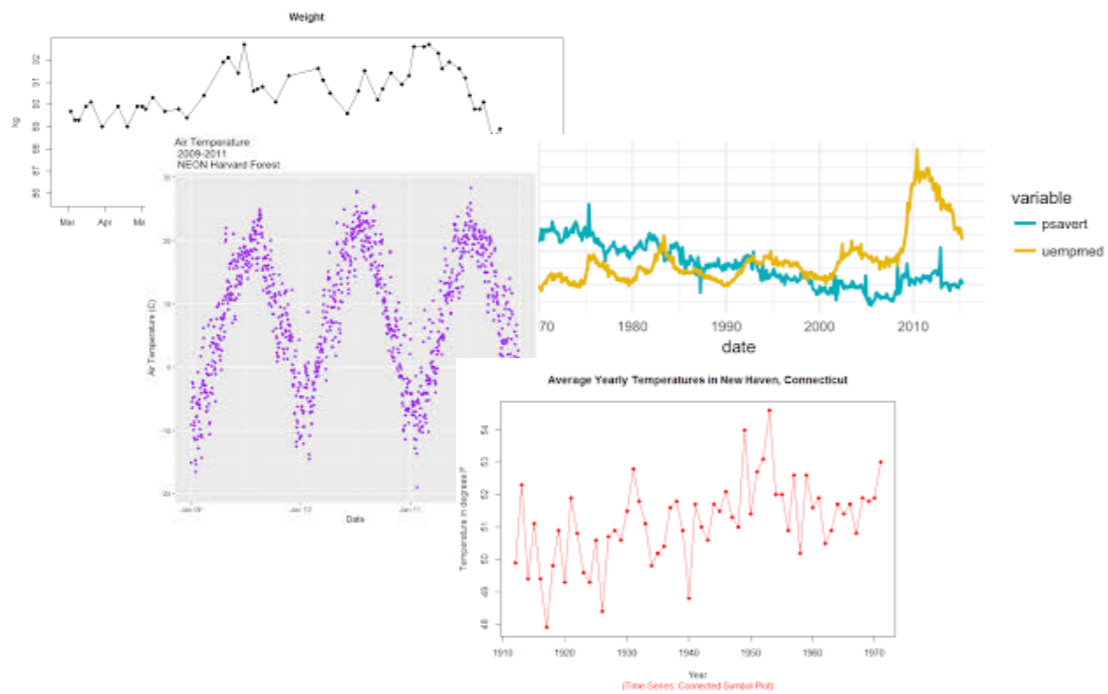


Model

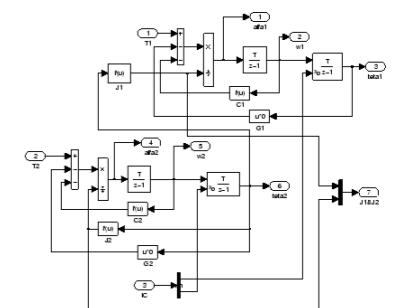
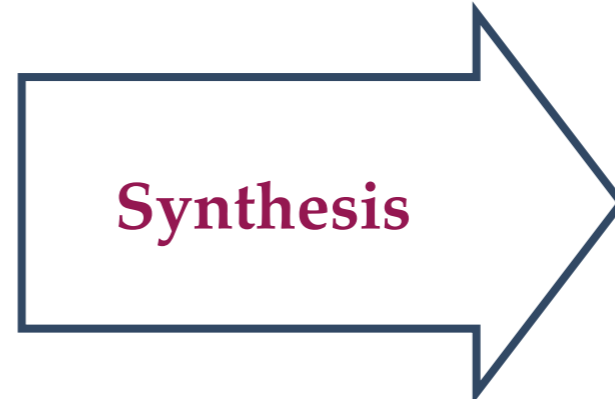
Challenge
How to automatically create a model?

Motivation

Main goal of many sciences is to create a model from a real system



Experimental data



Model

Challenge
How to automatically create a model?

Model: Linear Hybrid Automaton

Hybrid behavior

The Journal of Neuroscience, May 12, 2005 • 25(18):4389–4397 • 4389

Step Response Analysis of Thermo taxis in *Caenorhabditis elegans*

Hatim A. Zariwala,^{1,*} Adam C. Miller,^{2,*} Serge Faumont,^{2,*} and Shawn R. Lockery¹
¹Graduate Program in Neurobiology and Behavior, State University of New York at Stony Brook, Stony Brook, New York 11794, and ²Department of Neurobiology and Behavior, State University of New York at Stony Brook, Stony Brook, New York 11794

The nematode *Caenorhabditis elegans* migrates toward a preferred temperature on a thermotaxis in *C. elegans* has been identified, but the behavioral strategy implemented by this we tested whether thermal migration is achieved by modulating the probability of turning be done by subjecting unrestrained wild-type, cryophilic, or thermophilic worms to rapid spat down from the cultivation temperature. Each of the three types of worms we analyzed showed of steps. Comparison of wild-type and mutant response patterns suggested a model in a response to the gradient depending on the orientation of the worm relative to its prefer probability was modulated in a manner consistent with a role for turning behavior in thermat phase for thermotaxis and chemotaxis may converge on a common behavioral mechanis **Key words:** *C. elegans*; nematode; thermotaxis; spatial orientation; stochastic model; sensory

Introduction

Caenorhabditis elegans orients to both chemical (chemotaxis) and thermal (thermotaxis) gradients (Ward, 1973; Hedgecock and Russell, 1975), making it a promising experimental system for investigating the neuronal basis of spatial orientation. Previous studies have established a plausible behavioral mechanism for chemotaxis in *C. elegans* (Dusenberry, 1980; Pierce-Shimomura et al., 1999). Locomotion consists of periods of relatively straight-forward movement punctuated approximately twice per minute by bouts of turning (Rutherford and Croll, 1979). Two main kinds of turns are recognized in *C. elegans*: “reversals,” in which the animal moves backward for several seconds and then goes forward again in a new direction, and “omega turns,” in which the animal’s head bends around to touch the tail during forward locomotion, momentarily forming a shape like the Greek letter (Croll, 1975b). Statistical analysis reveals that reversals and omega turns occur in bursts that have been termed piroettes (Pierce-Shimomura et al., 1999). Piroette probability is modulated by the rate of change of chemical concentration (dC/dt): when $dC/dt < 0$, piroette probability is increased, whereas when $dC/dt > 0$, piroette probability is decreased. Thus, runs down the gradient are truncated, and runs up the gradient are extended, resulting in net movement toward the gradient peak.

The piroette strategy in chemotaxis suggests a reasonable hypothesis for the behavioral mechanism of the migration phase of thermotaxis (Ryu and Samuel, 2002). Tracks of *C. elegans* in

Materials and Methods

C. elegans that were grown in minimal medium containing nematode growth factor (Brenner, 1974). From each strain (wild-type, cryophilic, or thermophilic) obtained by loss of the AFD and/or AIX neurons), or acticle (obtained by loss of the AFD neuron or multiple neurons). Thus, Mori and Ohshima (1995) proposed that the AFD and AIX neurons comprise the thermophilic drive. A comparison of the two did not explain how rapid the worm’s movement is in a direction selected during isothermal trials. The present study using thermotaxis, its movements, and the mechanisms of worm tracking worms down fluid spatial or temp the swimming motion joined to thermal sites studied thermotaxis a behavioral deficit is identified by Mori thermotaxis. We found strong ev gradients toward the spatial to the cryophilic competing evidence that might contribute for migration down g orientation, absolute temperature, and its changes in tempera modulates run ve

Thermotaxis in *Caenorhabditis elegans* Analyzed by Measuring Responses to Defined Thermal Stimuli

William S. Ryu and Aravinthan D. T. Samuel
Department of Molecular and Cellular Biology, Harvard University, Cambridge, Massachusetts 02138

In a spatial thermal gradient, *Caenorhabditis elegans* migrates toward and then isothermally tracks near its cultivation temperature. A current model for thermotactic behavior involves a thermophilic drive (involving the neurons AFD and AIX) and cryophilic drive (involving the neuron AIZ) that balance at the cultivation temperature. Here, we analyze the movements of individual worms responding to defined thermal gradients. We found evidence for a mechanism for migration down thermal gradients that is active at temperatures above the cultivation temperature, and a mechanism for isothermal tracking that is active near the cultivation temperature. However, we found no evidence for a mechanism for migration on thermal gradients at temperatures below the cultivation temperature that might have supported the model of opposing drives. The mechanisms for migration down gradients and isothermal tracking control the

thermophilic drive. A comparison of the two did not explain how rapid the worm’s movement is in a direction selected during isothermal trials. The present study using thermotaxis, its movements, and the mechanisms of worm tracking worms down fluid spatial or temp the swimming motion joined to thermal sites studied thermotaxis a behavioral deficit is identified by Mori thermotaxis. We found strong ev gradients toward the spatial to the cryophilic competing evidence that might contribute for migration down g orientation, absolute temperature, and its changes in tempera modulates run ve

Cell

Global Brain Dynamics Embed the Motor Command Sequence of *Caenorhabditis elegans*

Saul Katz,^{1,4} Harris S. Kaplan,^{1,4} Tina Schröder,^{1,4} Susanne Storz,^{1,4} Theodore H. Lindsay,^{2,3} Eraniv Yemini,¹ Shawn Lockery,¹ and Manuel Zimmer¹
¹Institute of Neurobiology, University of Osnabrück, Osnabrück, Germany, ²Department of Neurobiology, University of Oregon, Eugene, OR 97403, USA, ³Department of Biochemistry and Molecular Biophysics, Howard Hughes Medical Institute, Columbia University Medical Center, New York, NY 10032, USA, ⁴Cell-NetLab, ⁵Present address: Division of Biology and Biological Engineering, California Institute of Technology, Pasadena, CA 91125, USA

Summary

While isolated motor actions can be correlated with activities of neuronal networks, an unresolved problem is how the brain assembles these activities into organized behaviors like action sequences. Using brain-wide calcium imaging in *Caenorhabditis elegans*, we show that a large proportion of neurons across the brain share information by engaging in coordinated, dynamical network activity. This brain state evolves on a cycle, each segment of which recruits the activities of different neuronal sub-populations and can be explicitly mapped, on a single trial basis, to the animals’ major motor commands. This organization defines the assembly of motor commands into a string of run-and-turn action sequences cycles, including decisions between alternative behaviors. These dynamics serve as a robust scaffold for action selection in response to sensory input. This study shows that the coordination of neuronal activity patterns into global brain dynamics underlies the high-level organization of behavior.

Introduction

Behavior is composed of individual motor actions and motifs, such as limb movements or gait, which do not achieve organized goals unless they are orchestrated into longer-lasting action sequences and behavioral strategies, like navigation, grooming, or courtship (Anderson and Perona, 2004; Gray et al., 2005; Soetedjo et al., 2010). Ethologists often make quantitative descriptions of the high-level organization using state transition diagrams, consisting of distinct, repeatable high-level motor states and switches between them (Anderson and Perona, 2004). The brain’s representation of behavior must account for both detailed metrics of individual actions (e.g., strength and extent of movement or speed of gait), as well as for their higher-level orchestration. Identifying how these aspects of behavior correspond to measurable neural activity is a necessary step toward understanding how the brain encodes and produces

A circuit for navigation in *Caenorhabditis elegans*

Jesse M. Gray¹, Joseph J. Hill, and Cornelia I. Bargmann^{1*}

¹Program in Developmental Biology, Genetics, and Neuroscience, Department of Anatomy and Department of Biochemistry and Biophysics, Howard Hughes Medical Institute, University of California, San Francisco, CA 94143
This contribution is part of the special series of Inaugural Articles by members of the National Academy of Sciences elected on April 29, 2003.

Contributed by Cornelia I. Bargmann, December 9, 2004

Caenorhabditis elegans explores its environment by interrupting its forward movement with occasional turns and reversals. Turns and reversals occur at stable frequencies but irregular intervals, producing probabilistic exploratory behaviors. Here we dissect the sensory neurons, interneurons, and motor behaviors under different conditions. After withdrawal from bacterial food, they initiate a local circuit consisting of reversals and deep omega-shaped UVC olfactory neurons, ASX gustatory neurons, and AIX. Over the following 30 min, the animals’ ASX and omega turns are suppressed by ASI and AIX interneurons. Interneurons and motor units of AIX and AIX encode specific aspects of sensory neurons, interneurons, and also implicated in chemotaxis and thermotaxis. They represent a common substrate for multiple

Key words: navigation; sensory neurons; interneurons; motor behaviors; neural circuit

travels through its environment, its nervous system senses, evaluates them based on experience of the animal, and converts this adaptive movement. For simple behaviors, sometimes communicate directly with motor neurons complex behavioral circuits, several layers integrate sensory information and relay it to a path from sensory input to motor output has a few cases, including circuits for crustacean circuits for rapid escape in fish, flies, and

In the nematode *Caenorhabditis elegans*, the defined by using a complete synaptic wiring diagram in its nervous system (4, 5). Six neurons that detect noxious stimuli synapse with interneurons called forward and backward. The command neurons synapse in turn onto possible for forward and backward locomotion withdrawal from the stimulus. The definition circuit has enabled analysis of its development and modification by experience (6–11). The C. elegans provides an opportunity to define many paths from sensory stimuli to behavior.

The escape circuit, the neuronal control of exploratory behavior is poorly characterized, to its favorable conditions by chemotaxis, acetotaxis. In these sensory behaviors and in the absence of informative sensory moves forward and occasionally changes its tent either by a transient reversal or by turning toward movement. The largest change is noted in a sharp omega turn during which the resembles the Greek letter Ω (12, 13) (Fig. 1). This circuit can also change during naviga-

tion, switching between shallow and deep bends. The neuronal pathways that convert sensory information into specific turning behaviors are incompletely defined. The command neurons are not required for the generation of sinusoidal forward movement (4, 11); the neurons that modulate forward movement to cause curving and omega turns are unknown.

In *C. elegans*, such behaviors have features of a biased random walk (14–16) (J.M.G. and C.I.B., unpublished data), a strategy first described in bacterial chemotaxis (17). During a biased random walk, periods of relatively straight movement (“runs”) are occasionally interrupted by periods of rapid direction change (“tumbles” in bacteria and “piroettes” in *C. elegans*). A piroette is a period marked by reversals or sharp turns (14). In a biased random walk, individual trajectories are not predictable. However, when the environment is improving (e.g., when concentrations of attractant increase), piroettes become less frequent. When the environment is declining, piroettes become more frequent. This strategy biases the direction of travel toward favorable conditions.

We sought to understand the mechanisms by which sensory neurons modulate the frequency of reversals and turns in *C. elegans* and the downstream neuronal circuits that generate specific features of these behaviors. The presence of food affects many aspects of *C. elegans* locomotion (18–21). Here we use a systematic analysis of *C. elegans* neuroanatomy to dissect a circuit that uses sensory cues to modulate turning rates in two kinds of exploratory behavior: local search after animals are removed from food and long-range dispersal after prolonged food deprivation. These exploratory behaviors in the absence of directional cues share many features with a biased random walk. Our results delineate a behavioral circuit for navigation in *C. elegans* from sensory input to motor output.

Materials and Methods

Strains. *C. elegans* strains were maintained and grown according to standard procedures (22). The following strains were used: wild-type strain N2, PRS11 *omx-6(gf)* I V, PRS02 *omx-3(gf)* I V, CB1033 *chr-2(e1033)* X, CB112 *cut-2(e112)* II, GR1321 *ph-1(mg280)* II, QH8 *lin-3(mg150)* X, CX3299 *lin-15(e165a)* X, *kyt5[odr-2::gfp; lin-15(+)]*, CX3300 *lin-15(e165a)* X, *kyt5[odr-2::gfp; lin-15(+)]*, CX6896 *mgli1[unc-3::gfp]* IV, and *akls1[omx-1::gfp]* (23).

Behavioral Assays. Young adult animals were first observed on OP50 food in covered plates for 5 min and then transferred onto a fresh, foodless nematode growth medium plate (22). Observation began 1 min after transfer (24). Reversals of three or more head swings were scored as long reversals. Omega turns were visually identified by the head nearly touching the tail or a

Freely available online through the PNAS open access option.

See accompanying Biography on page 3141.

*Present address: Howard Hughes Medical Institute, The Rockefeller University, 1230 York Avenue, New York, NY 10021.

To whom correspondence should be addressed. E-mail: corrieb@rockefeller.edu.

© 2005 by The National Academy of Sciences of the USA.

March 1, 2005 • vol. 25 | no. 9

www.pnas.org/cgi/doi/10.1073/pnas.0409009101

❖ Finite behavioral states

❖ Behavioral states depending on concentration of food, oxygen, temperature, etc.

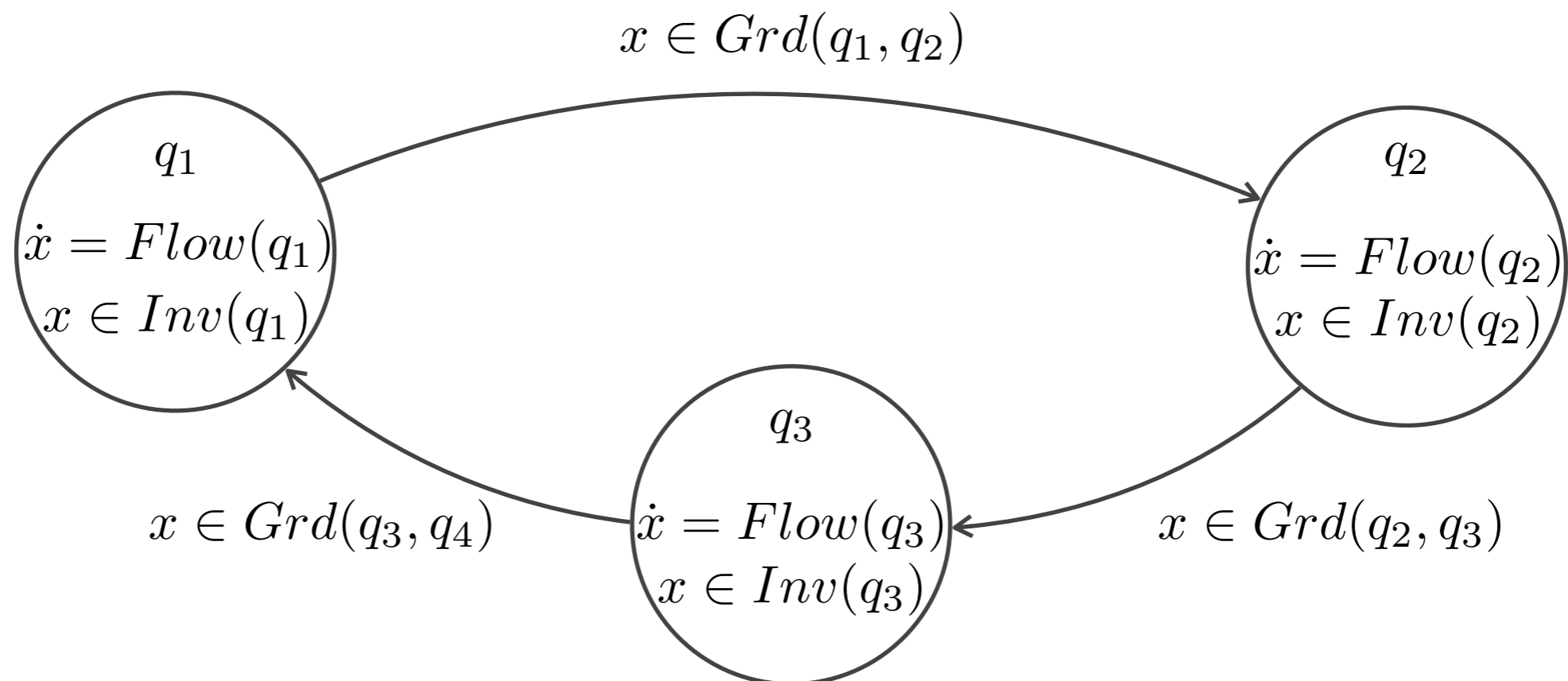
❖ Mixed discrete and continuous behavior

Hybrid Systems capture the mixed continuous and discrete behaviour.

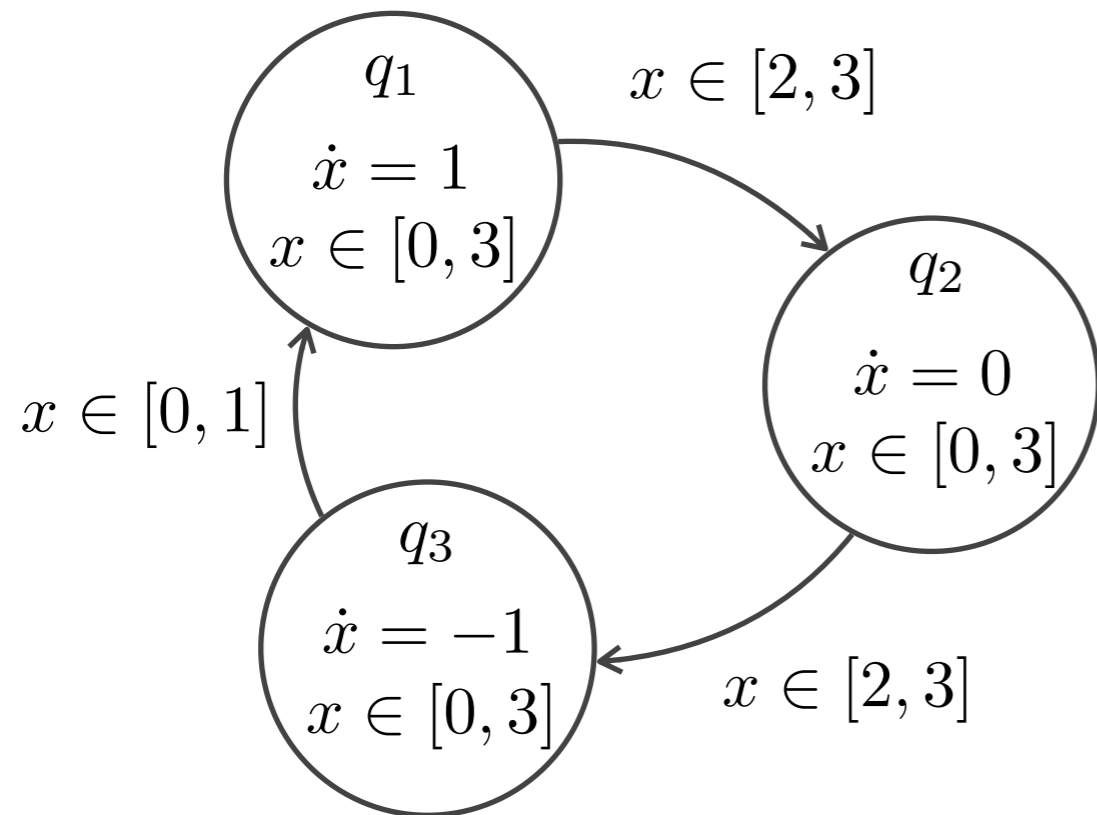
Linear Hybrid Automaton (LHA)

Definition. A **linear hybrid automaton** (LHA) H is a tuple $(Q, E, \mathbb{R}^n, \text{Flow}, \text{Inv}, \text{Grd})$

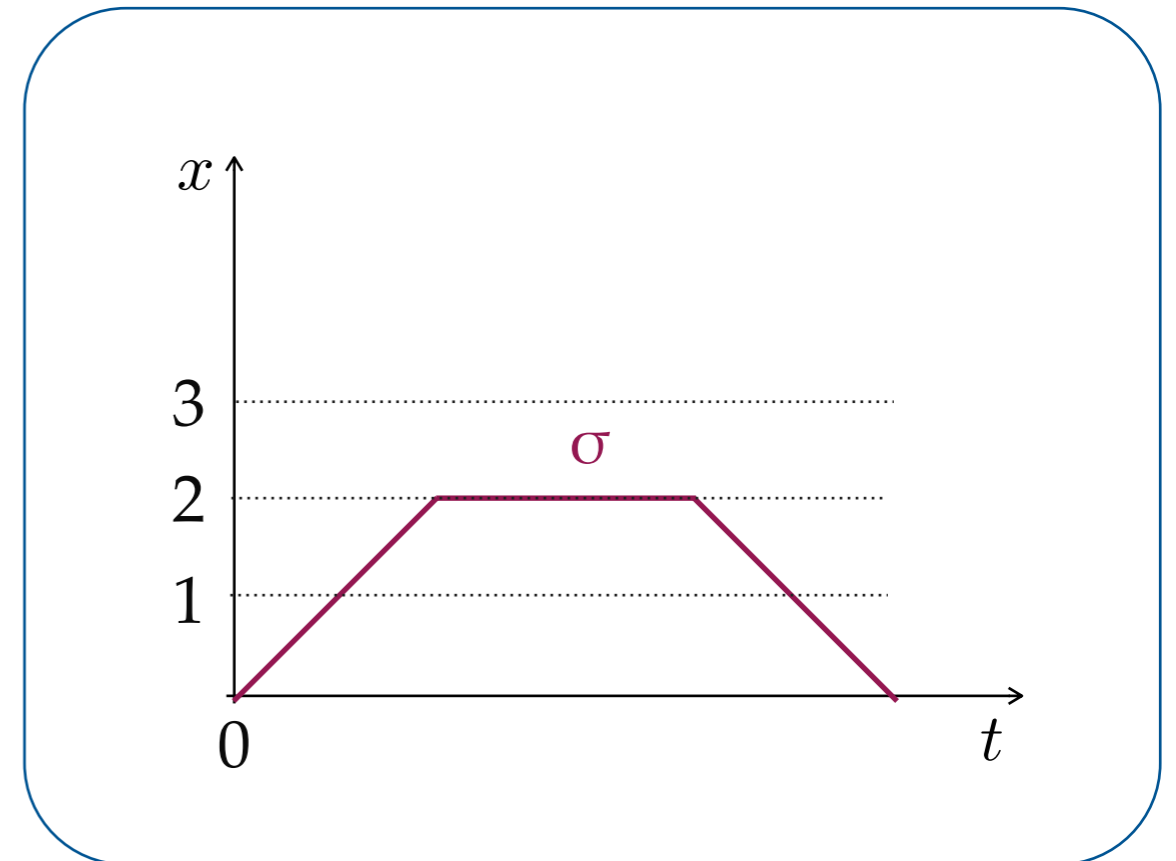
- * Q is a finite set of modes,
- * $E \in Q \times Q$ is a transition relation,
- * $\text{Flow}: Q \rightarrow \mathbb{R}^n$ is the flow function,
- * $\text{Inv}: Q \rightarrow \text{cpoly}(n)$ is the invariant function, and
- * $\text{Grd}: E \rightarrow \text{cpoly}(n)$ is the guard function, where $\text{cpoly}(n)$ is the set of compact and convex polyhedral sets over \mathbb{R}^n .



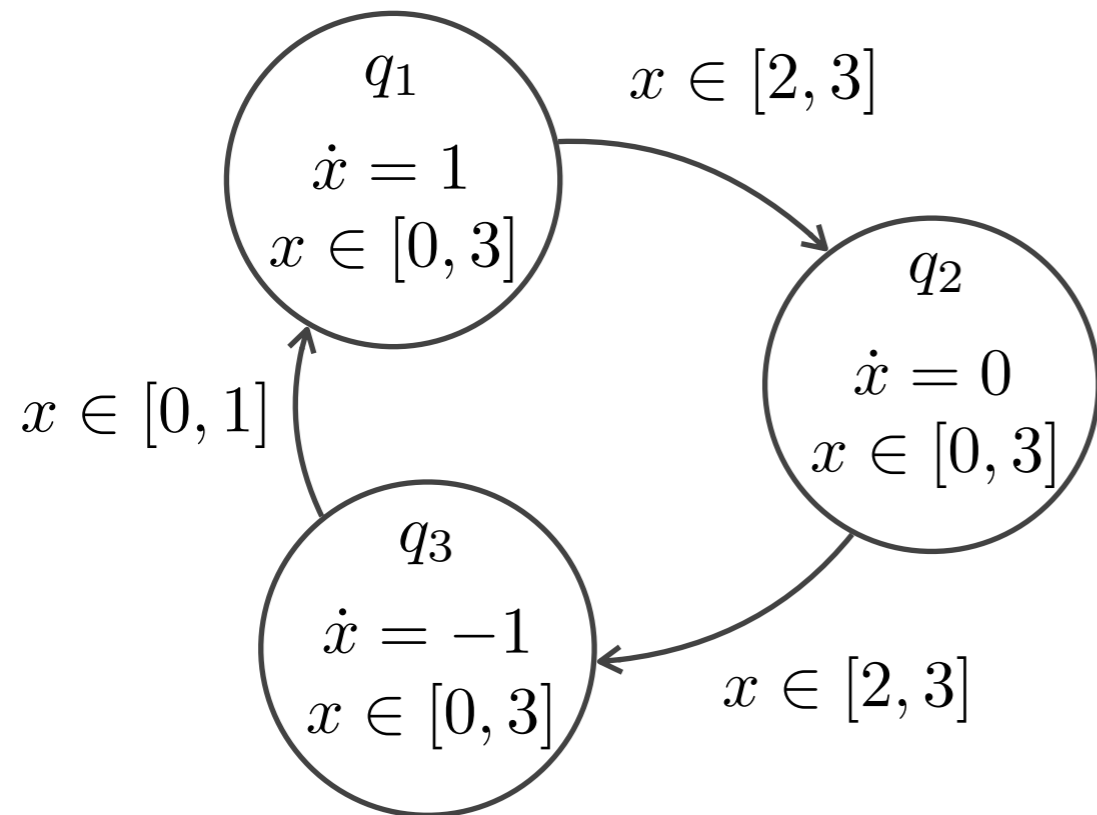
LHA sample



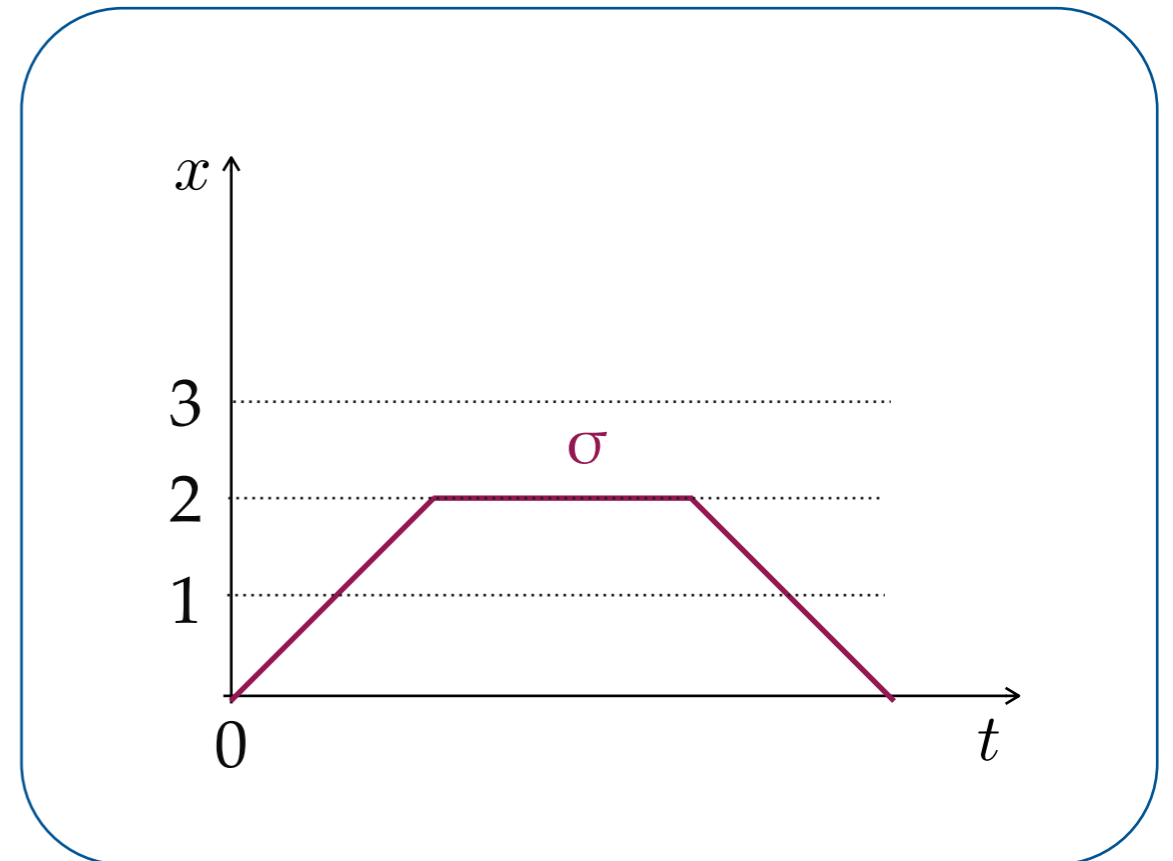
Execution



LHA sample

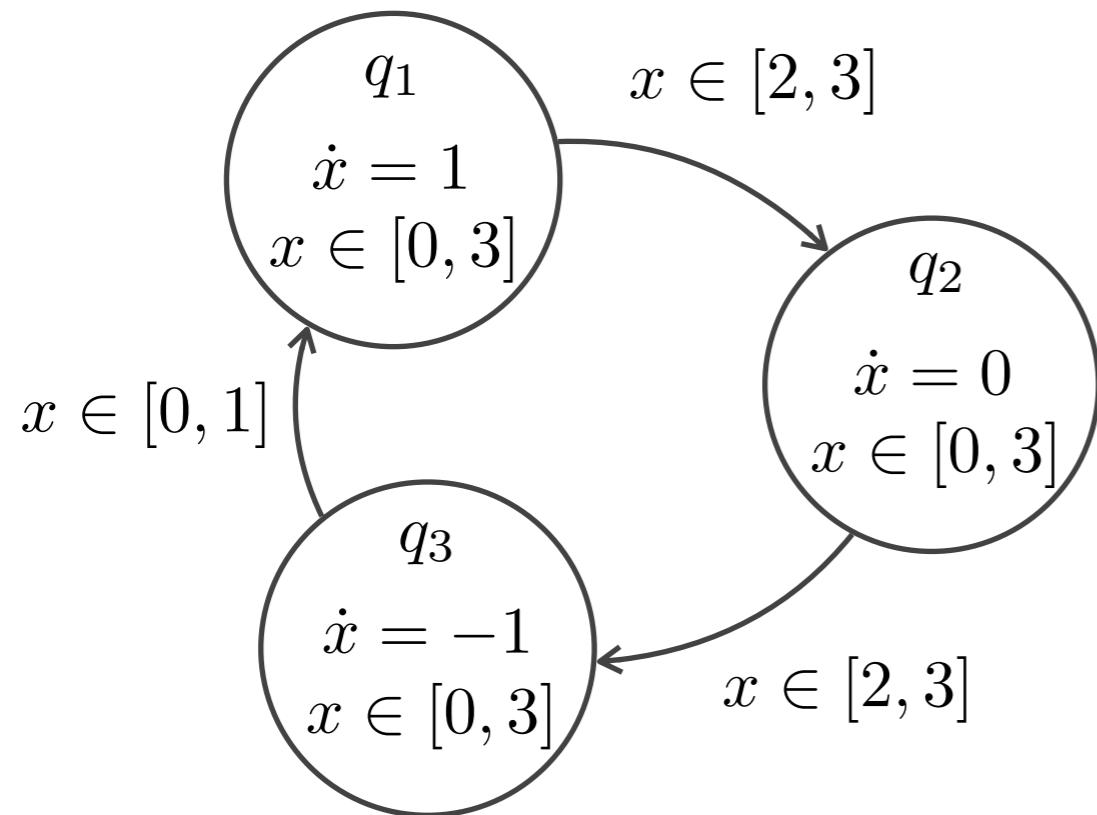


Execution

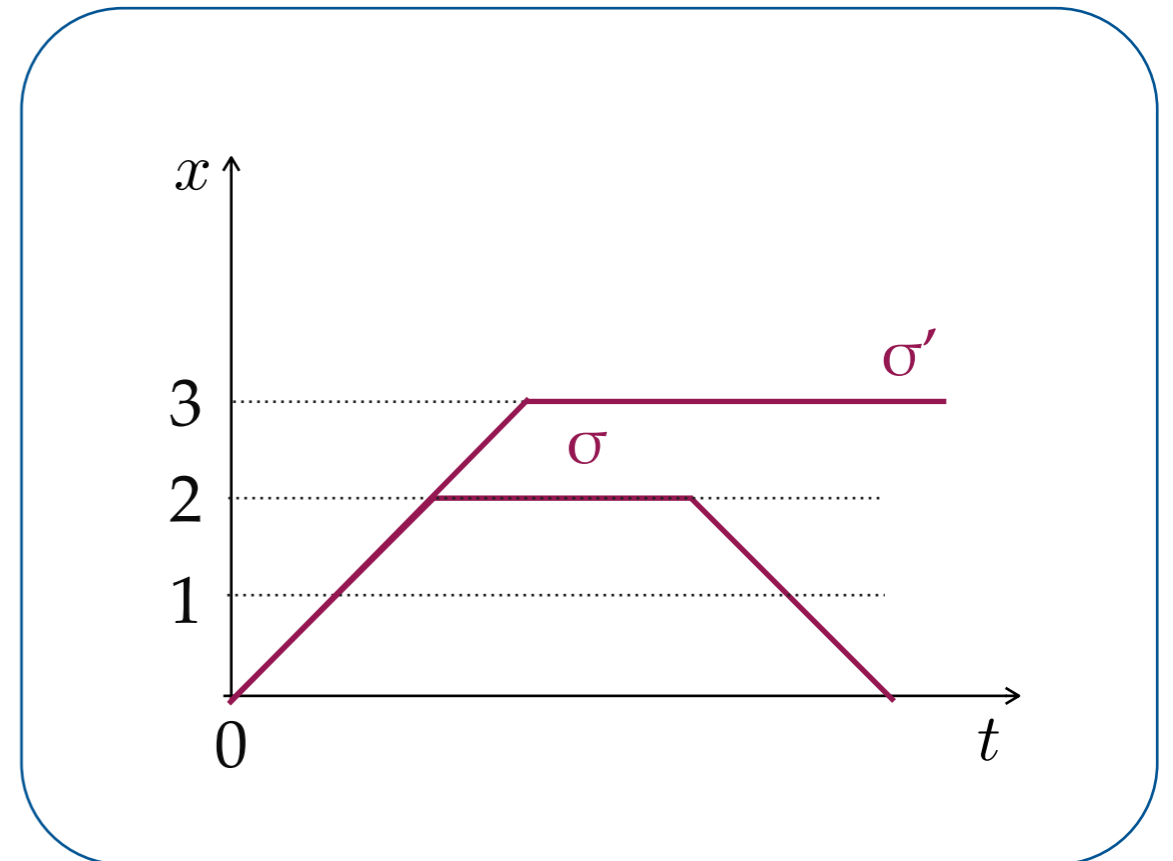


- * We consider non deterministic linear hybrid automata
- * The LHA features piecewise-linear executions

LHA sample



Execution

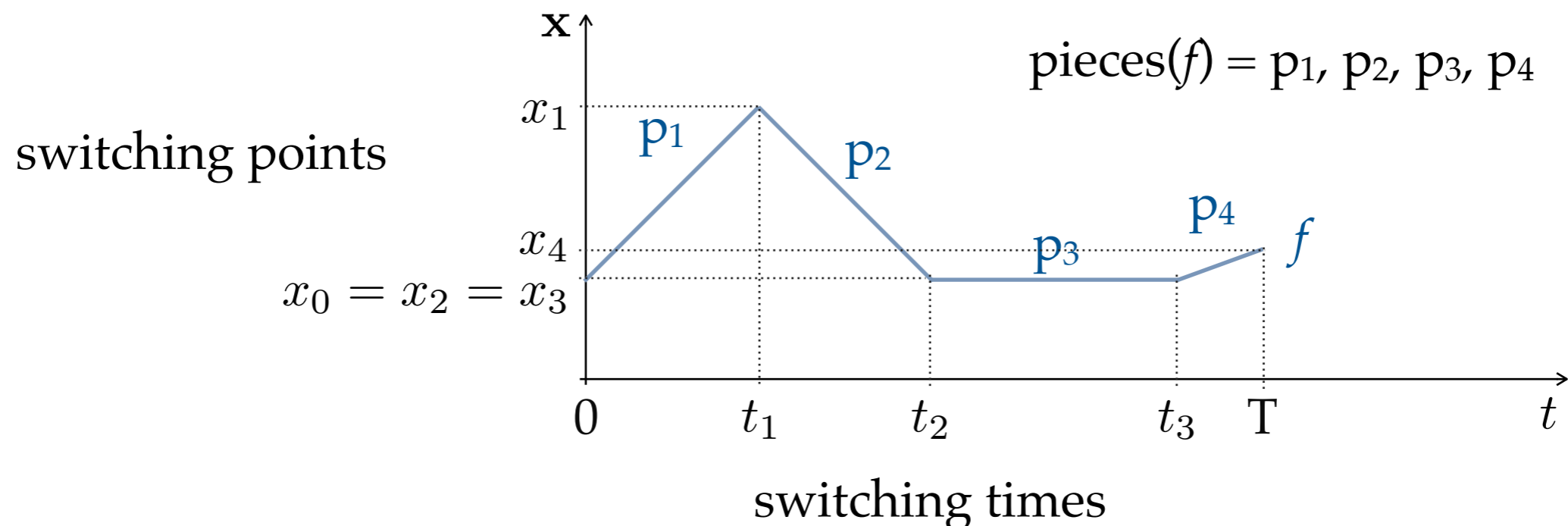


- * We consider non deterministic linear hybrid automata
- * The LHA features piecewise-linear executions

Piecewise-linear function

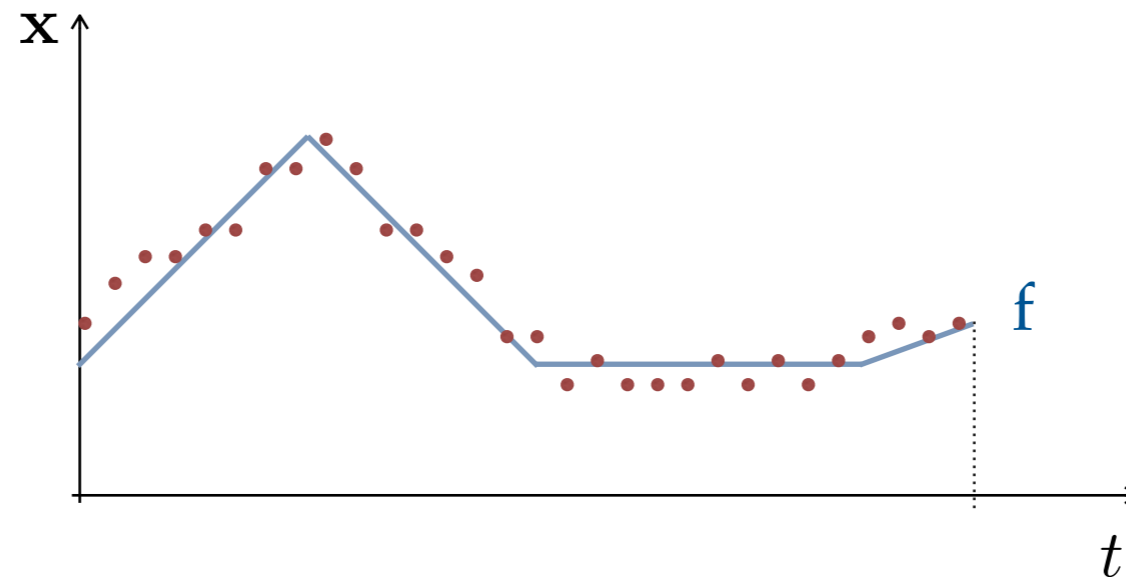
Definition. $f: [0, T] \longrightarrow \mathbb{R}^n$ is an **m-piecewise-linear (m-PWL) function** if

- * $f \equiv p_1, p_2, \dots, p_m$ sequence of m affine pieces of the form $p_i(t) = a_i t + b_i$, where:
 - * a_i is the slope(p_i) and b is the initial value
 - * $f(t) = p_i(t)$ for $t \in \text{dom}(p_i)$
 - * f is continuous



Data: Piecewise-linear approximation

Time series over-approximation

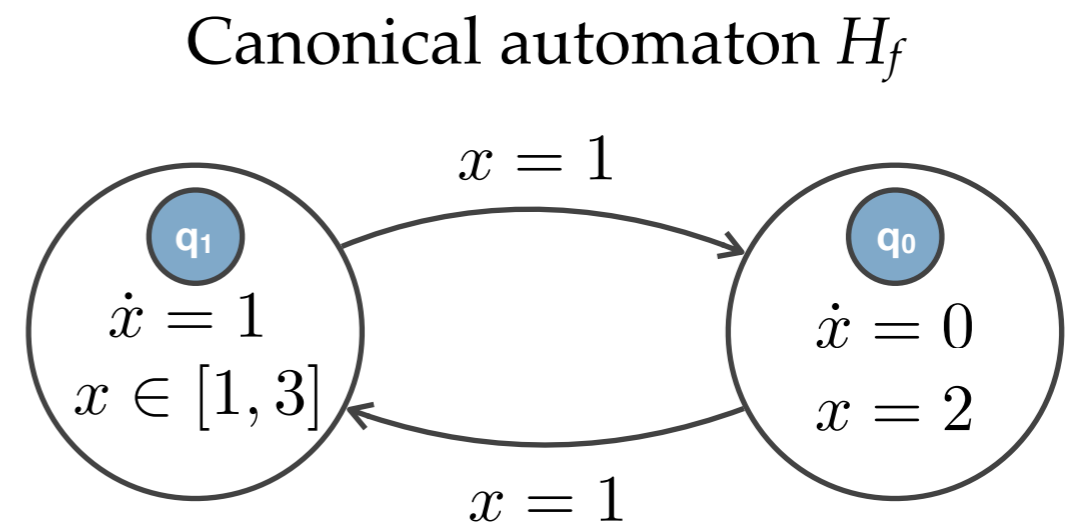
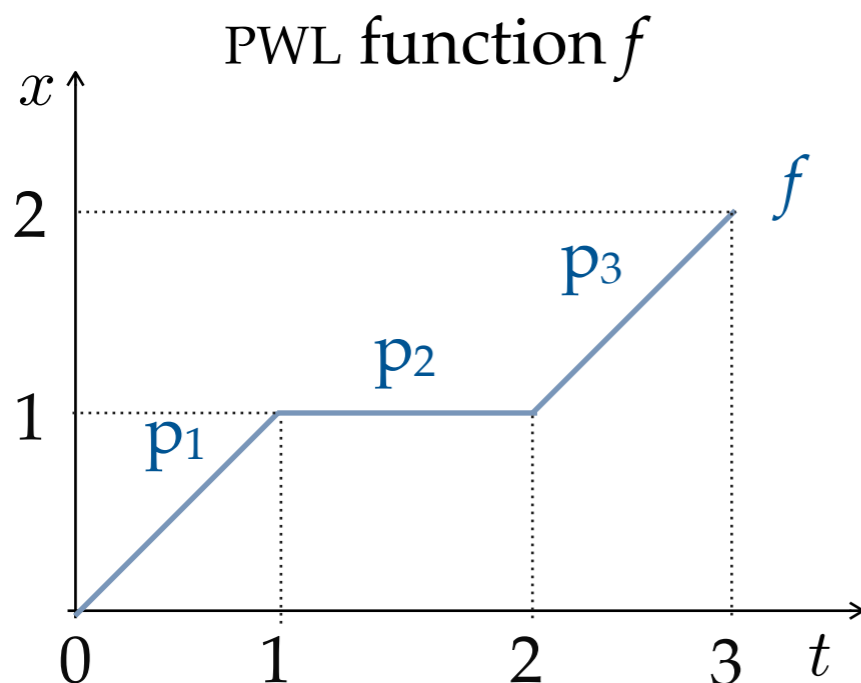


- * Douglas-Peucker line simplification algorithm
- * Linear regression
- * Hakimi-Schmeichel algorithm

Canonical linear hybrid automaton

Definition. Let f be an n -dimensional PWL function. Then, the **canonical automaton of f** is defined as $H_f = (Q, E, \mathbb{R}^n, \text{Flow}, \text{Inv}, \text{Grd})$ with

- * $Q = \{q_a : a \in \mathbb{R}^n, \exists p \in \text{pieces}(f) \text{ with } \text{slope}(p) = a\}$,
- * $E = \{ (q_a, q_{a'}) \in Q \times Q : \exists p, p' \in \text{pieces}(f) \text{ adjacent, } \text{slope}(p) = a, \text{slope}(p') = a' \}$
- * $\text{Flow}(q_a) = a$,
- * $\text{Inv}(q_a) = \text{convex_hull}(\{\text{img}(p) : p \in \text{pieces}(f), \text{slope}(p) = a\})$, and
- * $\text{Grd}(q_a, q_{a'}) = \text{convex_hull}(\{\text{end_point}(p) : \exists p, p' \in \text{pieces}(f) \text{ adjacent, } \text{slope}(p) = a, \text{slope}(p') = a'\})$

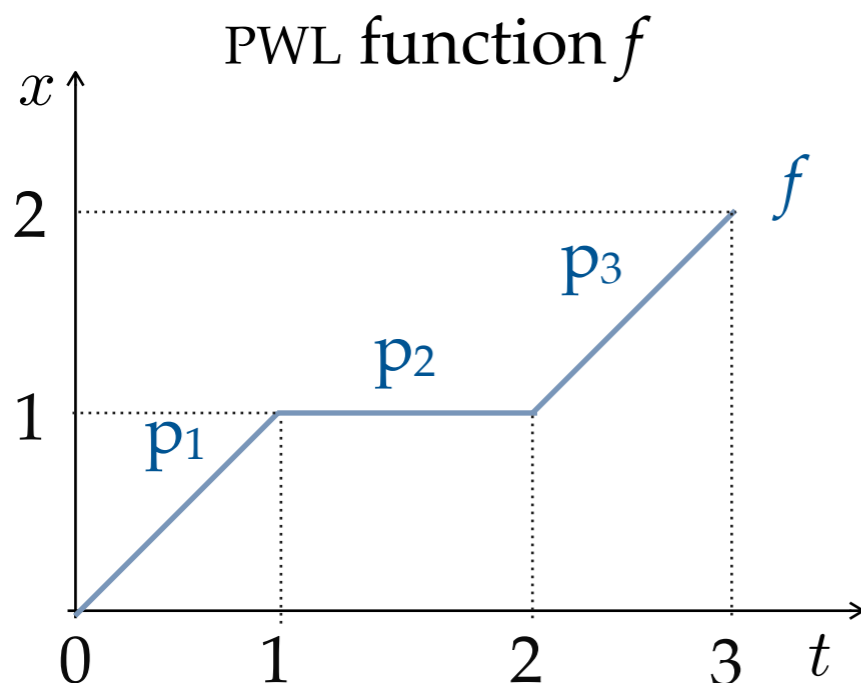


$$\text{slope}(p_1) = 1, \text{slope}(p_2) = 0, \text{slope}(p_3) = 1$$

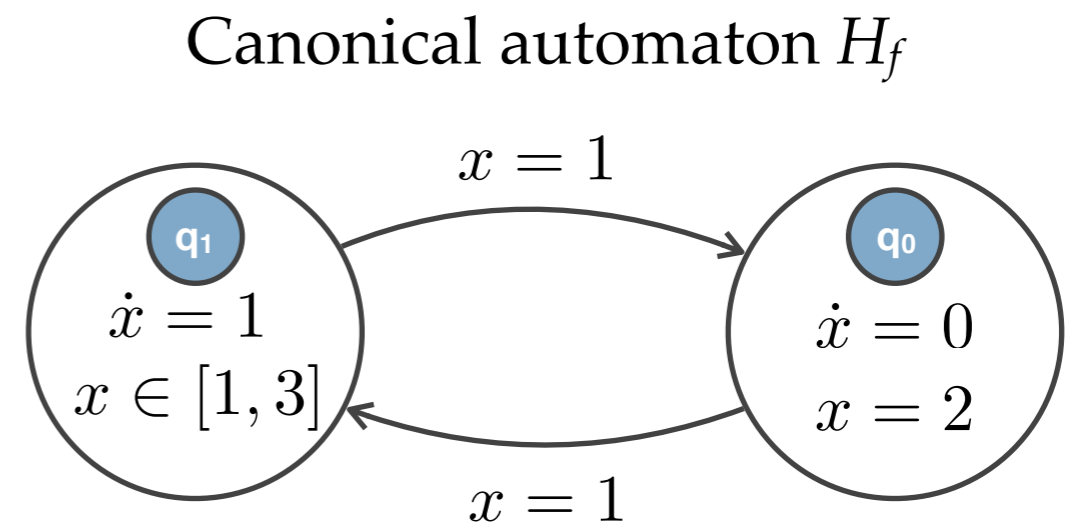
Canonical linear hybrid automaton

Definition. Let f be an n -dimensional PWL function. Then, the **canonical automaton of f** is defined as $H_f = (Q, E, \mathbb{R}^n, \text{Flow}, \text{Inv}, \text{Grd})$ with

- * $Q = \{q_a : a \in \mathbb{R}^n, \exists p \in \text{pieces}(f) \text{ with } \text{slope}(p) = a\}$,
- * $E = \{ (q_a, q_{a'}) \in Q \times Q : \exists p, p' \in \text{pieces}(f) \text{ adjacent, } \text{slope}(p) = a, \text{slope}(p') = a' \}$
- * $\text{Flow}(q_a) = a$,
- * $\text{Inv}(q_a) = \text{convex_hull}(\{\text{img}(p) : p \in \text{pieces}(f), \text{slope}(p) = a\})$, and
- * $\text{Grd}(q_a, q_{a'}) = \text{convex_hull}(\{\text{end_point}(p) : \exists p, p' \in \text{pieces}(f) \text{ adjacent, } \text{slope}(p) = a, \text{slope}(p') = a'\})$



$$\text{slope}(p_1) = 1, \text{slope}(p_2) = 0, \text{slope}(p_3) = 1$$



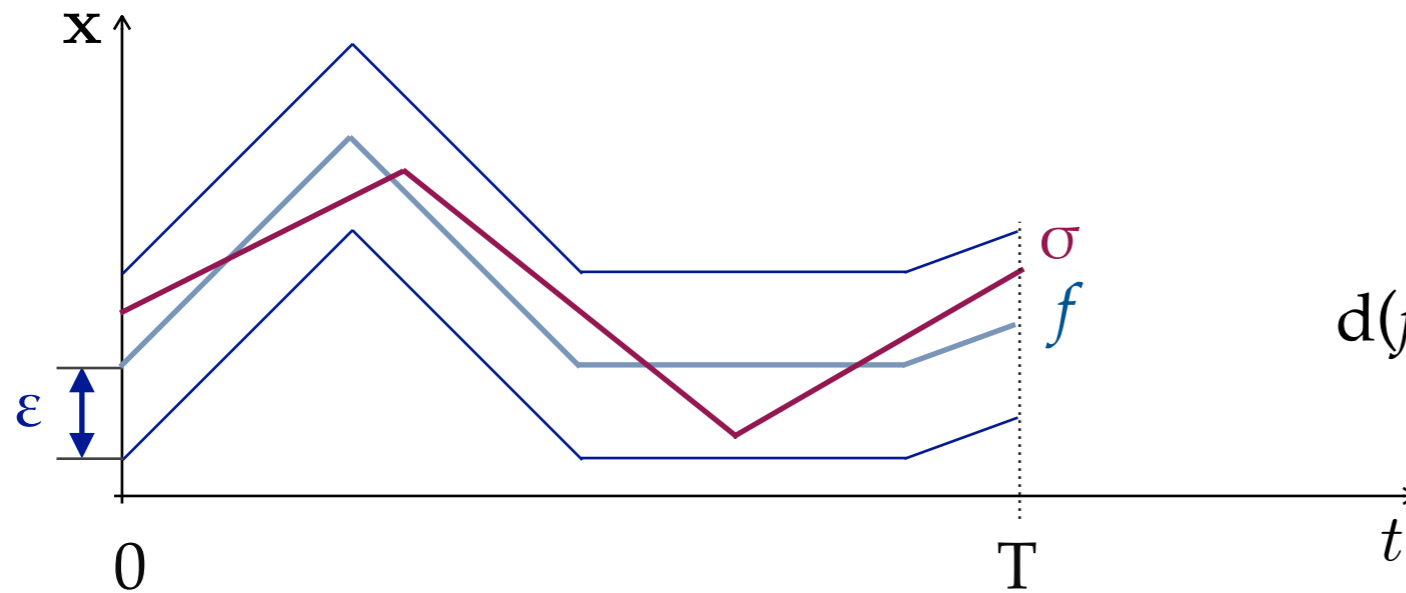
Given a PWL function f ,
 f is an execution of H_f

Synthesis Problem

Synthesis problem

Given a finite set of PWL functions F and a value $\varepsilon \in \mathbb{R}_{\geq 0}$, construct an LHA H that ε -captures every function $f \in F$.

Definition. Given a PWL function f and a value $\varepsilon \in \mathbb{R}_{\geq 0}$, we say that an LHA H ε -captures f if there exists an execution σ in H with $d(f, \sigma) \leq \varepsilon$.



$$d(f, \sigma) = \max_{t \in [0, T]} \|f(t) - \sigma(t)\|$$

ε is a trade-off between the size and the precision of the model.

Specifications

We solve the synthesis problem for two different specifications, in addition to H ε -capturing every function f in F :

Synchronous specification

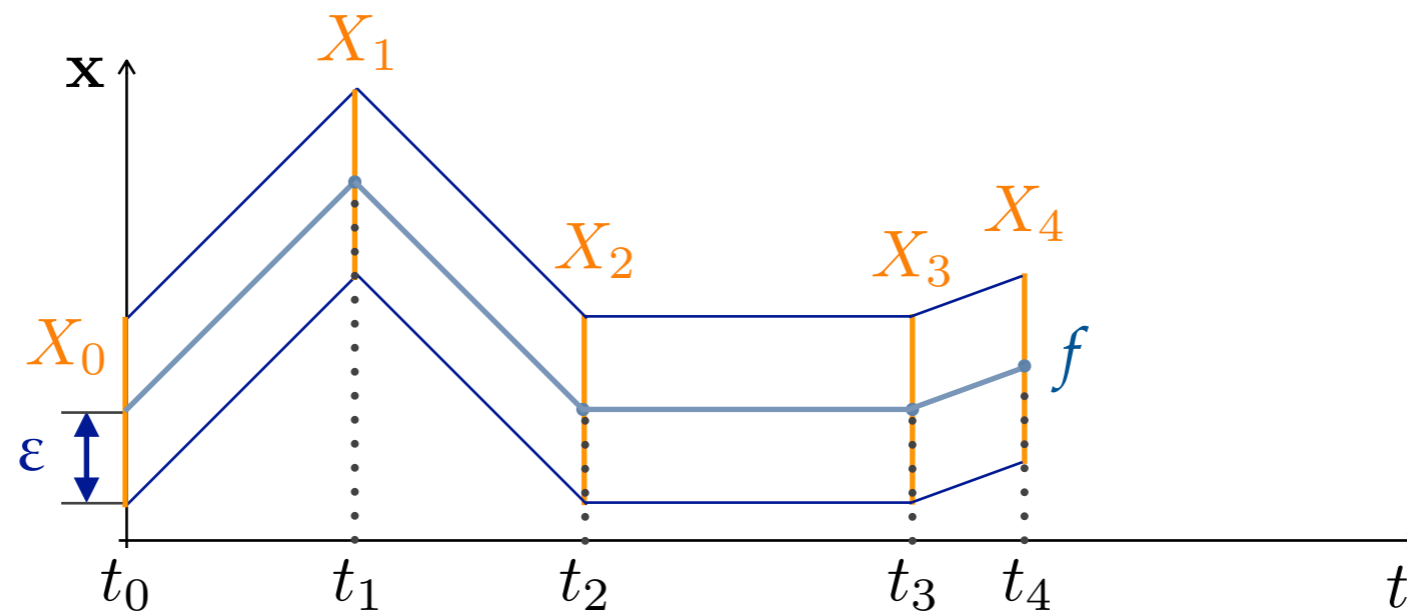
Asynchronous specification

Synchronous specification

Synchronous specification

Specification

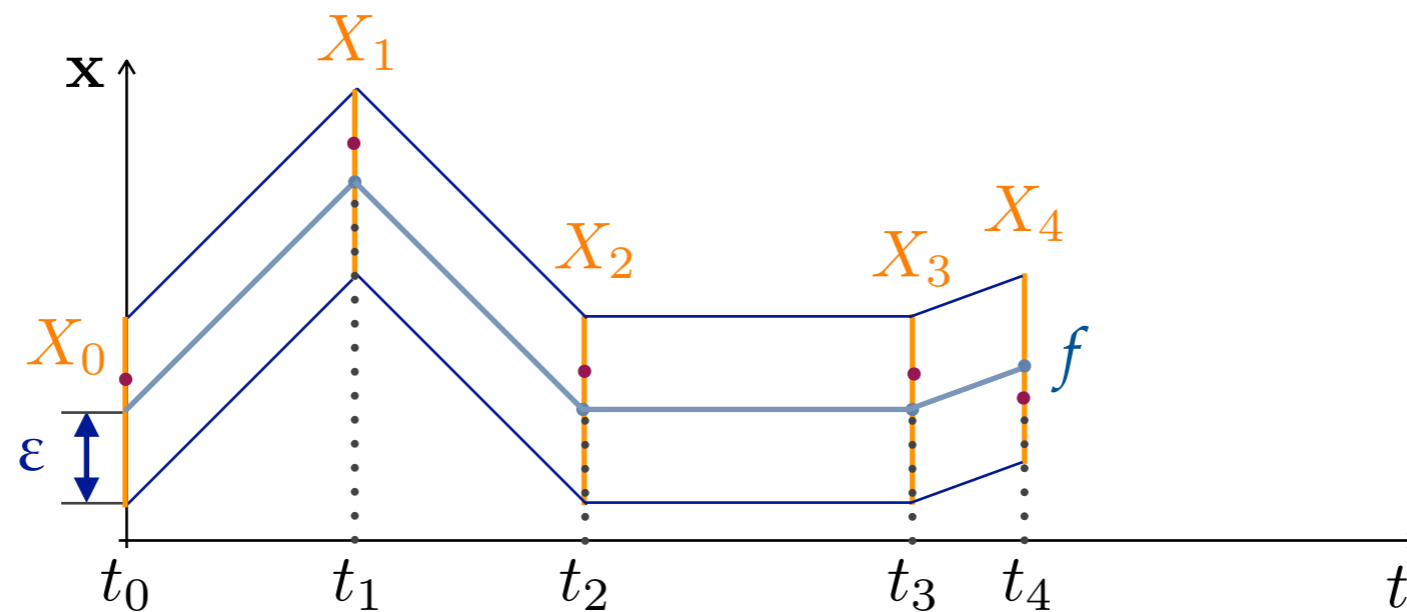
- ❖ H ε -captures every function f in F
- ❖ H switches synchronously with the functions in F



Synchronous specification

Specification

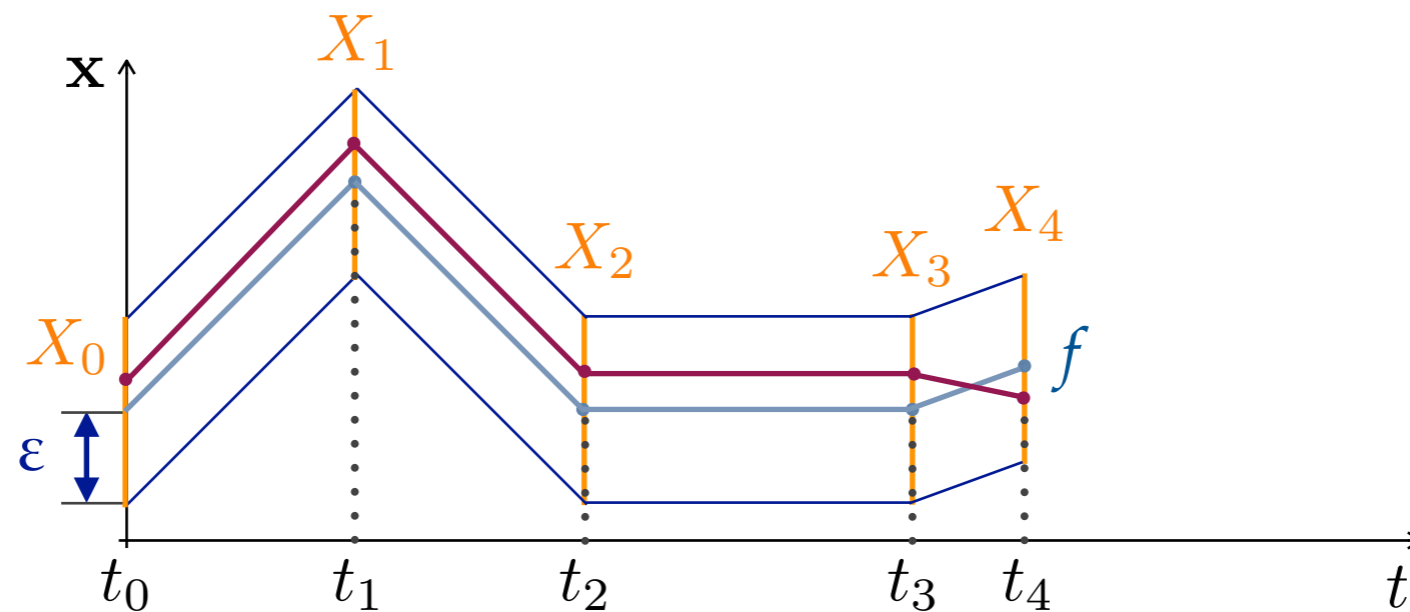
- ❖ H ε -captures every function f in F
- ❖ H switches synchronously with the functions in F



Synchronous specification

Specification

- ❖ H ε -captures every function f in F
- ❖ H switches synchronously with the functions in F

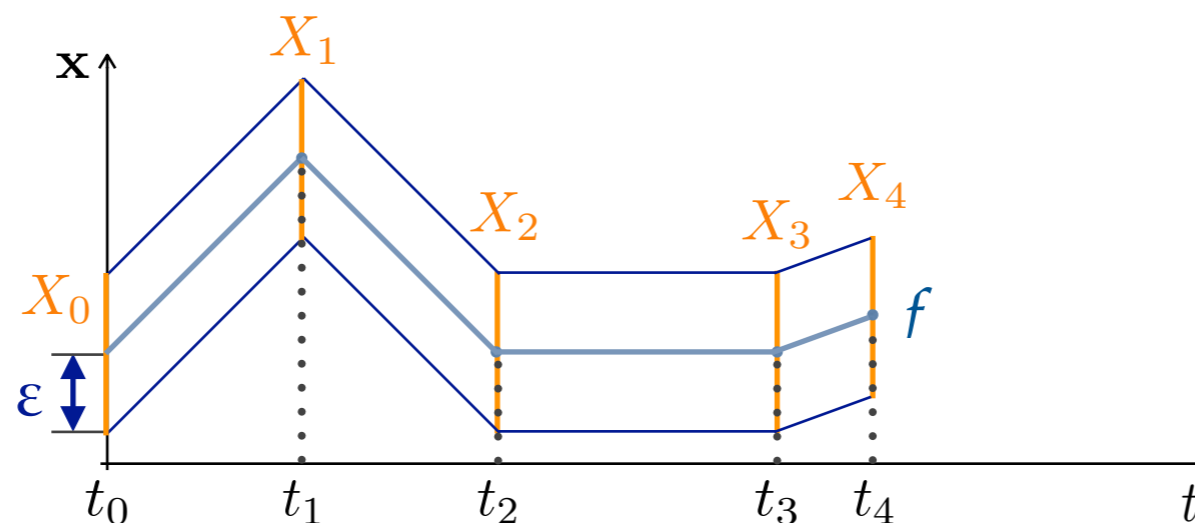


SMT-based synthesis approach

SMT-based synthesis approach

- ❖ For every function f in F , construct a PWL function g satisfying the specification with ℓ different slopes:
 - ❖ $g \equiv q_1, \dots, q_m$
 - ❖ Switching times of g are the same as switching times of $f: t_0, \dots, t_m$
 - ❖ Switching points of g are y_0, \dots, y_m and they are ε -close to switching points of f , $x_0 = f(t_0), \dots, x_m = f(t_m) \longrightarrow$ Expressed as $y_j \in X_j$
 - ❖ $\mathbf{b}_1 = \text{slope}(q_1), \dots, \mathbf{b}_m = \text{slope}(q_m)$ with ℓ different values $(\mathbf{c}_1, \dots, \mathbf{c}_\ell)$

$$\phi_{f,\varepsilon}(\ell) \equiv \bigwedge_{j=1}^m \mathbf{y}_j = \mathbf{y}_{j-1} + \mathbf{b}_j(t_j - t_{j-1}) \wedge \bigwedge_{j=0}^m \mathbf{y}_j \in X_j \wedge \bigwedge_{j=1}^m \bigvee_{k=1}^{\ell} \mathbf{b}_j = \mathbf{c}_k$$

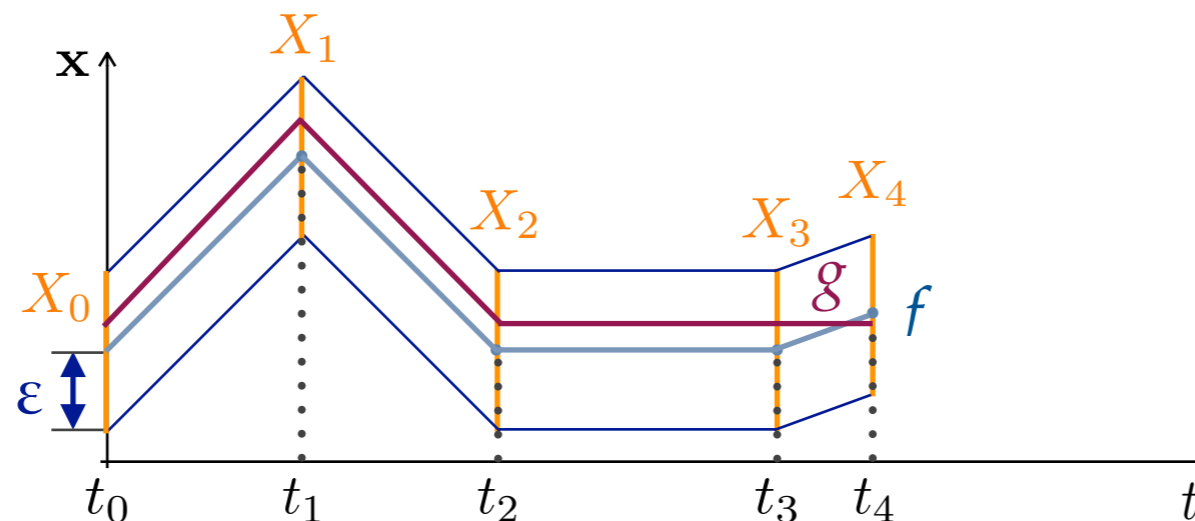


SMT-based synthesis approach

SMT-based synthesis approach

- ❖ For every function f in F , construct a PWL function g satisfying the specification with ℓ different slopes:
 - ❖ $g \equiv q_1, \dots, q_m$
 - ❖ Switching times of g are the same as switching times of $f: t_0, \dots, t_m$
 - ❖ Switching points of g are y_0, \dots, y_m and they are ε -close to switching points of f , $x_0 = f(t_0), \dots, x_m = f(t_m) \longrightarrow$ Expressed as $y_j \in X_j$
 - ❖ $\mathbf{b}_1 = \text{slope}(q_1), \dots, \mathbf{b}_m = \text{slope}(q_m)$ with ℓ different values $(\mathbf{c}_1, \dots, \mathbf{c}_\ell)$

$$\phi_{f,\varepsilon}(\ell) \equiv \bigwedge_{j=1}^m \mathbf{y}_j = \mathbf{y}_{j-1} + \mathbf{b}_j(t_j - t_{j-1}) \wedge \bigwedge_{j=0}^m \mathbf{y}_j \in X_j \wedge \bigwedge_{j=1}^m \bigvee_{k=1}^{\ell} \mathbf{b}_j = \mathbf{c}_k$$



g satisfies $\phi_{f,\varepsilon}(3)$

SMT-based synthesis approach

SMT-based synthesis approach

- ❖ Global linear arithmetic formula for F

$$\phi_{F,\varepsilon}(\ell) \equiv \bigwedge_{f \in F} \phi_{f,\varepsilon}(\ell) \quad \Rightarrow \quad \text{SMT solver}$$

- ❖ Minimization of the number of different slopes

$$\begin{aligned} \min_{\ell \leq M} \phi_{F,\varepsilon}(\ell) \text{ is satisfiable} \\ M = \# \text{ different slopes in } F \end{aligned} \quad \Rightarrow \quad G = \{g_f : f \in F\}$$

- ❖ Canonical automaton of G $\Rightarrow H_G$

SMT-based synthesis approach

SMT-based synthesis approach

- ❖ Global linear arithmetic formula for F

$$\phi_{F,\varepsilon}(\ell) \equiv \bigwedge_{f \in F} \phi_{f,\varepsilon}(\ell) \quad \Rightarrow \quad \text{SMT solver}$$

- ❖ Minimization of the number of different slopes

$$\begin{aligned} \min_{\ell \leq M} \phi_{F,\varepsilon}(\ell) \text{ is satisfiable} \\ M = \# \text{ different slopes in } F \end{aligned} \quad \Rightarrow \quad G = \{g_f : f \in F\}$$

- ❖ Canonical automaton of G

$$\Rightarrow H_G$$

Theorem - SMT-based synthesis

Given a finite set of PWL functions F and a value $\varepsilon \geq 0$, the LHA H_G solves the synthesis problem with **minimal number of modes**.

SMT-based synthesis approach

- ❖ **SMT-based synthesis approach** provides an **optimal global solution**
- ❖ Works well with **short** and **low-dimensional** input PWL functions
- ❖ Does **not scale** to realistic problem sizes

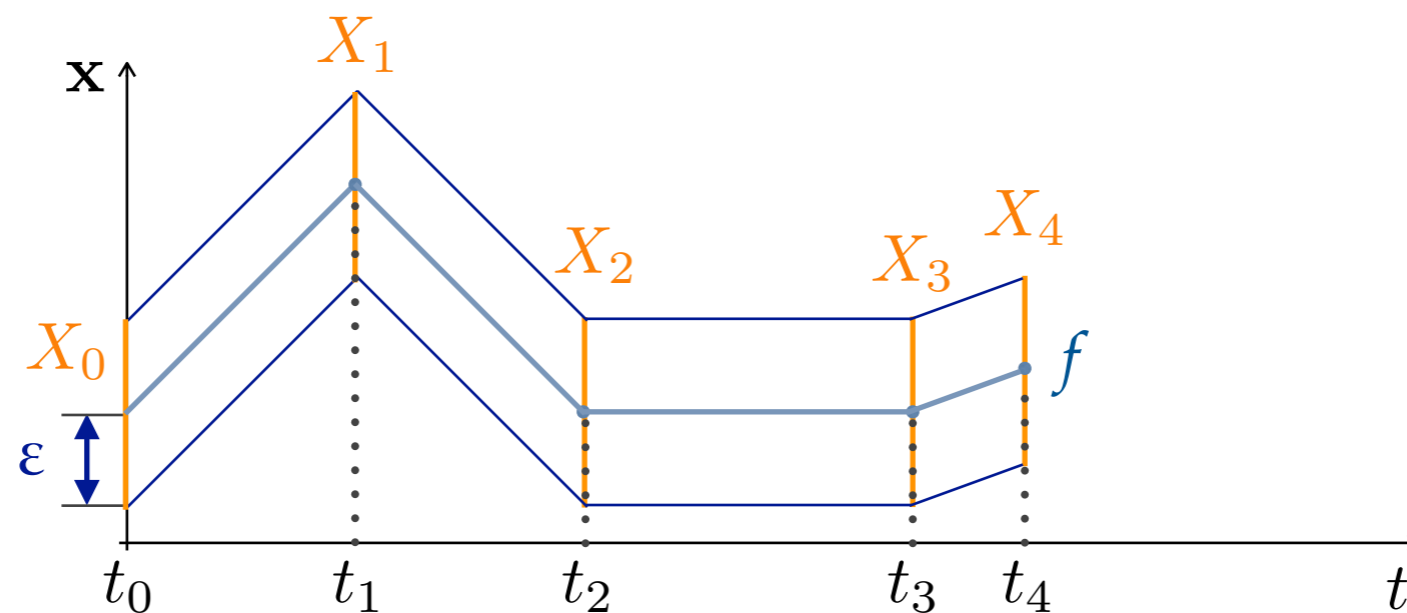
Alternate approach
Membership-based synthesis approach

Asynchronous specification

Asynchronous specification

Specification

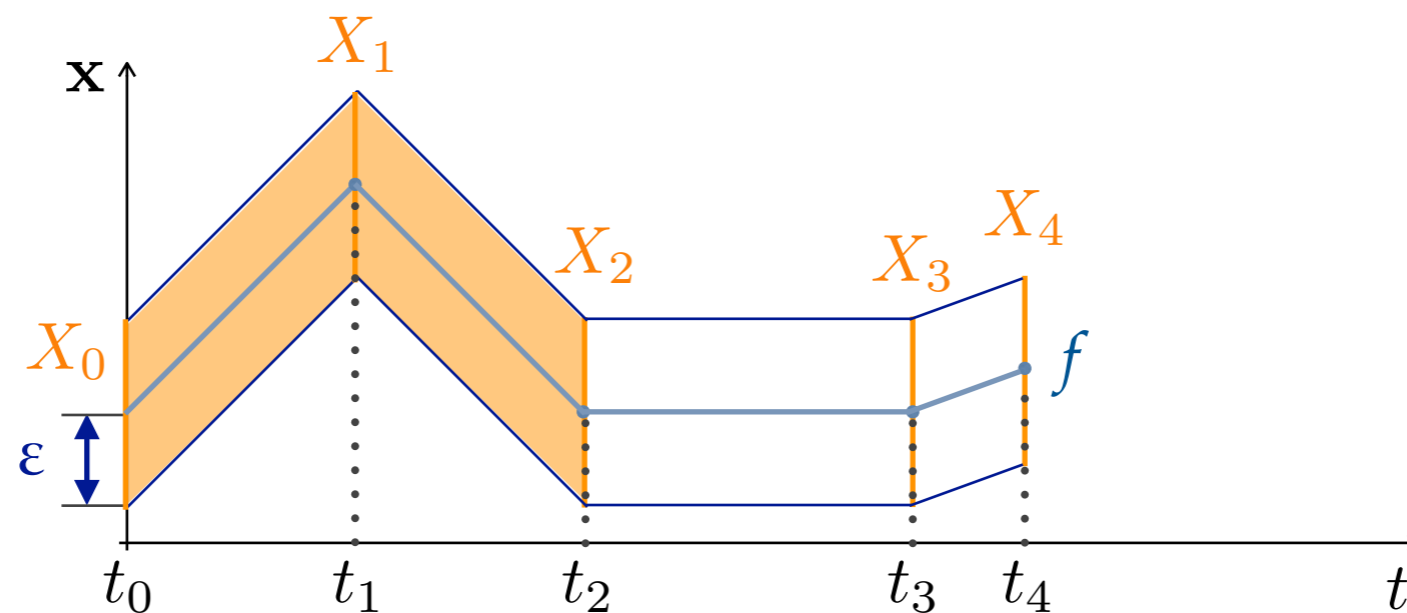
- * H ε -captures every function f in F
- * H switches in the time interval determined by the previous and posterior time instants in the functions f in F



Asynchronous specification

Specification

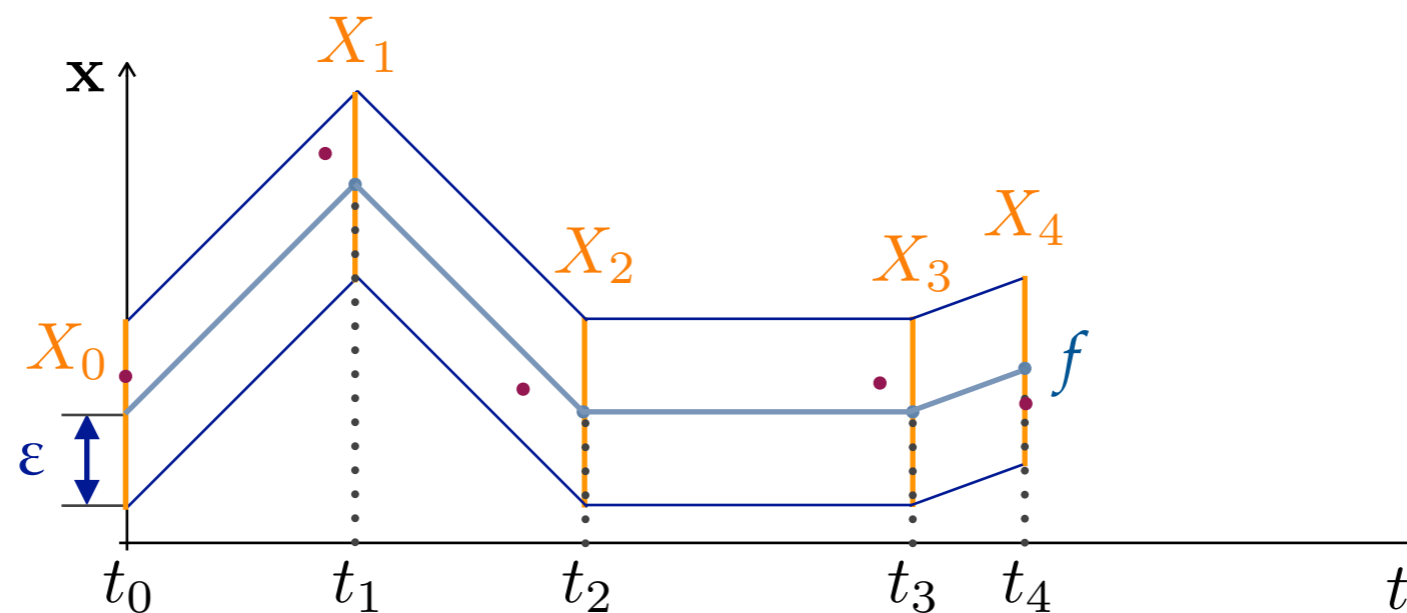
- * H ε -captures every function f in F
- * H switches in the time interval determined by the previous and posterior time instants in the functions f in F



Asynchronous specification

Specification

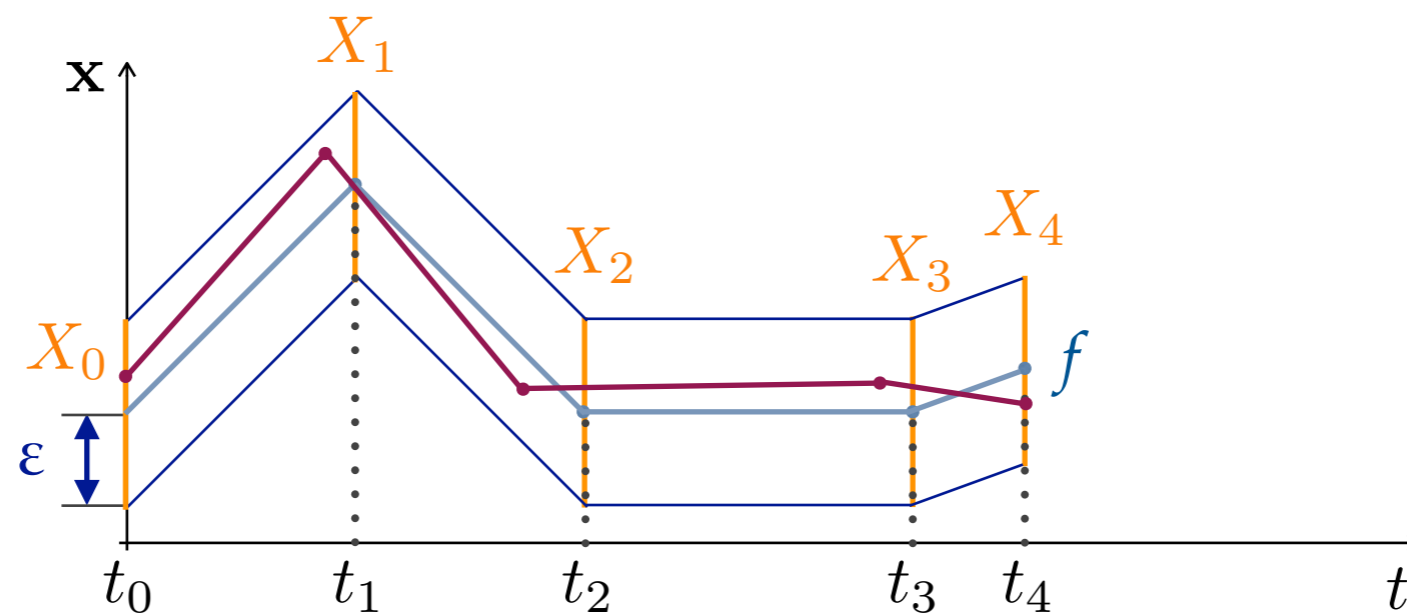
- * H ε -captures every function f in F
- * H switches in the time interval determined by the previous and posterior time instants in the functions f in F



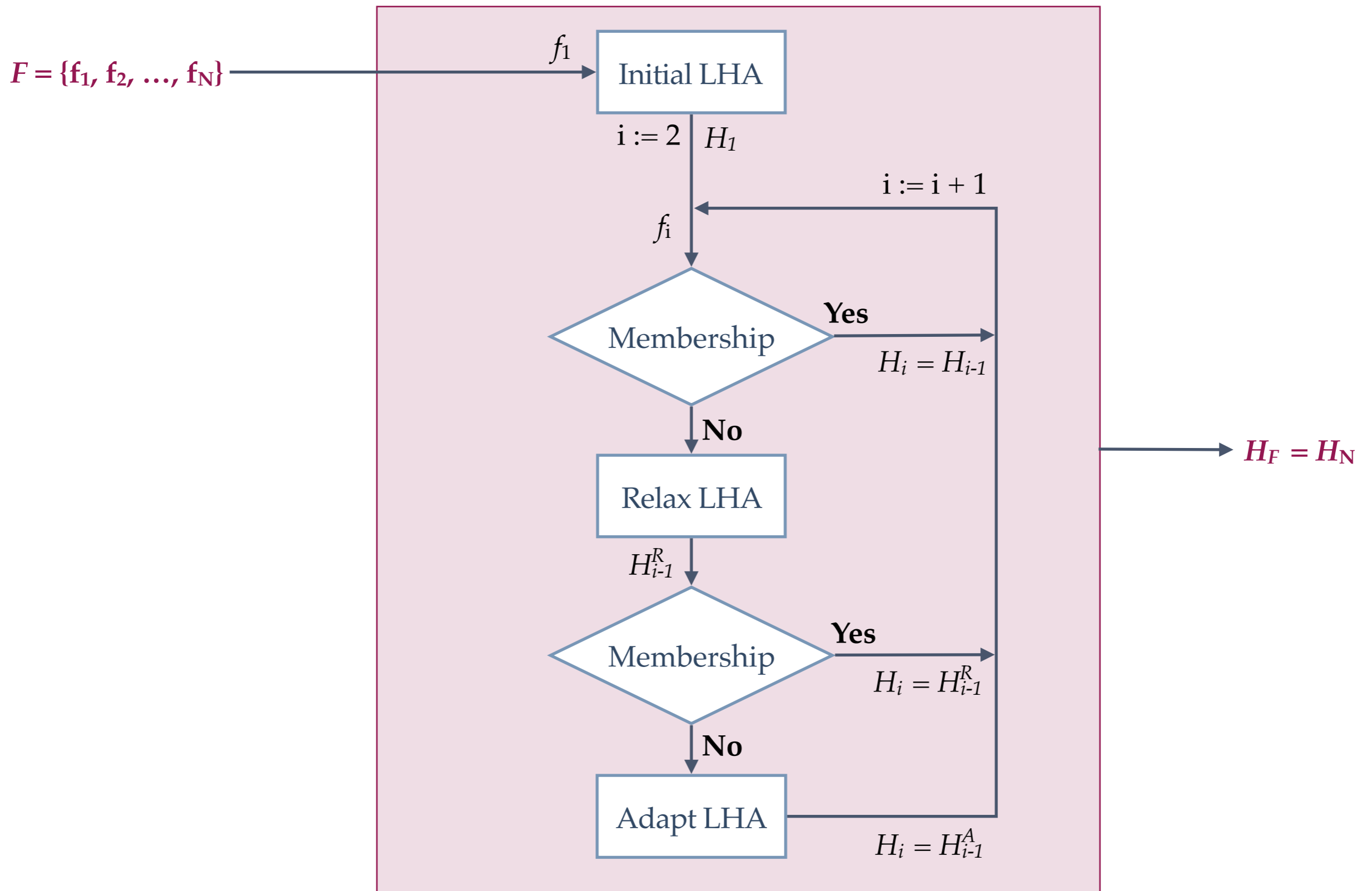
Asynchronous specification

Specification

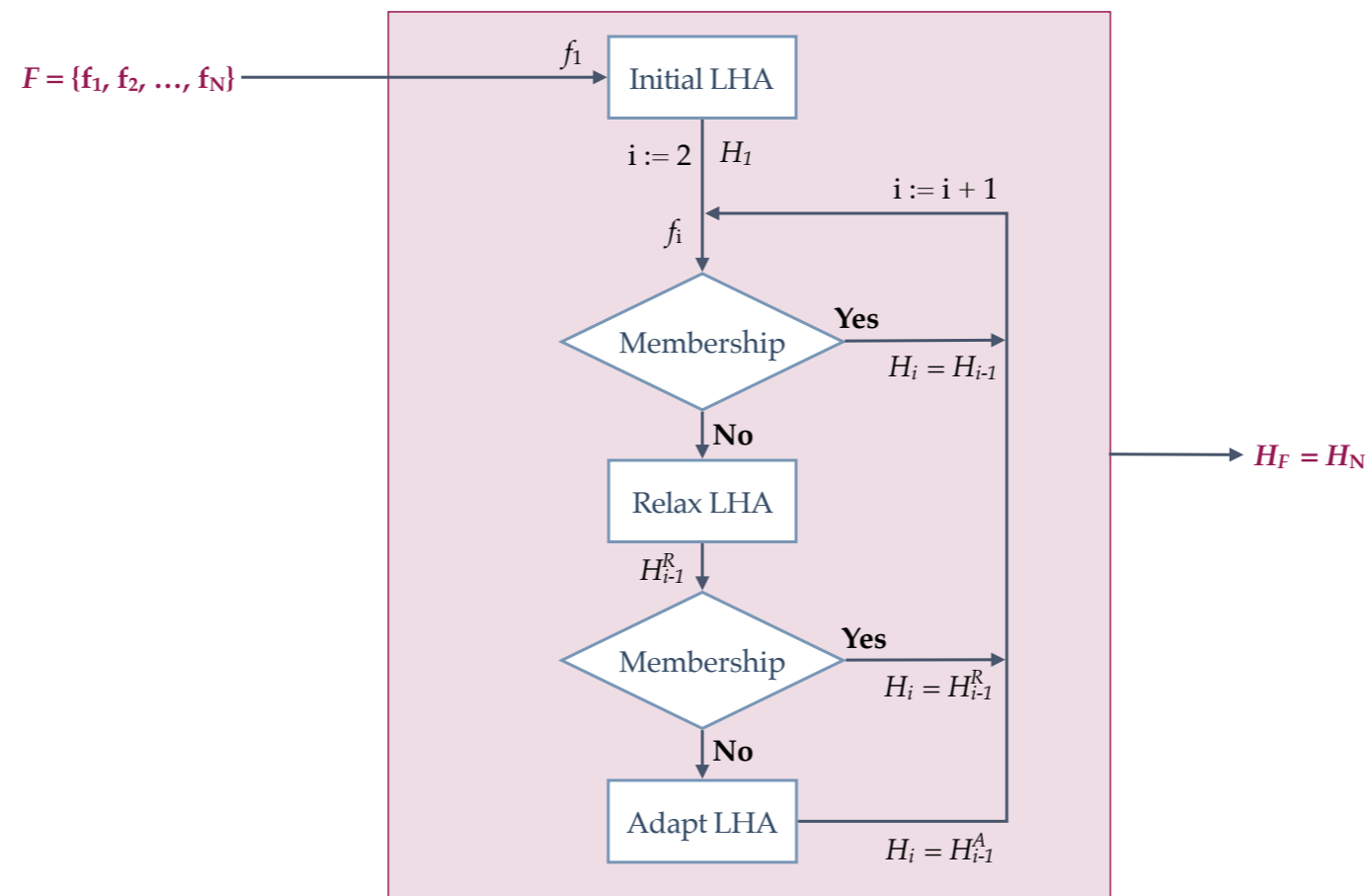
- * H ε -captures every function f in F
- * H switches in the time interval determined by the previous and posterior time instants in the functions f in F



Membership-based synthesis approach



Membership-based synthesis approach



- ❖ **Systematically iterates** over new data
- ❖ A **positive initial membership** result requires **no modification** of the LHA
- ❖ **Returns a counterexample** when the **membership** in the relaxed LHA **fails**
- ❖ The **counterexample** is used to not only increment the guards and invariants of the LHA but to **add** discrete modes

Membership

Membership problem

Given an LHA H and a PWL function f , decide if there exists an execution σ in H consistent with f and such that $d(f, \sigma) \leq \varepsilon$

Definition. An execution σ of H is **consistent with** an m -PWL function $f \equiv p_1, \dots, p_m$ if σ has m affine pieces and its switching times t_1, \dots, t_{m-1} are such that $t_i \in \text{domain}(p_i \cup p_{i+1})$

Membership problem

Given an LHA H and a PWL function f , decide if there exists an execution σ in H consistent with f and such that $d(f, \sigma) \leq \varepsilon$

Definition. An execution σ of H is **consistent with** an m -PWL function $f \equiv p_1, \dots, p_m$ if σ has m affine pieces and its switching times t_1, \dots, t_{m-1} are such that $t_i \in \text{domain}(p_i \cup p_{i+1})$

Membership procedure

- * Construct the ε -tube of f
- * Search for all mode **paths** π of length m in H
- * Search for **executions** determined by π **satisfying** the problem **constraints**

Membership problem

Given an LHA H and a PWL function f , decide if there exists an execution σ in H consistent with f and such that $d(f, \sigma) \leq \varepsilon$

Definition. An execution σ of H is **consistent with** an m -PWL function $f \equiv p_1, \dots, p_m$ if σ has m affine pieces and its switching times t_1, \dots, t_{m-1} are such that $t_i \in \text{domain}(p_i \cup p_{i+1})$

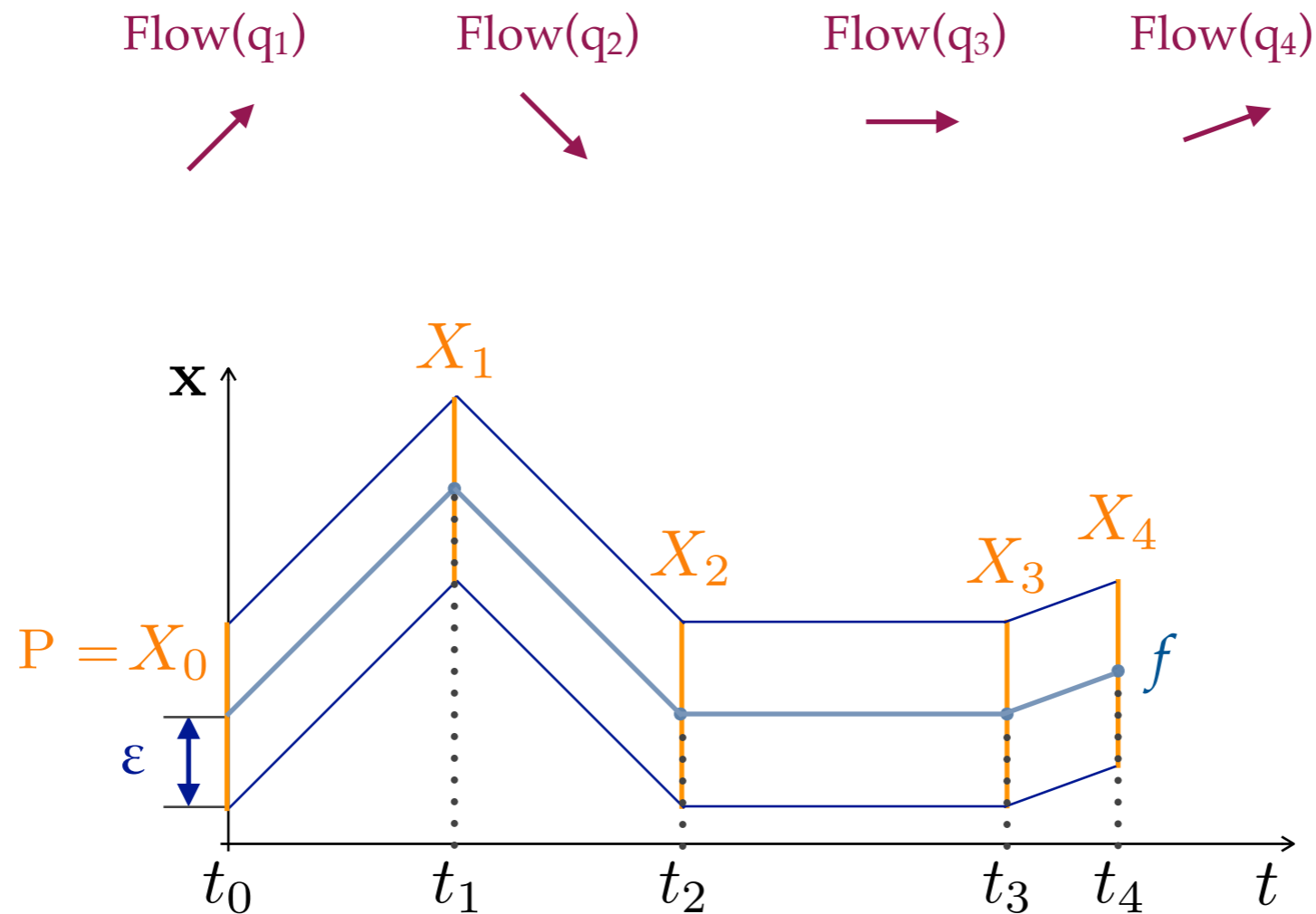
Membership procedure

- * Construct the ε -tube of f
- * Search for all mode **paths** π of length m in H
- * Search for **executions** determined by π **satisfying** the problem **constraints**

Analysis core:
reachability computation

Reachability computation

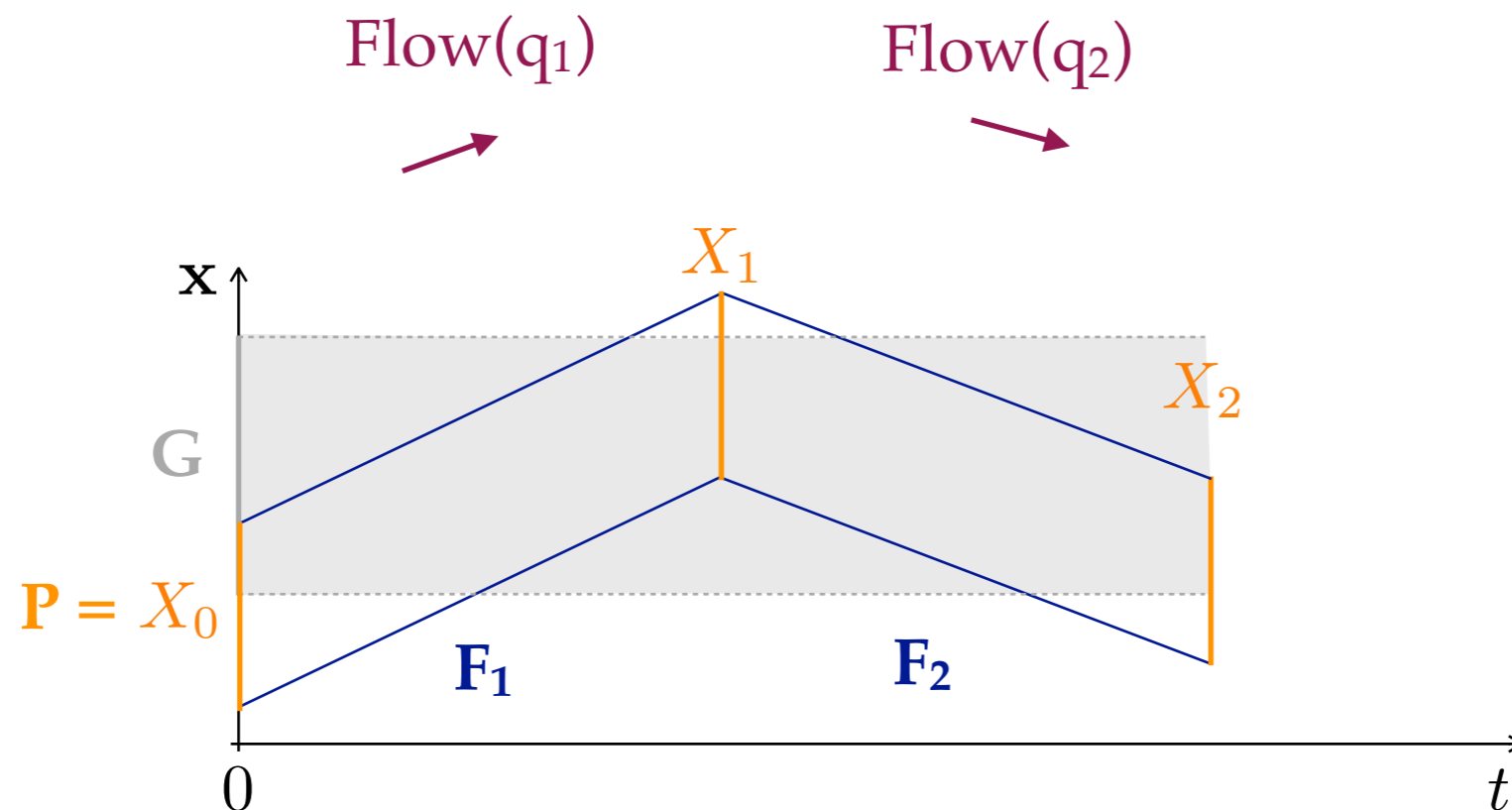
- ❖ Linear hybrid automaton H and PWL function f
- ❖ $\pi \equiv q_1, q_2, q_3, q_4$ mode path in H



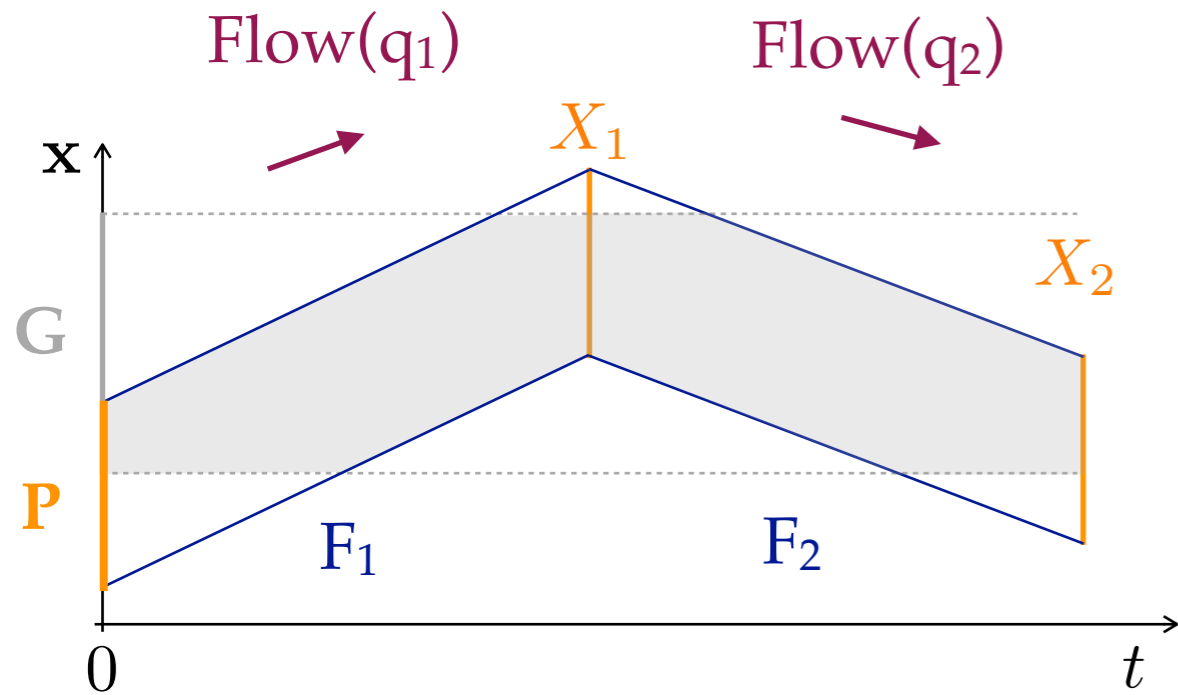
Reachability computation

Basic computation: reachable switching set

- ❖ Consider two ε -tube pieces F_1 and F_2
- ❖ Consider two modes, q_1 and q_2 , their flows, $\text{Flow}(q_1)$ and $\text{Flow}(q_2)$, and the guard from q_1 to q_2 , G .
- ❖ Consider an initial polyhedral set P



Reachability computation

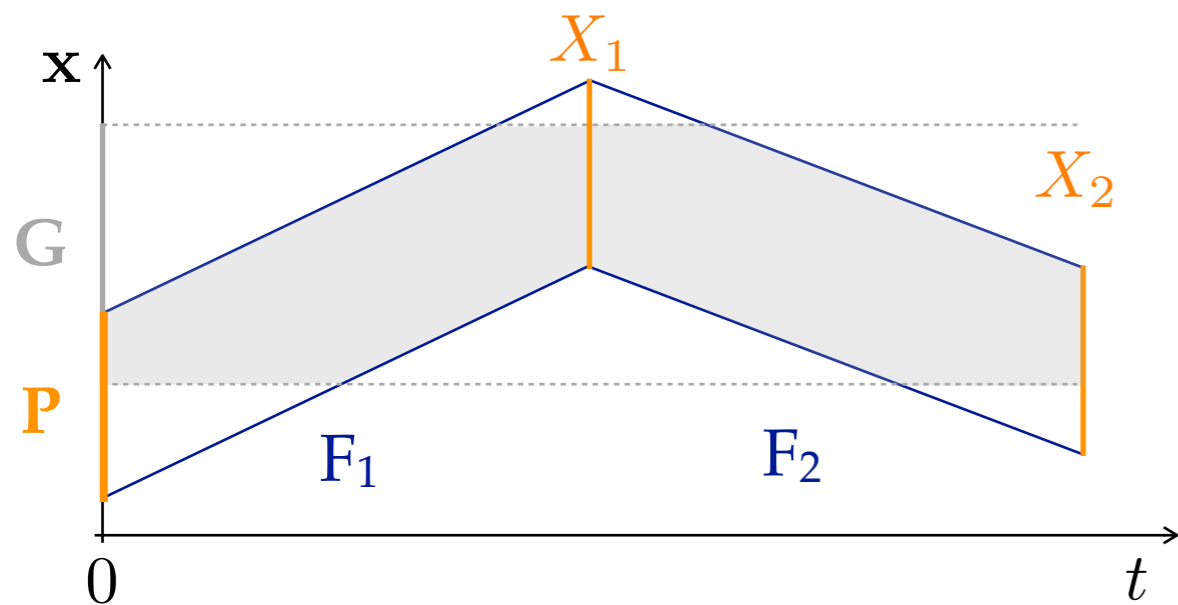


$$G_1 = F_1 \cap G$$

$$G_2 = F_2 \cap G$$

$$A^{\text{aux}} = \text{POST} (P, G_1, \text{Flow}(q_1))$$

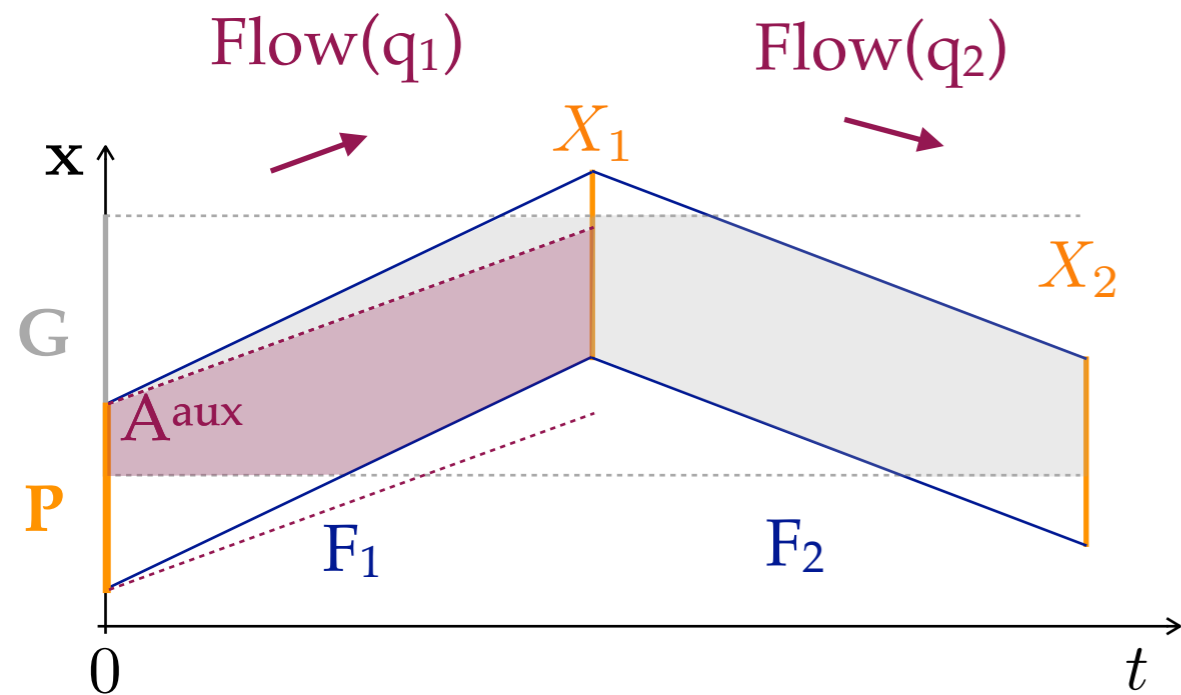
$$A = \text{PRE} (X_1, A^{\text{aux}}, \text{Flow}(q_2))$$



$$B^{\text{aux}} = \text{PRE} (X_1, P, \text{Flow}(q_1))$$

$$B = \text{POST} (B^{\text{aux}}, G_2, \text{Flow}(q_1))$$

Reachability computation

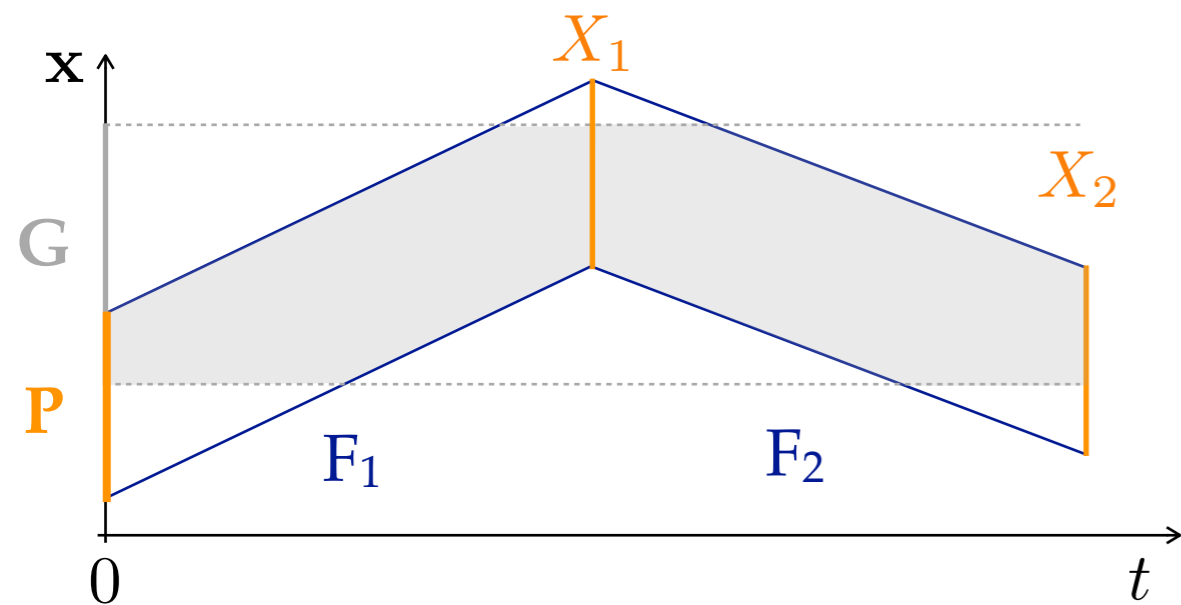


$$G_1 = F_1 \cap G$$

$$G_2 = F_2 \cap G$$

$$A^{\text{aux}} = \text{POST} (P, G_1, \text{Flow}(q_1))$$

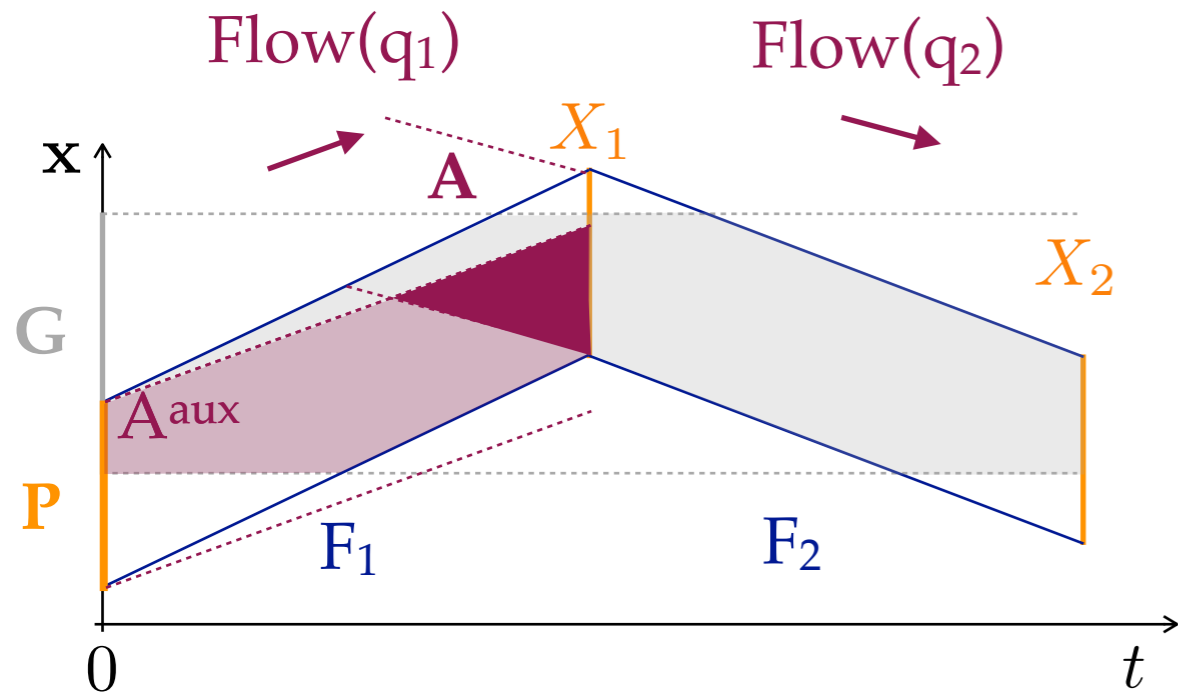
$$A = \text{PRE} (X_1, A^{\text{aux}}, \text{Flow}(q_2))$$



$$B^{\text{aux}} = \text{PRE} (X_1, P, \text{Flow}(q_1))$$

$$B = \text{POST} (B^{\text{aux}}, G_2, \text{Flow}(q_1))$$

Reachability computation

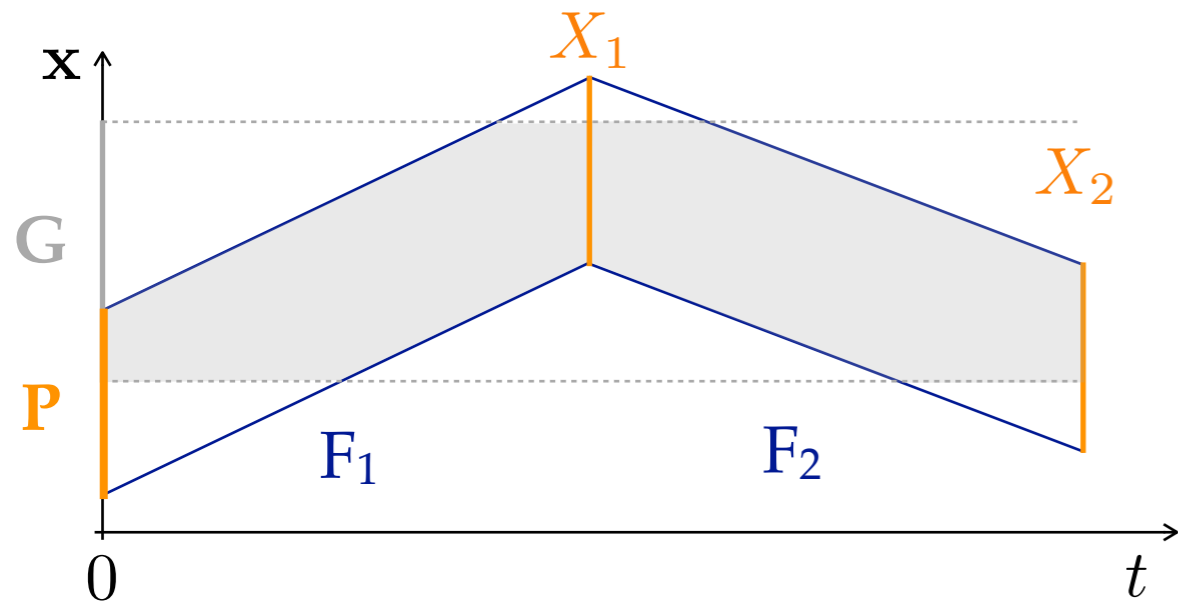


$$G_1 = F_1 \cap G$$

$$G_2 = F_2 \cap G$$

$$A^{\text{aux}} = \text{POST} (P, G_1, \text{Flow}(q_1))$$

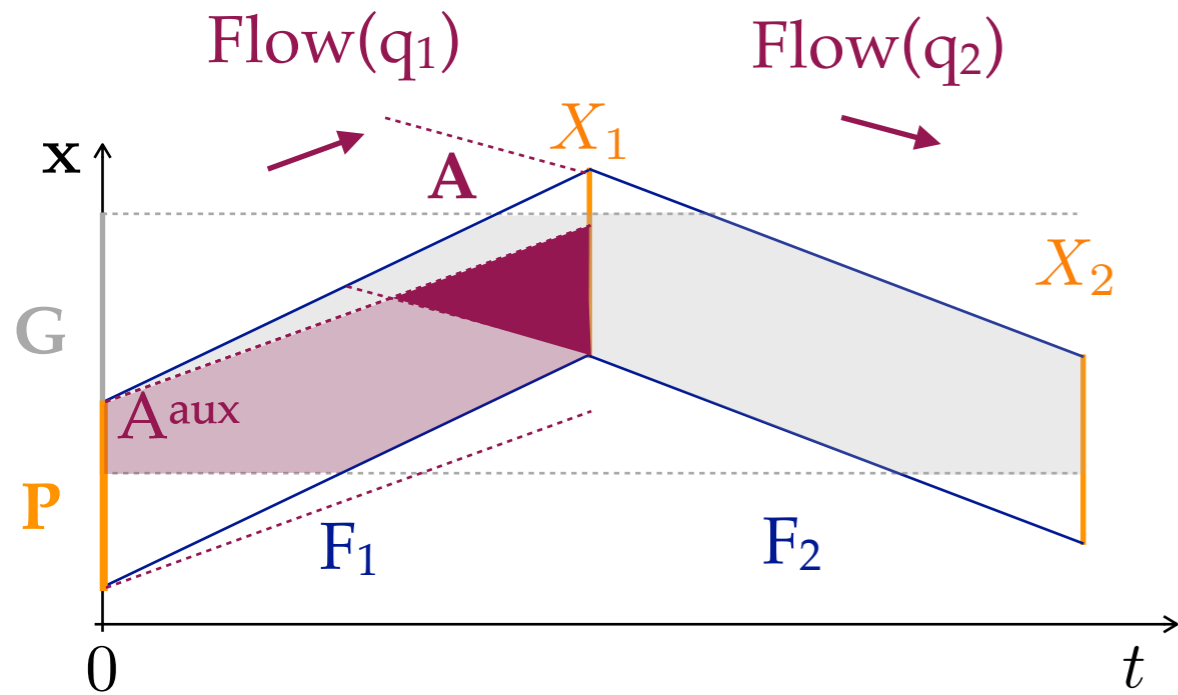
$$A = \text{PRE} (X_1, A^{\text{aux}}, \text{Flow}(q_2))$$



$$B^{\text{aux}} = \text{PRE} (X_1, P, \text{Flow}(q_1))$$

$$B = \text{POST} (B^{\text{aux}}, G_2, \text{Flow}(q_2))$$

Reachability computation

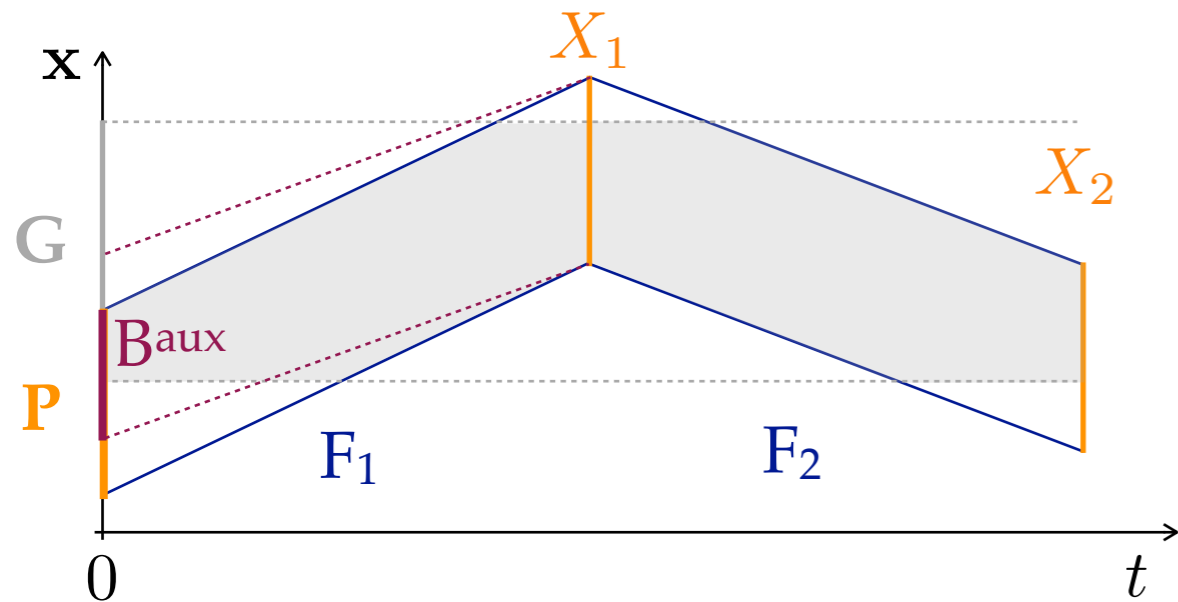


$$A_{aux} = \text{POST} (P, G_1, \text{Flow}(q_1))$$

$$A = \text{PRE} (X_1, A_{aux}, \text{Flow}(q_2))$$

$$G_1 = F_1 \cap G$$

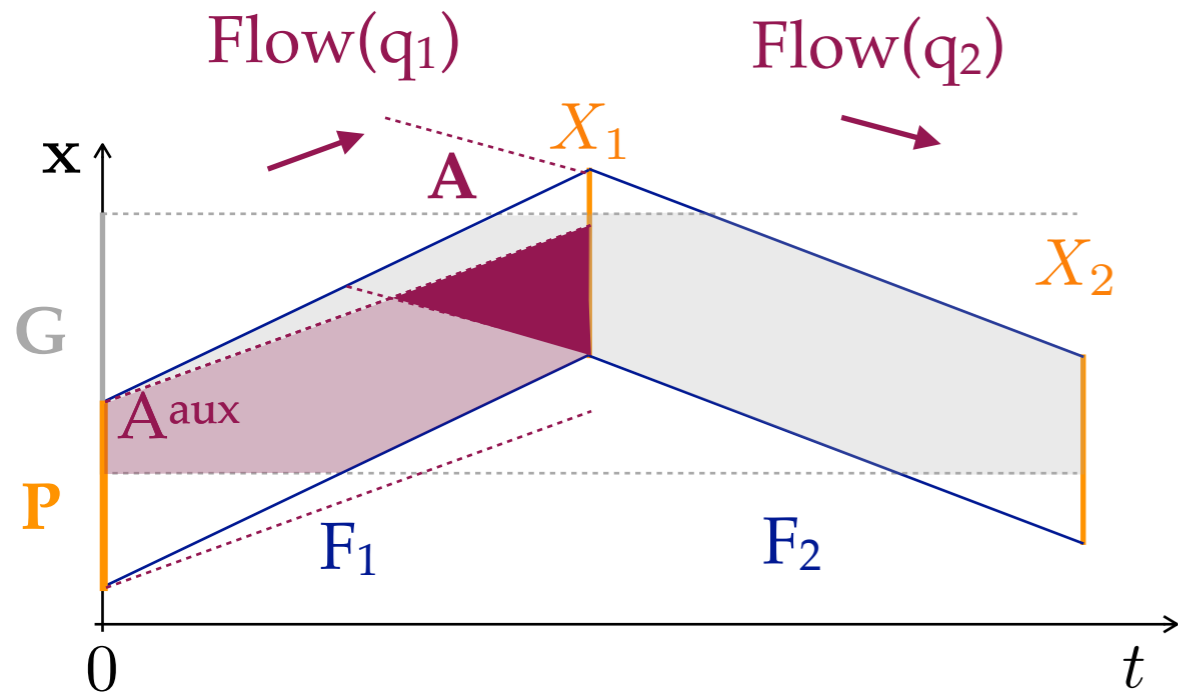
$$G_2 = F_2 \cap G$$



$$B_{aux} = \text{PRE} (X_1, P, \text{Flow}(q_1))$$

$$B = \text{POST} (B_{aux}, G_2, \text{Flow}(q_2))$$

Reachability computation

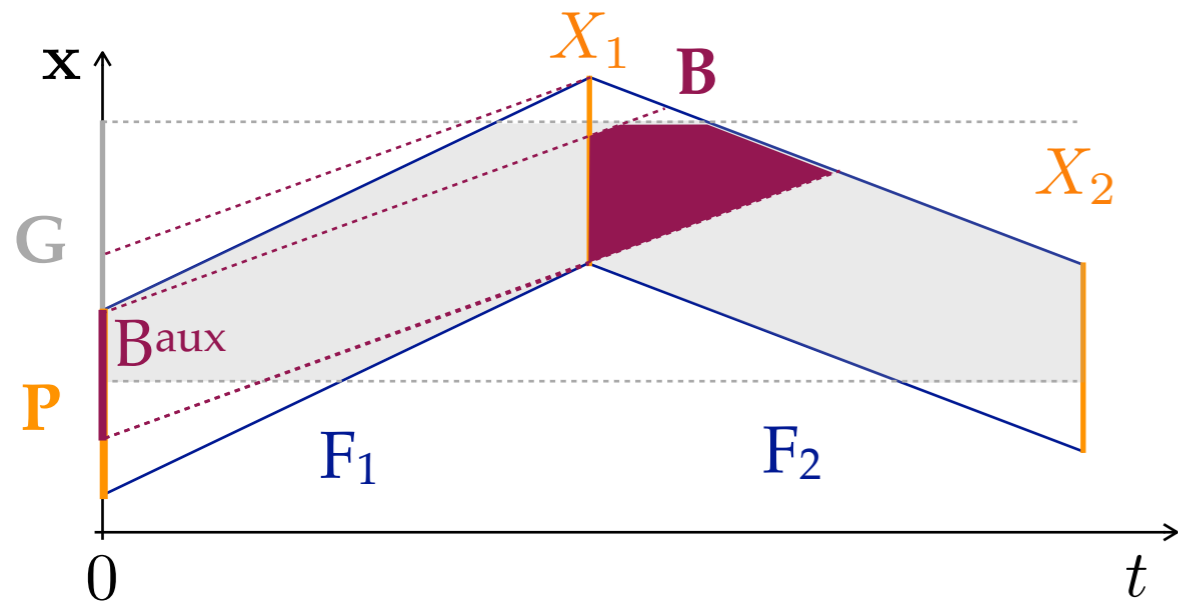


$$A_{aux} = \text{POST} (P, G_1, \text{Flow}(q_1))$$

$$A = \text{PRE} (X_1, A_{aux}, \text{Flow}(q_2))$$

$$G_1 = F_1 \cap G$$

$$G_2 = F_2 \cap G$$

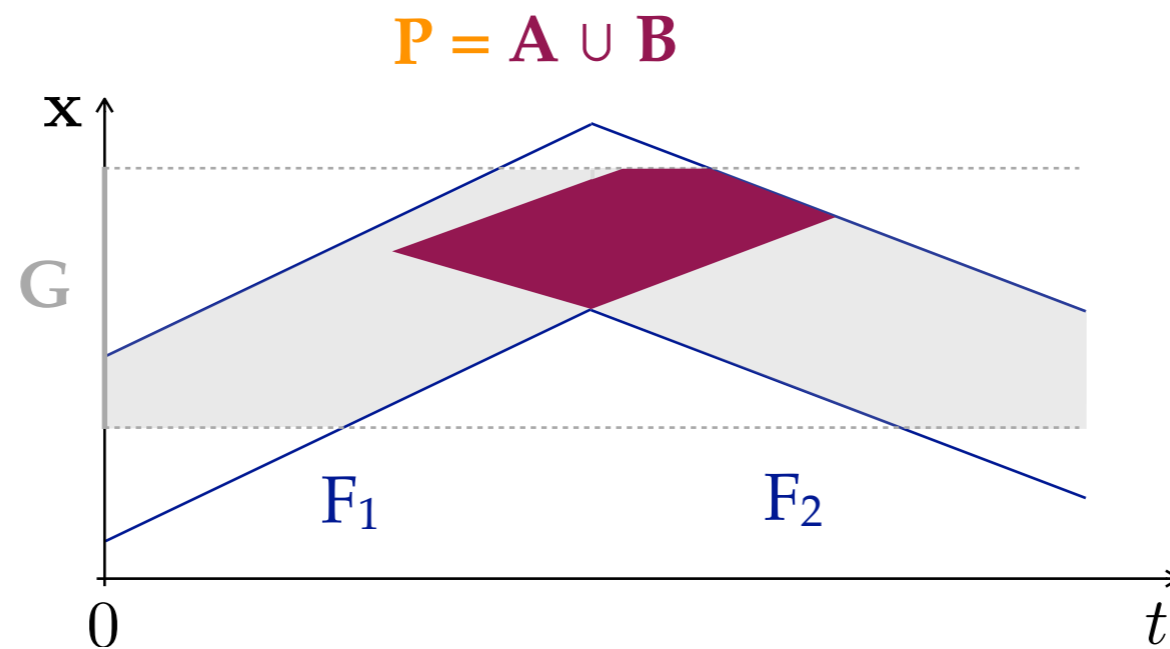


$$B_{aux} = \text{PRE} (X_1, P, \text{Flow}(q_1))$$

$$B = \text{POST} (B_{aux}, G_2, \text{Flow}(q_1))$$

Reachability computation

Reachable switching set



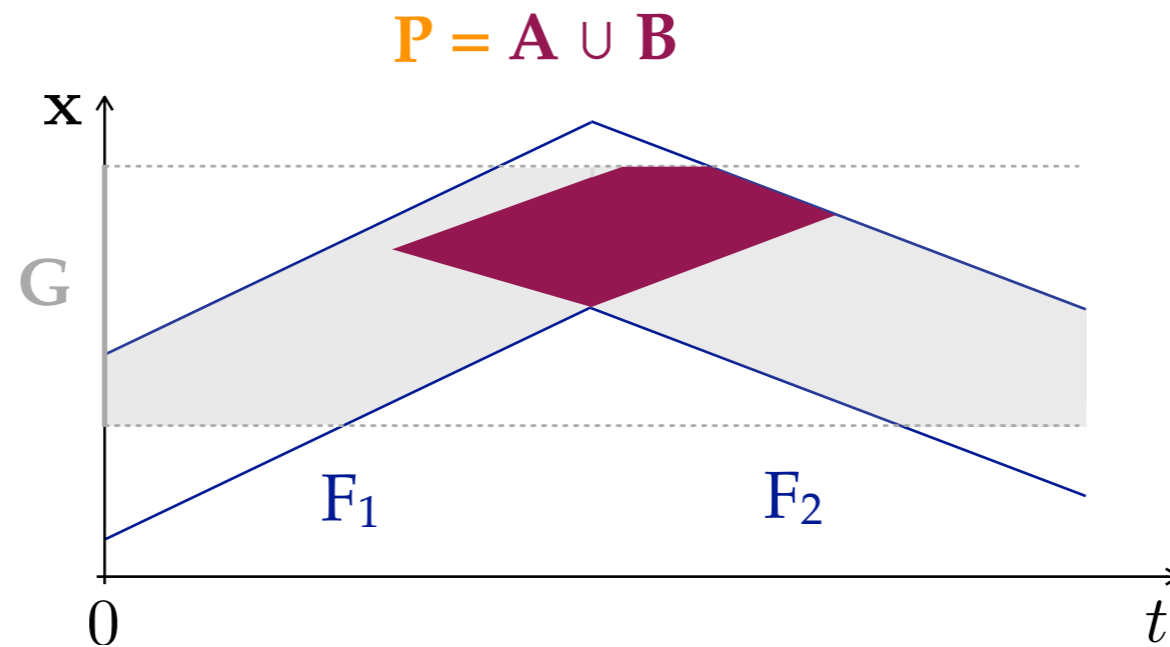
- ❖ Compute iteratively the reachable switching set P until the last ε -tube piece:

Last $P \neq \emptyset \Rightarrow$ membership True

Last $P = \emptyset \Rightarrow$ membership False

Reachability computation

Reachable switching set



- ❖ Compute iteratively the reachable switching set P until the last ε -tube piece:

Last $P \neq \emptyset \Rightarrow$ membership True

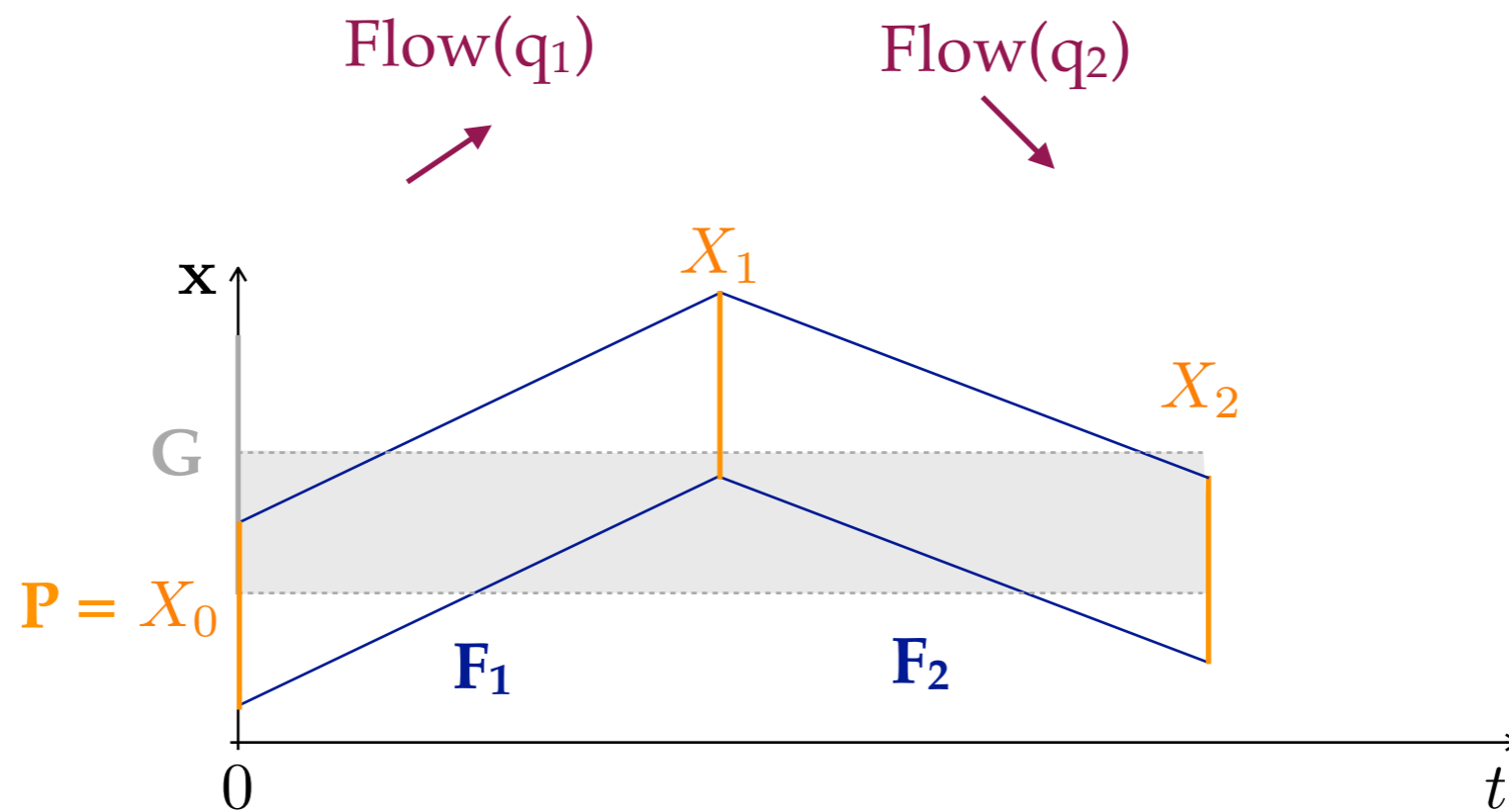
Last $P = \emptyset \Rightarrow$ membership False

If membership of f in H is False, then relax the LHA model.

Relaxation

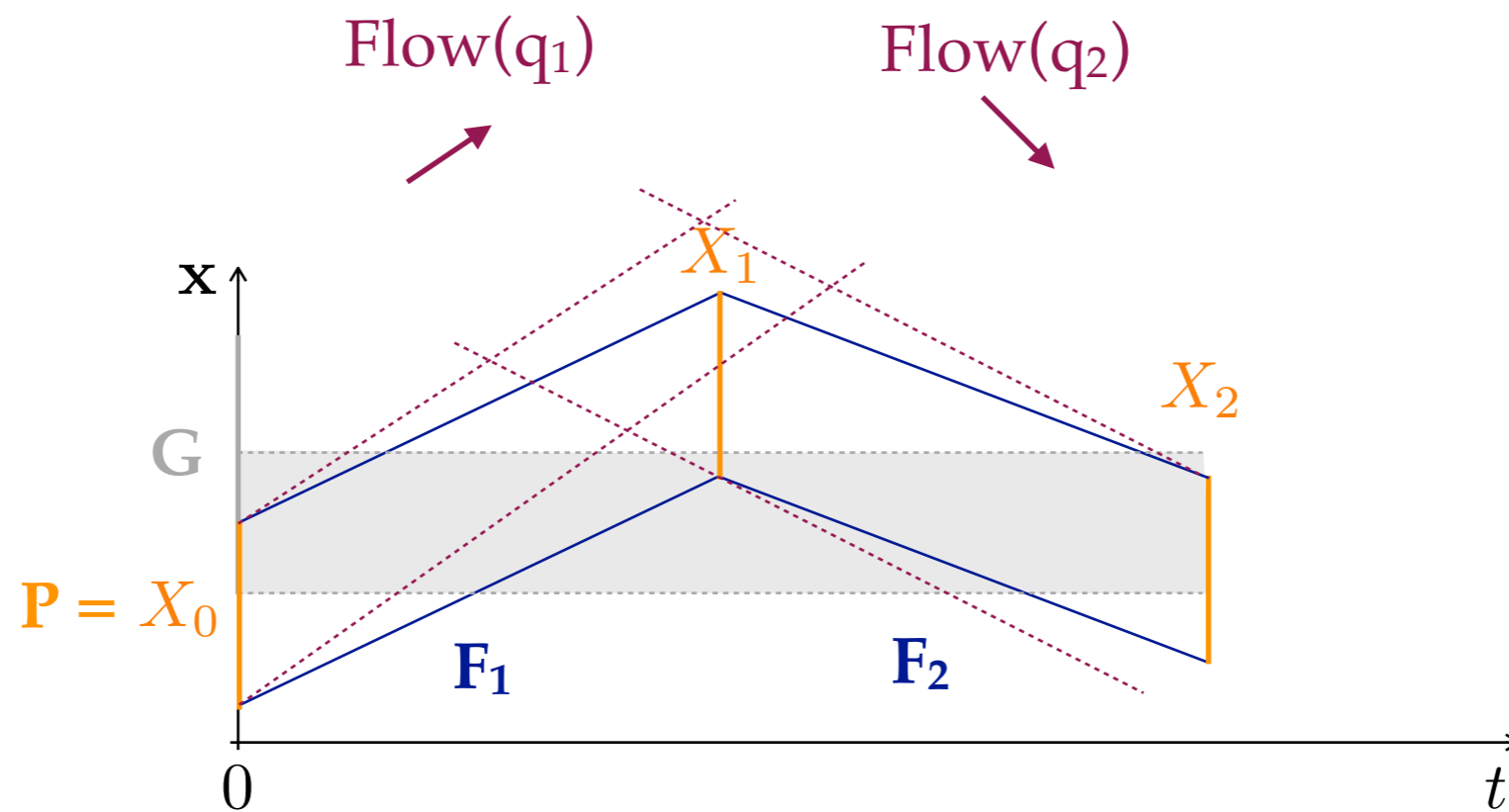
Relaxation problem

Given an LHA H and a m-PWL function f , does there exist an execution σ in H consistent with f and such that $d(f, \sigma) \leq \varepsilon$ by expanding the guards and invariants of H ?



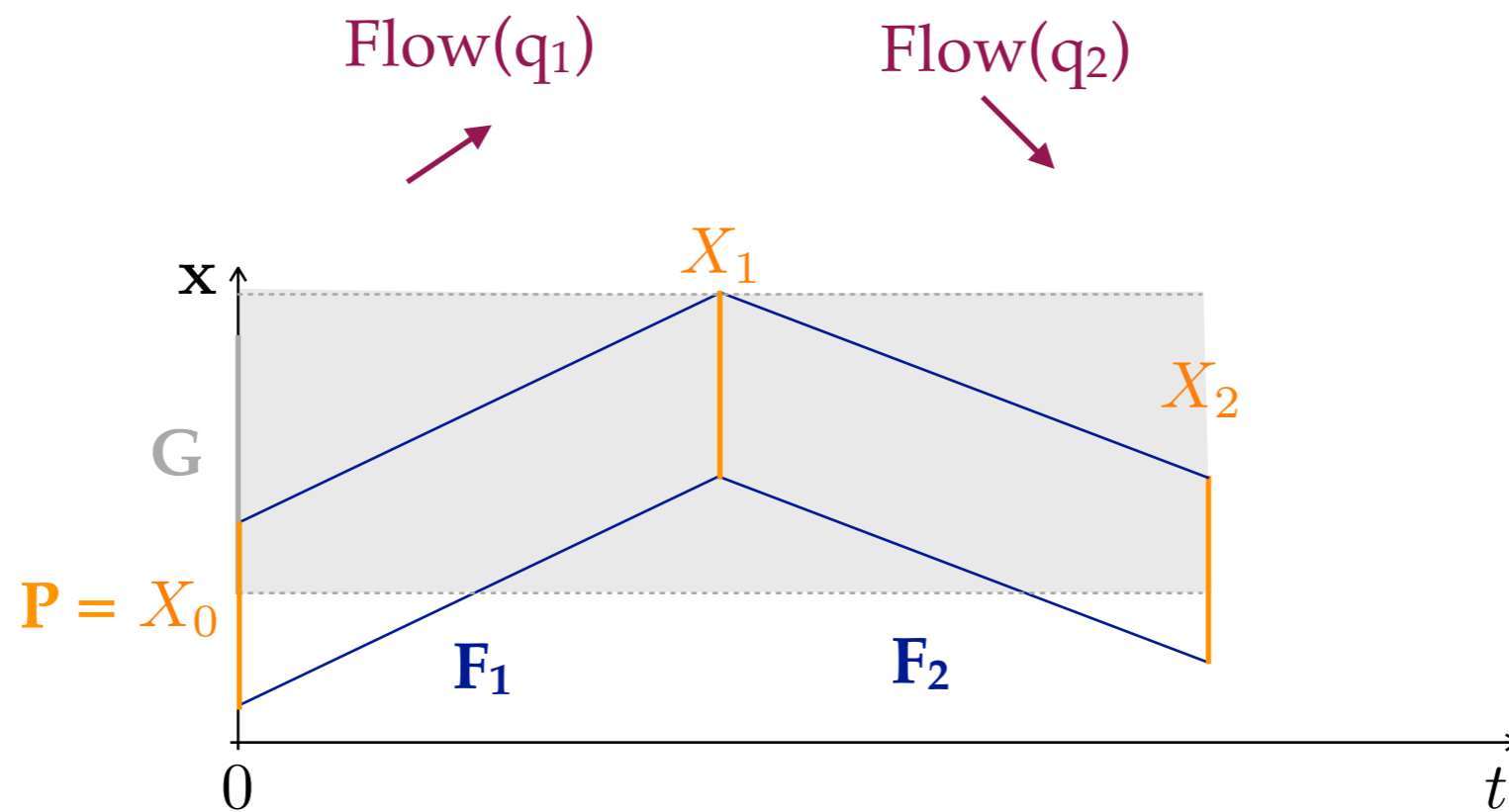
Relaxation problem

Given an LHA H and a m-PWL function f , does there exist an execution σ in H consistent with f and such that $d(f, \sigma) \leq \varepsilon$ by expanding the guards and invariants of H ?



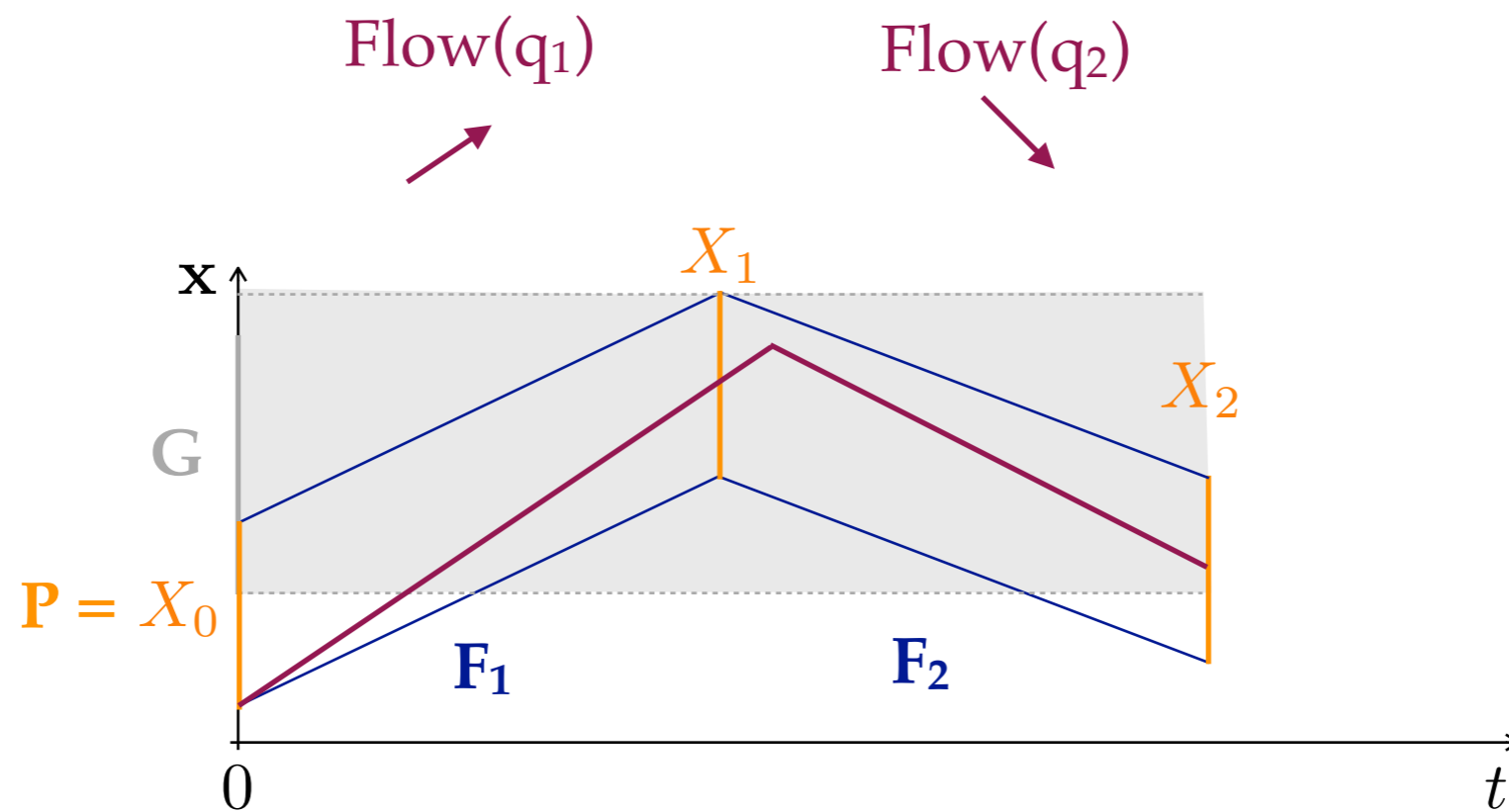
Relaxation problem

Given an LHA H and a m-PWL function f , does there exist an execution σ in H consistent with f and such that $d(f, \sigma) \leq \varepsilon$ by expanding the guards and invariants of H ?



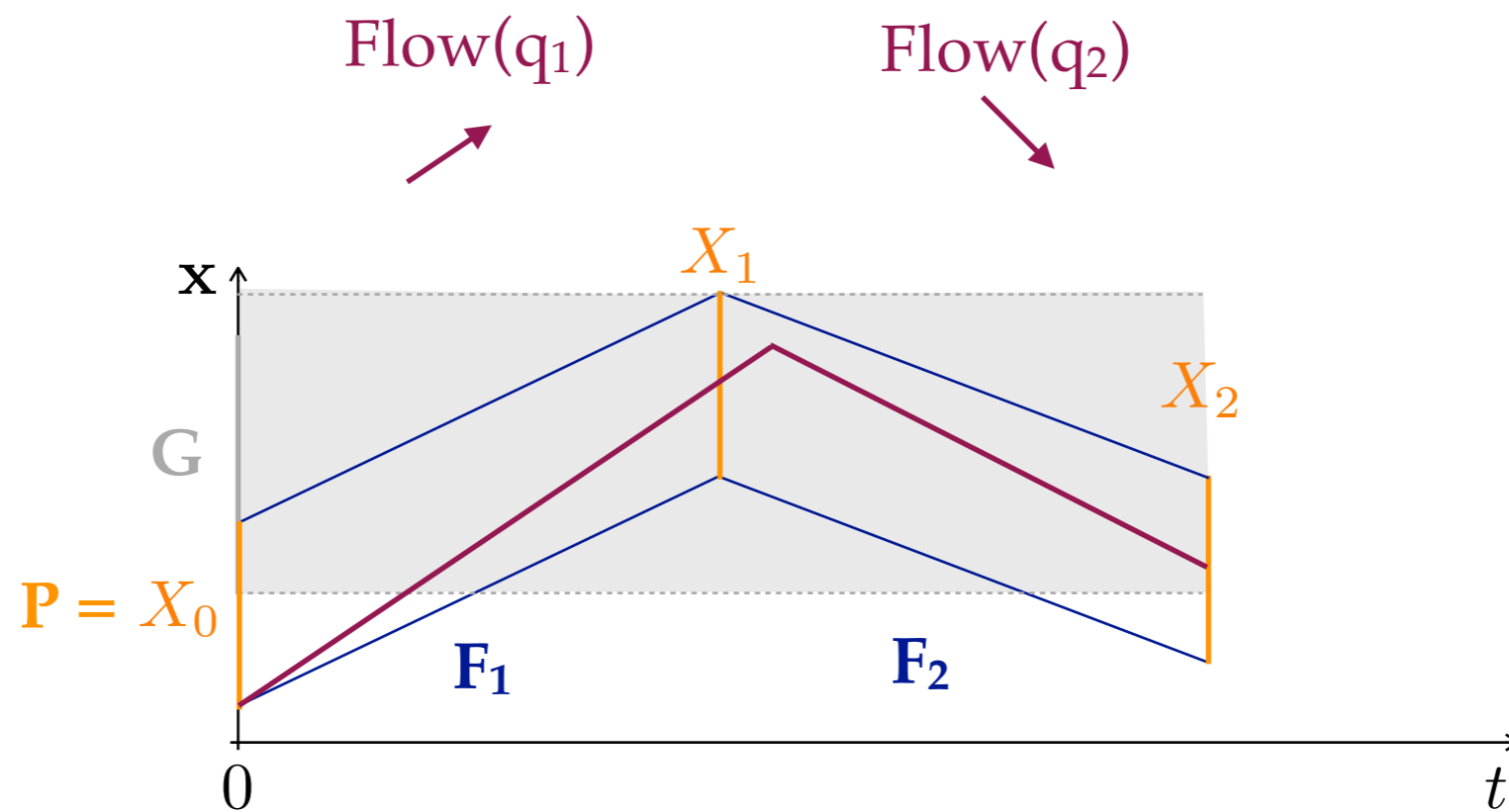
Relaxation problem

Given an LHA H and a m-PWL function f , does there exist an execution σ in H consistent with f and such that $d(f, \sigma) \leq \varepsilon$ by expanding the guards and invariants of H ?



Relaxation problem

Given an LHA H and a m-PWL function f , does there exist an execution σ in H consistent with f and such that $d(f, \sigma) \leq \varepsilon$ by expanding the guards and invariants of H ?



If relaxation of H fails, then modify the discrete structure of the LHA model.

Adaptation

Adaptation problem

Given an LHA H , a PWL function f and a path π in H , construct a path π' by preserving locations in π such that there exists an execution σ in H for π' with $d(f, \sigma) \leq \varepsilon$

Adaptation procedure

- ❖ Recall $f \equiv p_1, \dots, p_m$ and $\pi \equiv q_1, \dots, q_m$
- ❖ Construct $\pi' = \pi$
- ❖ Compute iteratively the reachable switching sets P_i until emptiness
- ❖ If $P_i = \emptyset$, then:
 1. Replace q_i by q'_i in π'
 2. Replace q_{i-1} by q'_{i-1} in π'
 3. From q_{i-2} to q_1
 4. Until $P_i \neq \emptyset$

Adaptation problem

Given an LHA H , a PWL function f and a path π in H , construct a path π' by preserving locations in π such that there exists an execution σ in H for π' with $d(f, \sigma) \leq \varepsilon$

Adaptation procedure

- ❖ Recall $f \equiv p_1, \dots, p_m$ and $\pi \equiv q_1, \dots, q_m$
- ❖ Construct $\pi' = \pi$
- ❖ Compute iteratively the reachable switching sets P_i until emptiness
- ❖ If $P_i = \emptyset$, then:
 1. Replace q_i by q'_i in π'
 2. Replace q_{i-1} by q'_{i-1} in π'
 3. From q_{i-2} to q_1
 4. Until $P_i \neq \emptyset$

Replacement consists of 2 different strategies:

1. Existing q'_i with $\text{Flow}(q'_i) \cong \text{slope}(p_i)$
2. New q'_i with $\text{Flow}(q'_i) = \text{slope}(p_i)$

Adaptation problem

Given an LHA H , a PWL function f and a path π in H , construct a path π' by preserving locations in π such that there exists an execution σ in H for π' with $d(f, \sigma) \leq \varepsilon$

Adaptation procedure

- ❖ Recall $f \equiv p_1, \dots, p_m$ and $\pi \equiv q_1, \dots, q_m$
- ❖ Construct $\pi' = \pi$
- ❖ Compute iteratively the reachable switching sets P_i until emptiness
- ❖ If $P_i = \emptyset$, then:
 1. Replace q_i by q'_i in π'
 2. Replace q_{i-1} by q'_{i-1} in π'
 3. From q_{i-2} to q_1
 4. Until $P_i \neq \emptyset$

Replacement consists of 2 different strategies:

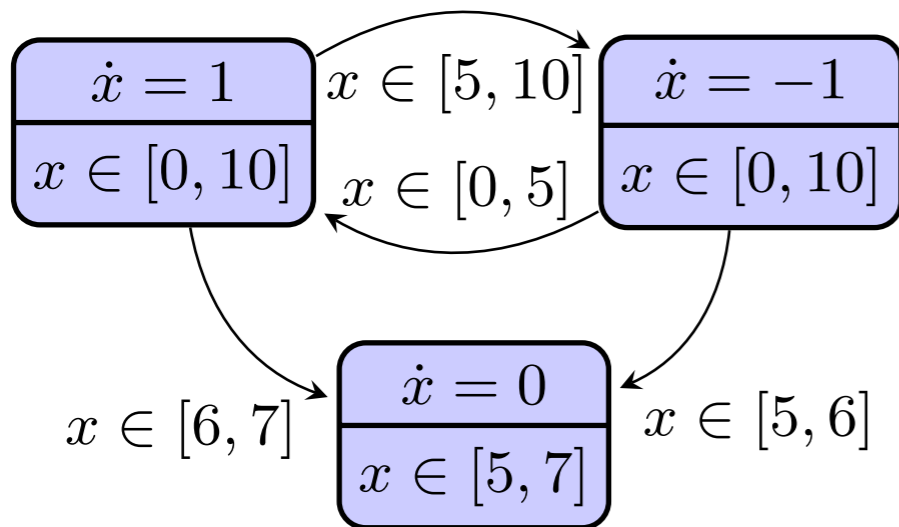
1. Existing q'_i with $\text{Flow}(q'_i) \cong \text{slope}(p_i)$
2. New q'_i with $\text{Flow}(q'_i) = \text{slope}(p_i)$

Modify the LHA H to include the location path π' .

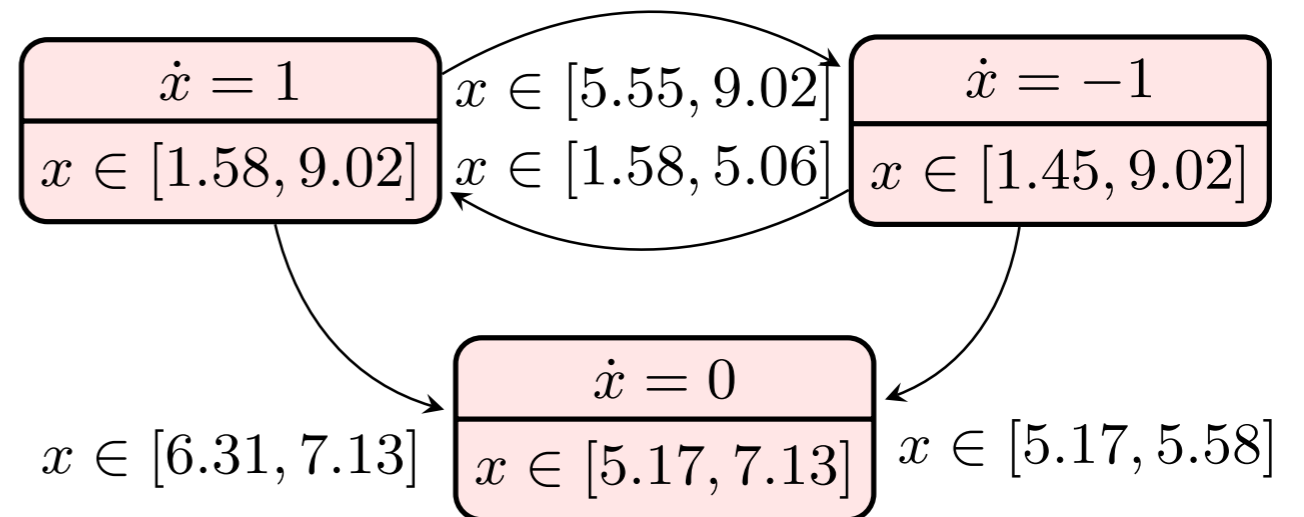
Experiments

Synthetic data

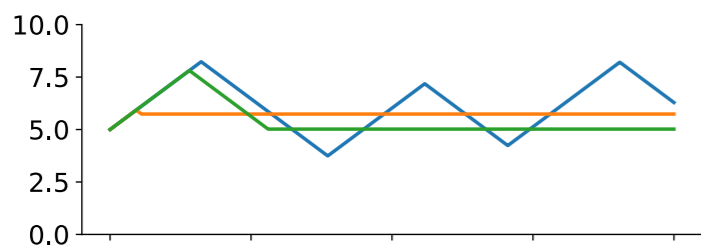
Original LHA



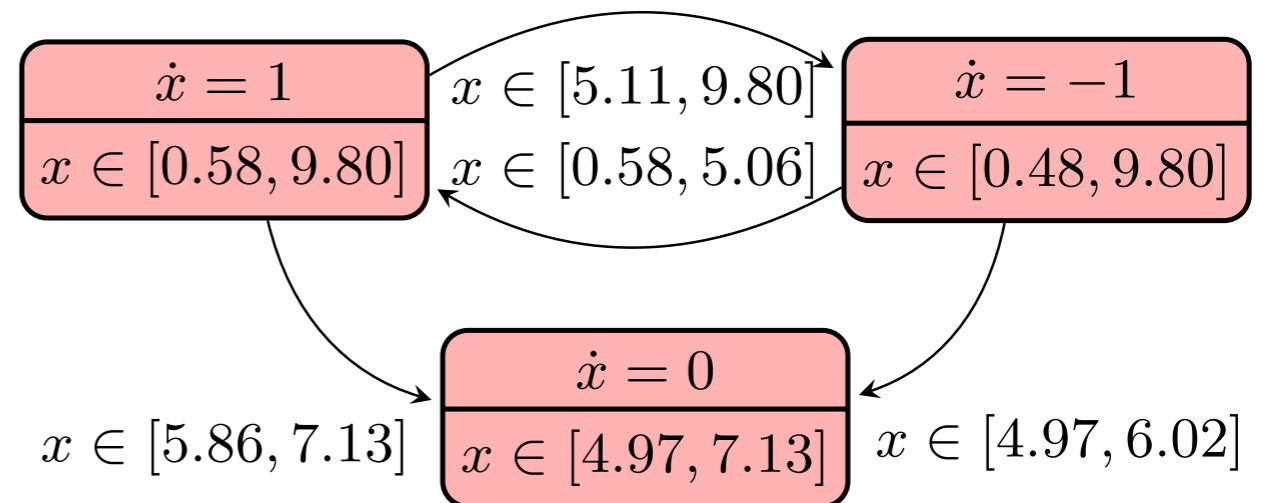
Synthesized LHA after 10 iterations



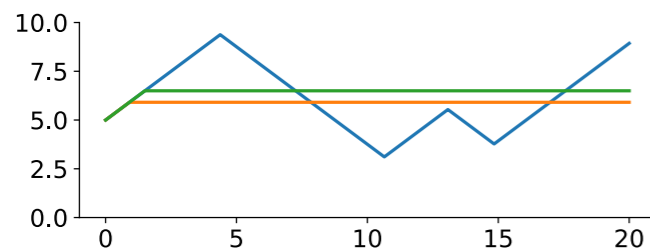
Executions of original LHA



Synthesized LHA after 100 iterations

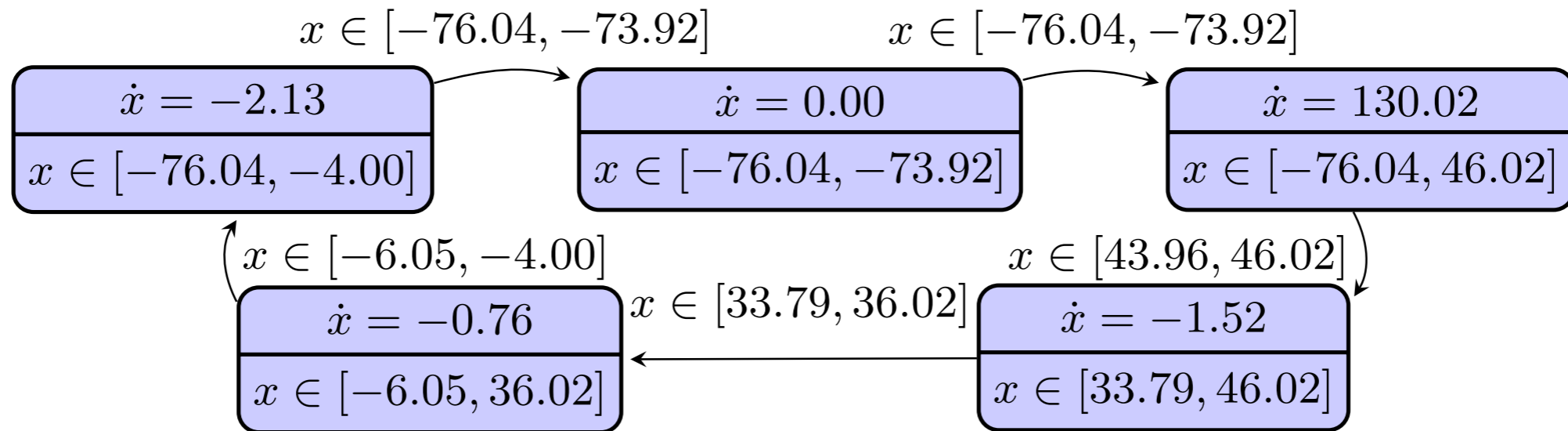


Executions of synthesized LHA

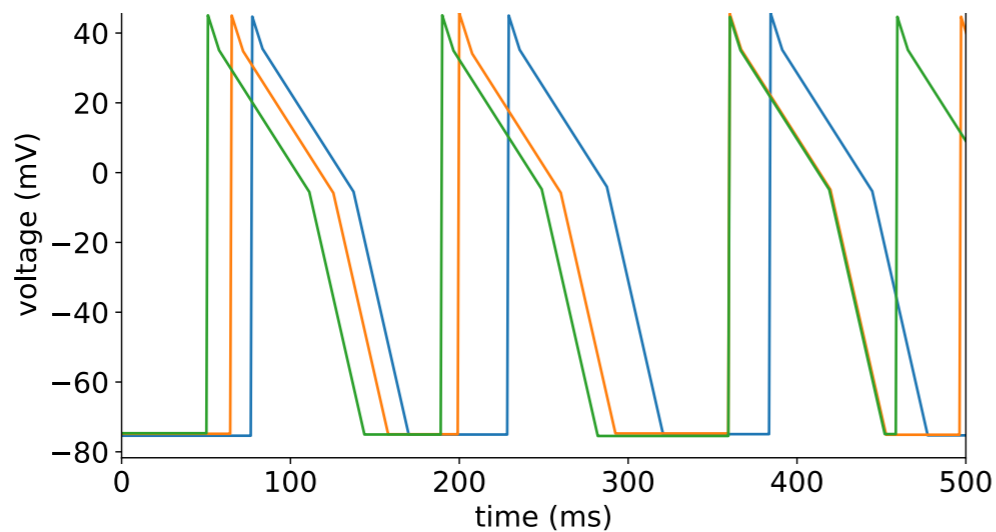


Cell model

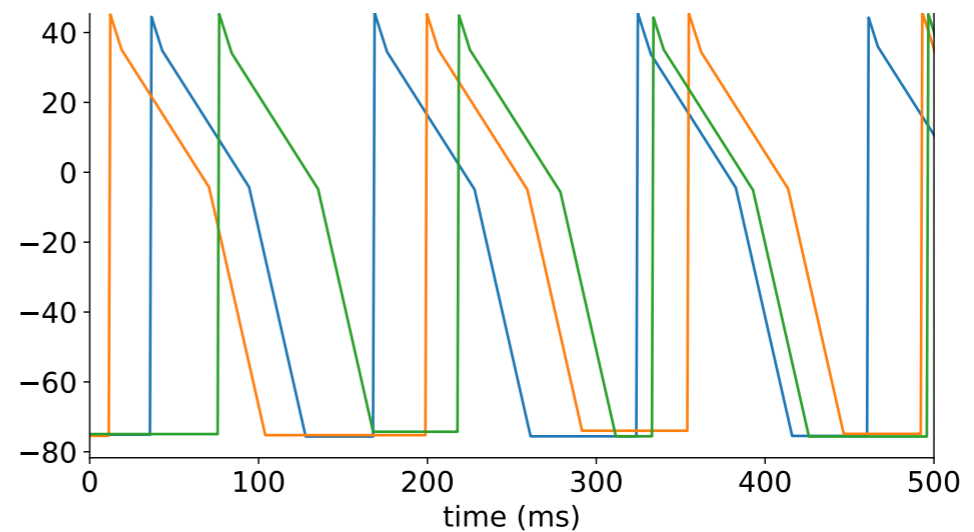
Synthesized LHA



Input PWL functions



Simulations of the synthesized LHA



Conclusion

Summary

- ❖ Development of **two automatic approaches for synthesizing** linear hybrid automata from experimental data
- ❖ **Nondeterministic** guards and invariants
- ❖ **Arbitrary** number and topology of **locations**
- ❖ The synthesized automaton reproduce the data up to a **tolerance**
- ❖ Both algorithms well-suited for **online and synthesis-in-the-loop applications**
- ❖ **SMT-based approach minimizes** the size of the **model** but it is feasible for limited data sets
- ❖ **Membership-based approach trades-off** between **size and precision** of the model

Questions?