

Synthesis of Linear Hybrid Automata from Experimental Data

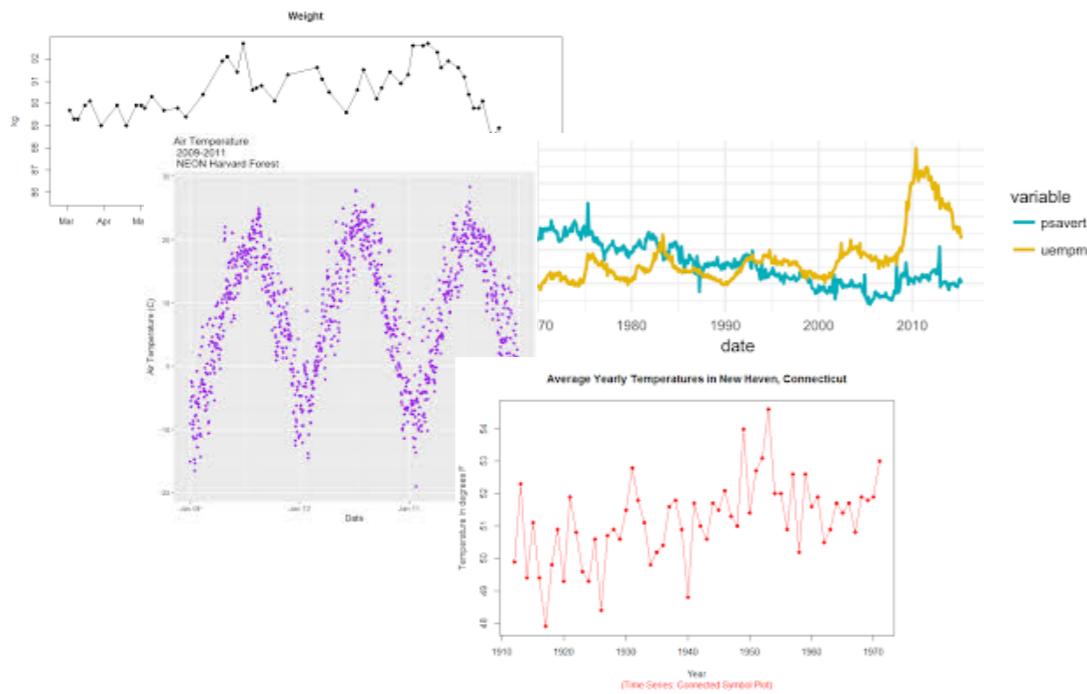
Miriam García Soto

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

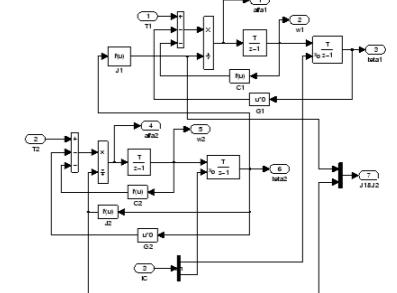
IST Austria - 2019

Motivation

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



Analysis

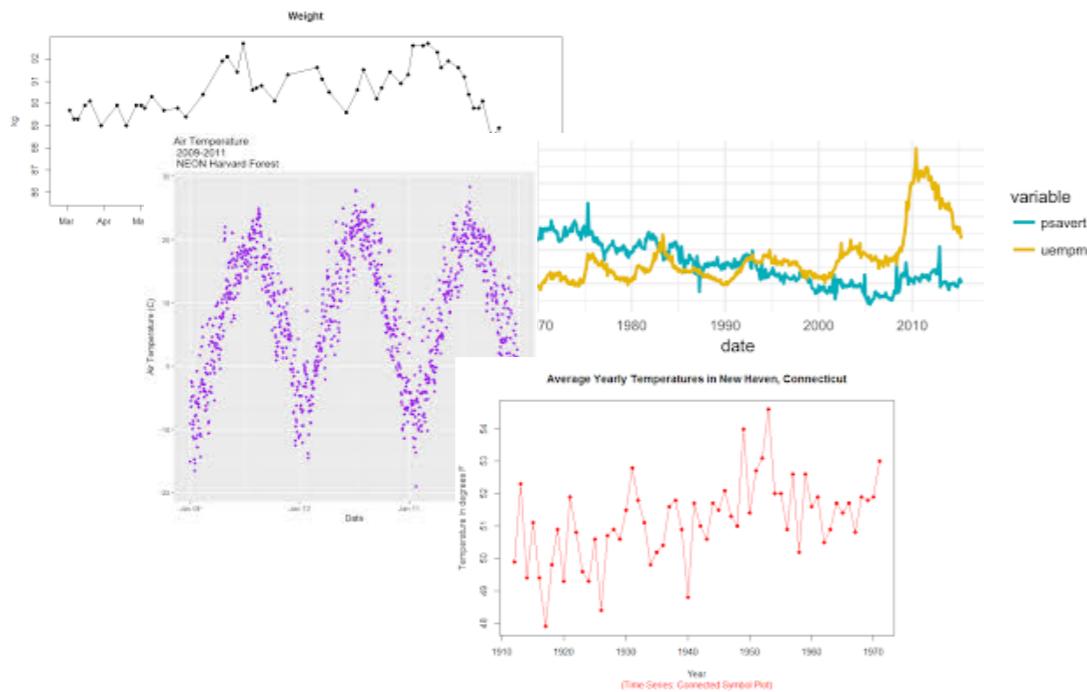


Model

Experimental data

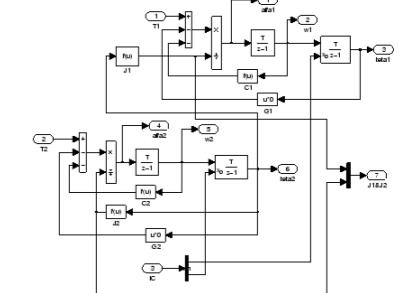
Motivation

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



Experimental data

Analysis



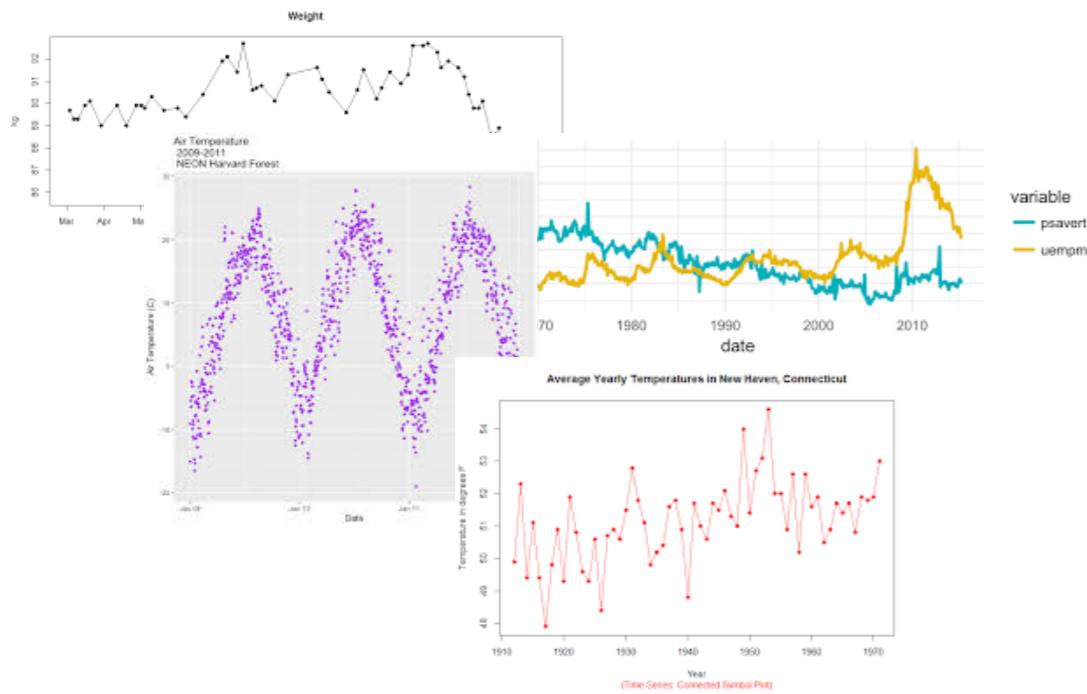
Model

Challenge

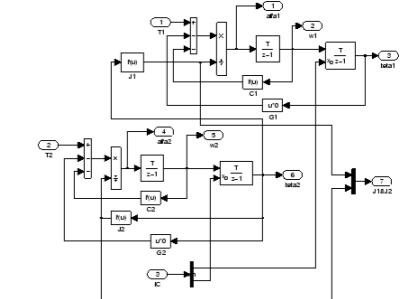
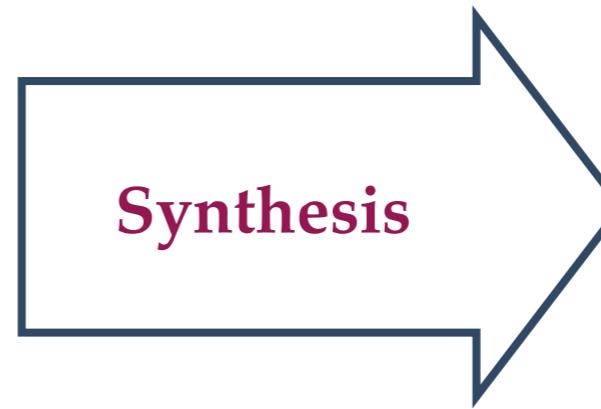
How to automatically create a model?

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

Step Response Analysis of Thermotaxis in *Caenorhabditis elegans*

Hatim A. Zariwala,^{1,*} Adam C. Miller,^{2,*} Serge Faumont,^{2,*} and Shawn R. Lockery²

¹Graduate Program in Neurobiology and Behavior, State University of New York at Stony Brook, Stony Brook Neuroscience, University of Oregon, Eugene, Oregon 97403

The nematode *Caenorhabditis elegans* migrates toward a preferred temperature on a thermal gradient in *C. elegans* has been identified, but the behavioral strategy implemented by this worm to detect thermal gradients and achieve thermotaxis is not fully understood. Wild-type, cryophilic, or thermophilic worms respond differently to thermal gradients. Each of these three types of worms we analyzed showed distinct response patterns. Comparison of wild-type and mutant response patterns suggested a model in which response to the gradient depending on the orientation of the worm relative to its preferred probability was modulated in a manner consistent with a rule for turning behavior in thermal systems for thermotaxis and chemotaxis may converge on a common behavioral mechanism.

Key words: *C. elegans*; nematode; thermotaxis; spatial orientation; stochastic model; sensor

Introduction

Caenorhabditis elegans orients to both chemical (chemotaxis) and thermal (thermotaxis) gradients (Ward, 1973; Hedgecock and Russell, 1975), making it a prime model system for investigating the neural 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 by brief turns, called "reversals" by bouts of turning (Rutherford and Croall, 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 the same 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 (Ω) (Croall, 1978). Step-response analysis reveals that reversal and omega turns in *C. elegans* have been termed "pirouettes" (Pierce-Shimomura et al., 1999). Pirouette probability is modulated by the rate of change of chemical concentrations (dC/dt): when $dC/dt < 0$, reversal probability is increased, whereas when $dC/dt > 0$, reversal probability is decreased. Thus, reversal down the gradient are truncated, and runs up the gradient are extended, resulting in net movement toward the gradient peak.

The committee on chemotaxis in *C. elegans* proposed the "two-pole hypothesis" for the behavioral mechanism of the migration phase of thermotaxis (Ryu and Samuel, 2002). Tracks of *C. elegans* in

In their classic study of thermotaxis in *Caenorhabditis elegans*, Hedgecock and Riddle (1975) demonstrated that worms migrate toward a spatial gradient, moving near the calibration temperature (T_c). Within 3°C of this temperature, worms turn back and forth, exhibiting a "stochastic walk." At temperatures further from T_c , however, the turns that failed to aggregate at T_c , and that could be classified as "irreversible" (aggregating in neither region of a platemetric legume), do reveal a regular, if slightly static (non-aggregating) pattern. On the basis of this classification, Hedgecock and Riddle (1975) concluded that the two-pole hypothesis for thermotaxis is correct, and that the two poles of the thermophilic drive produce thermotactic behavior and that isoether mal tracking occurs at their balance.

More recently, Ryu and Samuel (2002) have shown that the movements involved in thermotaxis are not the result of a simple two-pole model, but rather a complex process involving many neurons.

Materials and Methods

Animals. *C. elegans* [B1, N2, and CB112] were grown in mixed lawns containing nematode growth medium (OP50) (Brenner, 1974).

Thermal gradient. Apparatus. The device in Figure 1A, a thin agar plate with two glass plates, is used to measure the rate of change in temperature between two static fluid-filled chambers.

The chambers were made of two pieces of aluminum foil. Most of the time, we used two pieces of aluminum foil.

Copyright © 2002 Society for Neuroscience 0270-6474/02/22562-15\$15.00/0

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

William S. Ryu and Arvindhanan D. T. Samuel

Department of Molecular and Cellular Biology, Harvard University, Cambridge, Massachusetts 02142

In a spatial thermal gradient, *Caenorhabditis elegans* migrates toward and then isolates itself near the calibration temperature. This thermotactic drive (involving the neurons AFD and AIY) and cryophilic drive (involving the neuron AIZ) balance at the calibration temperature (T_c). If the worm is at T_c , it turns back to T_c . If so, then probability should increase with distance from T_c . If not, then probability should increase with distance from T_c . We tested this hypothesis by measuring the probability of reversals for different thermal gradients. We found evidence for a mechanism for migration toward thermal gradients, and a mechanism for isoether tracking that is active near the calibration temperature (T_c). However, we found no evidence for a mechanism for migration away from thermal gradients below the calibration temperature that might have supported the two-pole hypothesis. Our results support the two-pole hypothesis for thermotaxis and isoether tracking control the thermal gradient.

Key words: *C. elegans*; nematode; thermotaxis; spatial orientation; stochastic model; sensor

In their classic study of thermotaxis in *Caenorhabditis elegans*, Hedgecock and Riddle (1975) demonstrated that worms migrate toward a spatial gradient, moving near the calibration temperature (T_c). Within 3°C of this temperature, worms turn back and forth, exhibiting a "stochastic walk." At temperatures further from T_c , however, the turns that failed to aggregate at T_c , and that could be classified as "irreversible" (aggregating in neither region of a platemetric legume), do reveal a regular, if slightly static (non-aggregating) pattern. On the basis of this classification, Hedgecock and Riddle (1975) concluded that the two-pole hypothesis for thermotaxis is correct, and that the two poles of the thermophilic drive produce thermotactic behavior and that isoether mal tracking occurs at their balance.

More recently, Ryu and Samuel (2002) have shown that the movements involved in thermotaxis are not the result of a simple two-pole model, but rather a complex process involving many neurons.

Materials and Methods

Animals. *C. elegans* [B1, N2, and CB112] were grown in mixed lawns containing nematode growth medium (OP50) (Brenner, 1974).

Thermal gradient. Apparatus. The device in Figure 1A, a thin agar plate with two glass plates, is used to measure the rate of change in temperature between two static fluid-filled chambers.

The chambers were made of two pieces of aluminum foil. Most of the time, we used two pieces of aluminum foil.

Copyright © 2002 Society for Neuroscience 0270-6474/02/22562-15\$15.00/0

Cell



The Journal of Neuroscience, July 1, 2002, 22(15):4369–4377 • 4369

© 2002 Society for Neuroscience 0270-6474/02/224369-09\$15.00/0

Received Nov. 27, 2001; revised April 3, 2002; accepted April 12, 2002.

This work was supported by National Health Grant R01NS04138. We thank I. Winship for advice and sharing unpublished results and M. Roberts for the suggestion to adapt the trapezoidal apparatus for temperature response analysis. We also thank the International C. elegans Genetics Center, which is funded by the United States National Center for Research Resources.

*H.A.Z., A.C.M., and D.T.S. contributed equally to this work.

Correspondence should be addressed to Dr. Arvindhanan D. T. Samuel, Department of Molecular and Cellular Biology, Harvard University, Cambridge, MA 02142. E-mail: arvindhanan.samuel@harvard.edu.

Copyright © 2002 Society for Neuroscience 0270-6474/02/224369-09\$15.00/0

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

Saul Kato,^{1,4} Hattie S. Kaplan,^{1,4} Tina Schröder,^{1,4} Suzanne Scorsa,¹ Theodore H. Lindsay,^{2,22} Eviatar Yemini,³

Shawn Lockery,¹ and Manuel Zimmer^{1,4}

¹Research Institute of Molecular Pathology, IMP, Vienna Biocenter, VBC, Dr. Bohr-Gasse 7, 1030 Vienna, Austria

²Department of Biochemistry, University of Oregon, Eugene, OR 97403, USA

³Department of Biochemistry and Molecular Biology, Howard Hughes Medical Institute, Columbia University Medical Center, New York, NY 10032, USA

⁴Present address: Division of Biology and Biological Engineering, California Institute of Technology, Pasadena, CA 91125, USA

Correspondence should be addressed to Dr. Manuel Zimmer (E-mail: zimmer@imp.ac.at).

DOI:10.1162/027064702760161344

Copyright © 2002 Society for Neuroscience 0270-6474/02/224369-09\$15.00/0

Received June 1, 2001; revised April 12, 2002; accepted April 12, 2002.

This work was supported by grants from the Austrian Science Fund FWF (S.K., T.S., S.Z.) and the European Union (S.Z.).

We thank Veronika Müller and Howard Berg, and members of our laboratories for useful discussions.

Correspondence should be addressed to Dr. Manuel Zimmer (E-mail: zimmer@imp.ac.at).

DOI:10.1162/027064702760161344

Copyright © 2002 Society for Neuroscience 0270-6474/02/224369-09\$15.00/0

Received June 1, 2001; revised April 12, 2002; accepted April 12, 2002.

This work was supported by grants from the Austrian Science Fund FWF (S.K., T.S., S.Z.) and the European Union (S.Z.).

We thank Veronika Müller and Howard Berg, and members of our laboratories for useful discussions.

Correspondence should be addressed to Dr. Manuel Zimmer (E-mail: zimmer@imp.ac.at).

DOI:10.1162/027064702760161344

Copyright © 2002 Society for Neuroscience 0270-6474/02/224369-09\$15.00/0

Received June 1, 2001; revised April 12, 2002; accepted April 12, 2002.

This work was supported by grants from the Austrian Science Fund FWF (S.K., T.S., S.Z.) and the European Union (S.Z.).

We thank Veronika Müller and Howard Berg, and members of our laboratories for useful discussions.

Correspondence should be addressed to Dr. Manuel Zimmer (E-mail: zimmer@imp.ac.at).

DOI:10.1162/027064702760161344

Copyright © 2002 Society for Neuroscience 0270-6474/02/224369-09\$15.00/0

Received June 1, 2001; revised April 12, 2002; accepted April 12, 2002.

This work was supported by grants from the Austrian Science Fund FWF (S.K., T.S., S.Z.) and the European Union (S.Z.).

We thank Veronika Müller and Howard Berg, and members of our laboratories for useful discussions.

Correspondence should be addressed to Dr. Manuel Zimmer (E-mail: zimmer@imp.ac.at).

DOI:10.1162/027064702760161344

Copyright © 2002 Society for Neuroscience 0270-6474/02/224369-09\$15.00/0

Received June 1, 2001; revised April 12, 2002; accepted April 12, 2002.

This work was supported by grants from the Austrian Science Fund FWF (S.K., T.S., S.Z.) and the European Union (S.Z.).

We thank Veronika Müller and Howard Berg, and members of our laboratories for useful discussions.

Correspondence should be addressed to Dr. Manuel Zimmer (E-mail: zimmer@imp.ac.at).

DOI:10.1162/027064702760161344

Copyright © 2002 Society for Neuroscience 0270-6474/02/224369-09\$15.00/0

Received June 1, 2001; revised April 12, 2002; accepted April 12, 2002.

This work was supported by grants from the Austrian Science Fund FWF (S.K., T.S., S.Z.) and the European Union (S.Z.).

We thank Veronika Müller and Howard Berg, and members of our laboratories for useful discussions.

Correspondence should be addressed to Dr. Manuel Zimmer (E-mail: zimmer@imp.ac.at).

DOI:10.1162/027064702760161344

Copyright © 2002 Society for Neuroscience 0270-6474/02/224369-09\$15.00/0

Received June 1, 2001; revised April 12, 2002; accepted April 12, 2002.

This work was supported by grants from the Austrian Science Fund FWF (S.K., T.S., S.Z.) and the European Union (S.Z.).

We thank Veronika Müller and Howard Berg, and members of our laboratories for useful discussions.

Correspondence should be addressed to Dr. Manuel Zimmer (E-mail: zimmer@imp.ac.at).

DOI:10.1162/027064702760161344

Copyright © 2002 Society for Neuroscience 0270-6474/02/224369-09\$15.00/0

Received June 1, 2001; revised April 12, 2002; accepted April 12, 2002.

This work was supported by grants from the Austrian Science Fund FWF (S.K., T.S., S.Z.) and the European Union (S.Z.).

We thank Veronika Müller and Howard Berg, and members of our laboratories for useful discussions.

Correspondence should be addressed to Dr. Manuel Zimmer (E-mail: zimmer@imp.ac.at).

DOI:10.1162/027064702760161344

Copyright © 2002 Society for Neuroscience 0270-6474/02/224369-09\$15.00/0

Received June 1, 2001; revised April 12, 2002; accepted April 12, 2002.

This work was supported by grants from the Austrian Science Fund FWF (S.K., T.S., S.Z.) and the European Union (S.Z.).

We thank Veronika Müller and Howard Berg, and members of our laboratories for useful discussions.

Correspondence should be addressed to Dr. Manuel Zimmer (E-mail: zimmer@imp.ac.at).

DOI:10.1162/027064702760161344

Copyright © 2002 Society for Neuroscience 0270-6474/02/224369-09\$15.00/0

Received June 1, 2001; revised April 12, 2002; accepted April 12, 2002.

This work was supported by grants from the Austrian Science Fund FWF (S.K., T.S., S.Z.) and the European Union (S.Z.).

We thank Veronika Müller and Howard Berg, and members of our laboratories for useful discussions.

Correspondence should be addressed to Dr. Manuel Zimmer (E-mail: zimmer@imp.ac.at).

DOI:10.1162/027064702760161344

Copyright © 2002 Society for Neuroscience 0270-6474/02/224369-09\$15.00/0

Received June 1, 2001; revised April 12, 2002; accepted April 12, 2002.

This work was supported by grants from the Austrian Science Fund FWF (S.K., T.S., S.Z.) and the European Union (S.Z.).

We thank Veronika Müller and Howard Berg, and members of our laboratories for useful discussions.

Correspondence should be addressed to Dr. Manuel Zimmer (E-mail: zimmer@imp.ac.at).

DOI:10.1162/027064702760161344

Copyright © 2002 Society for Neuroscience 0270-6474/02/224369-09\$15.00/0

Received June 1, 2001; revised April 12, 2002; accepted April 12, 2002.

This work was supported by grants from the Austrian Science Fund FWF (S.K., T.S., S.Z.) and the European Union (S.Z.).

We thank Veronika Müller and Howard Berg, and members of our laboratories for useful discussions.

Correspondence should be addressed to Dr. Manuel Zimmer (E-mail: zimmer@imp.ac.at).

DOI:10.1162/027064702760161344

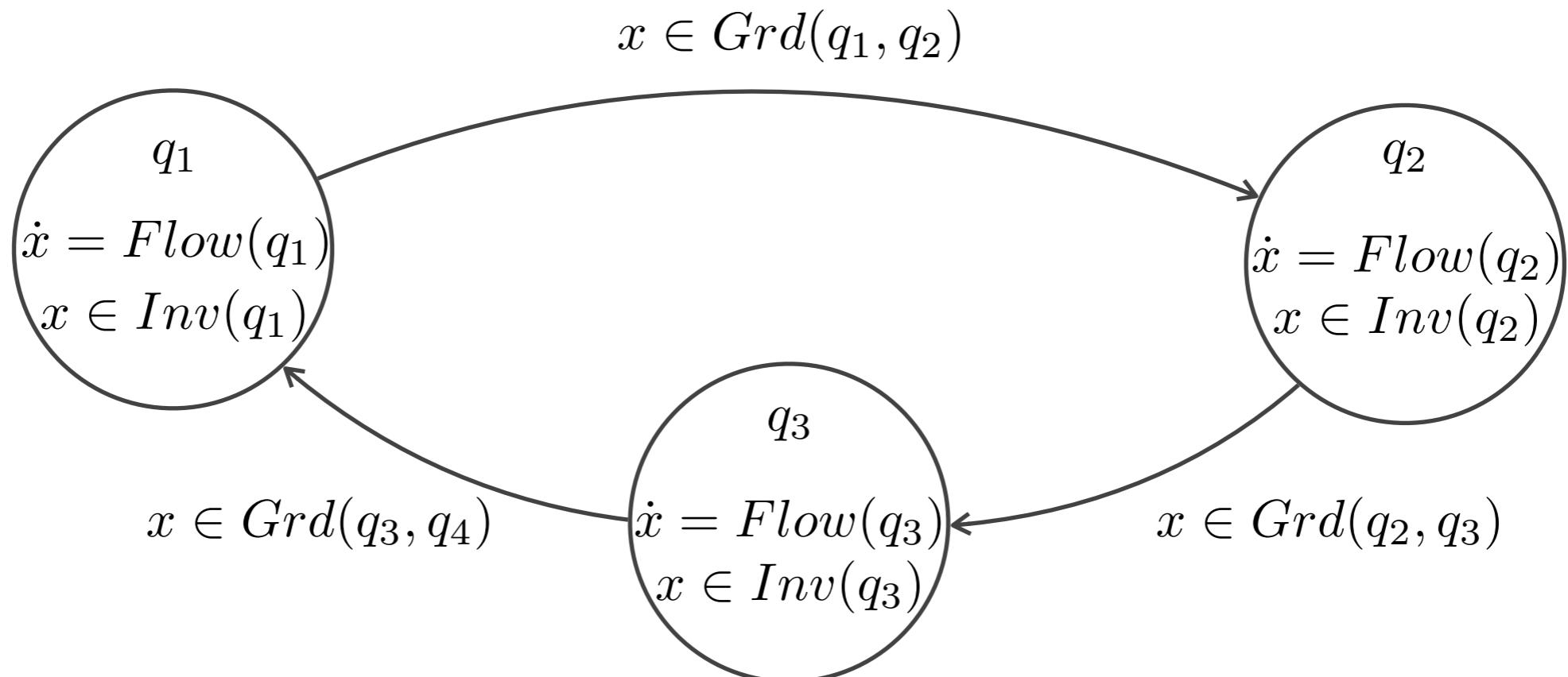
Copyright © 2002 Society for Neuroscience 0270-6474/02/224369-09\$15.00/0

Received June 1, 2001;

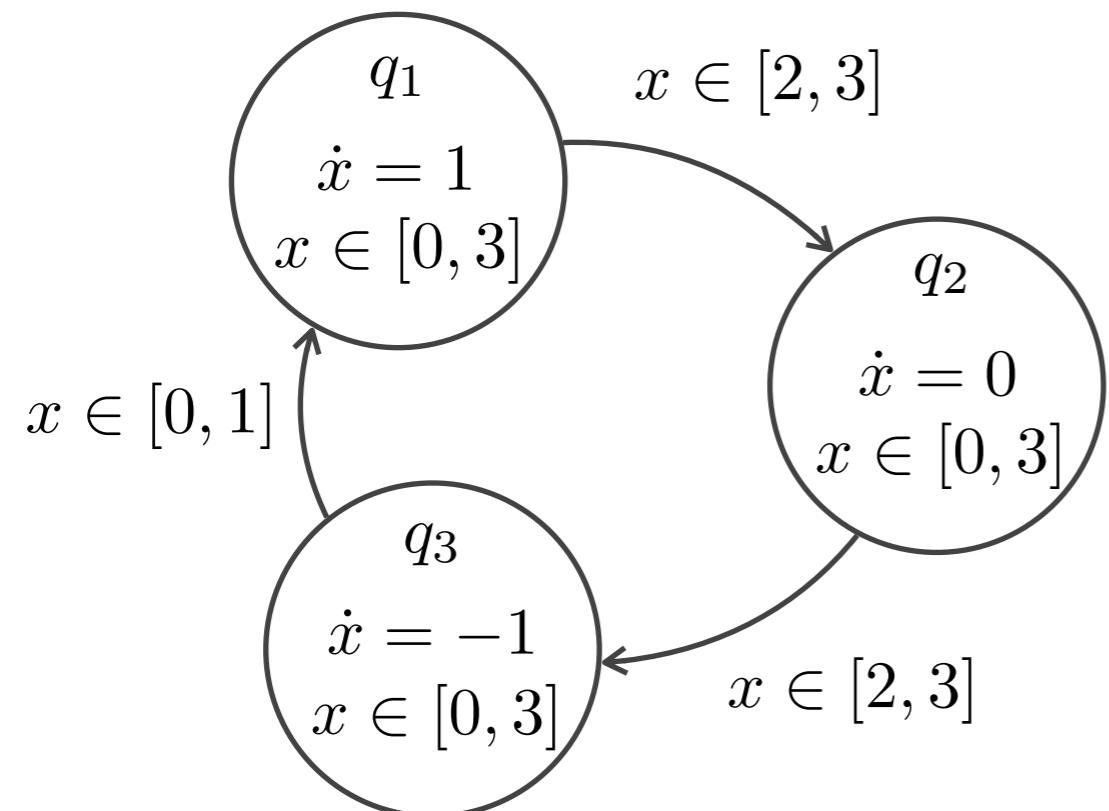
Linear Hybrid Automaton (LHA)

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

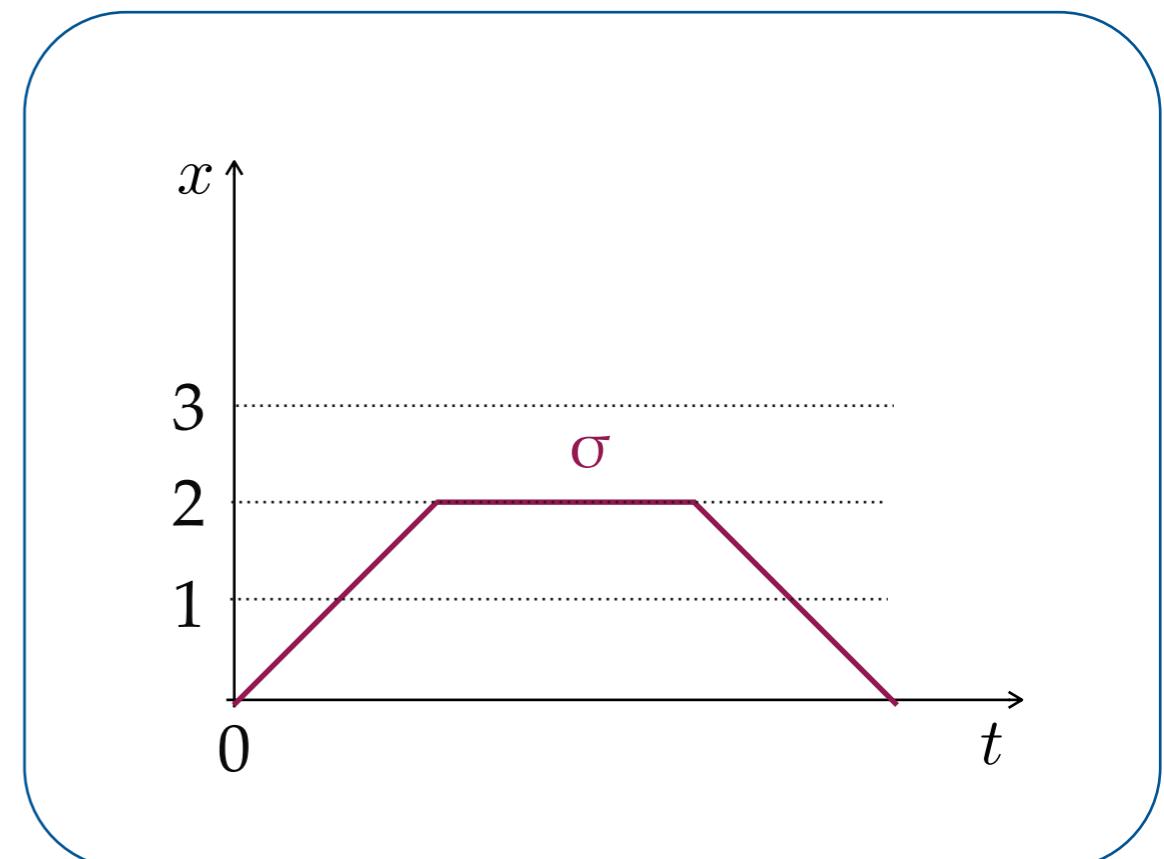
- * Q is a finite set of modes,
- * $E \subseteq Q \times Q$ is a transition relation,
- * $\text{Flow}: Q \rightarrow \mathbb{R}^n$ is the flow function,
- * $\text{Inv}: Q \rightarrow \text{cpoly}(n)$ is the invariant function, and
- * $\text{Grd}: E \rightarrow \text{cpoly}(n)$ is the guard function,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



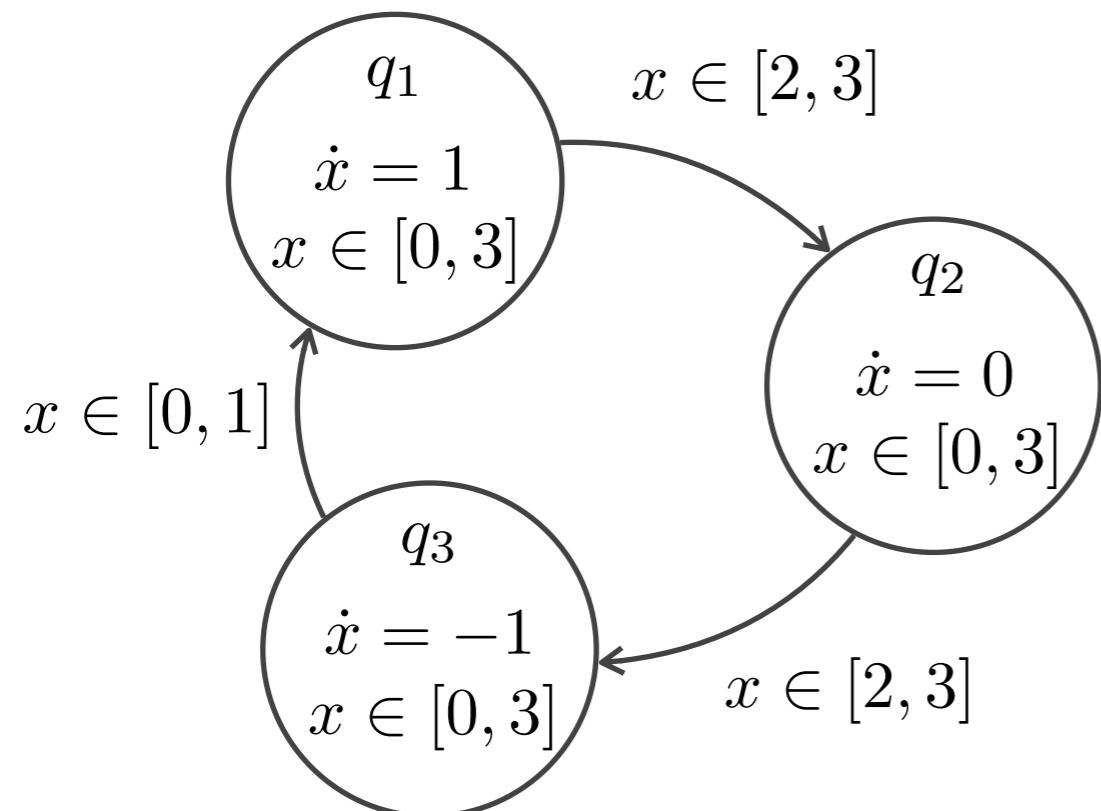
LHA sample



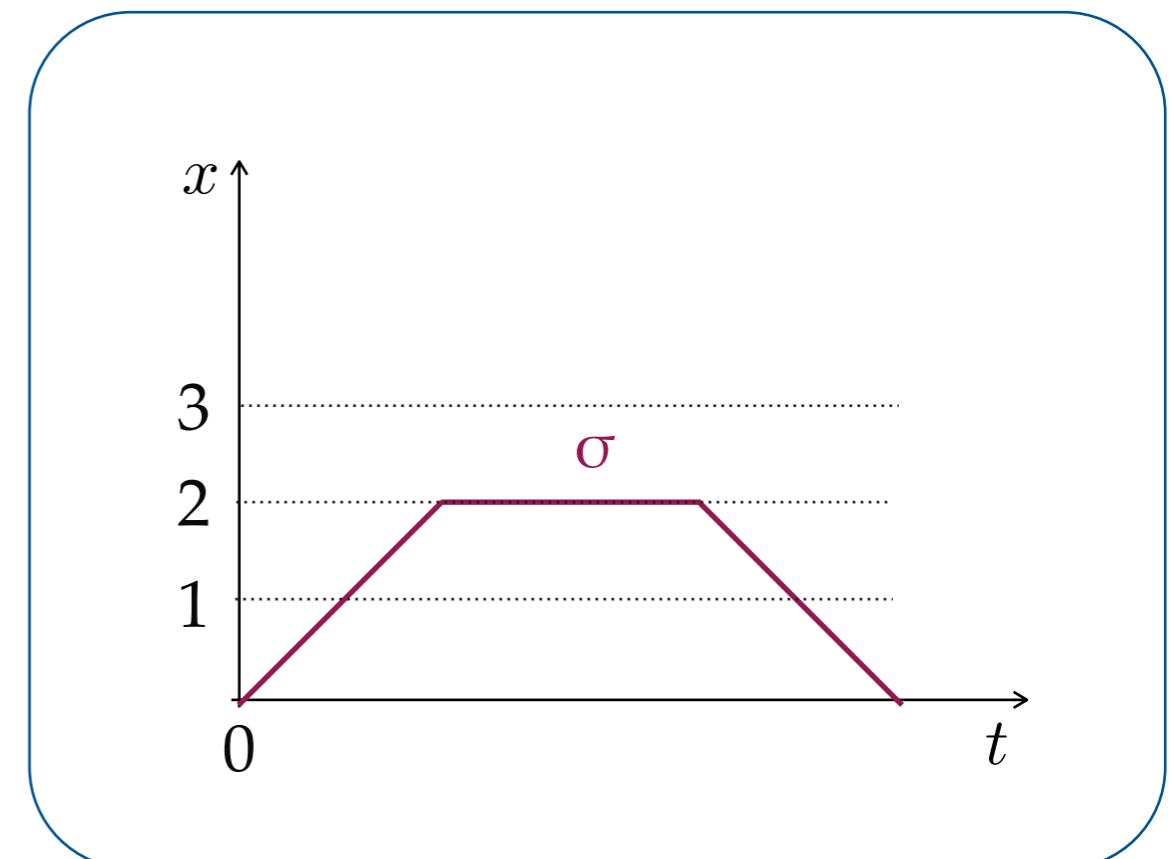
Execution



LHA sample

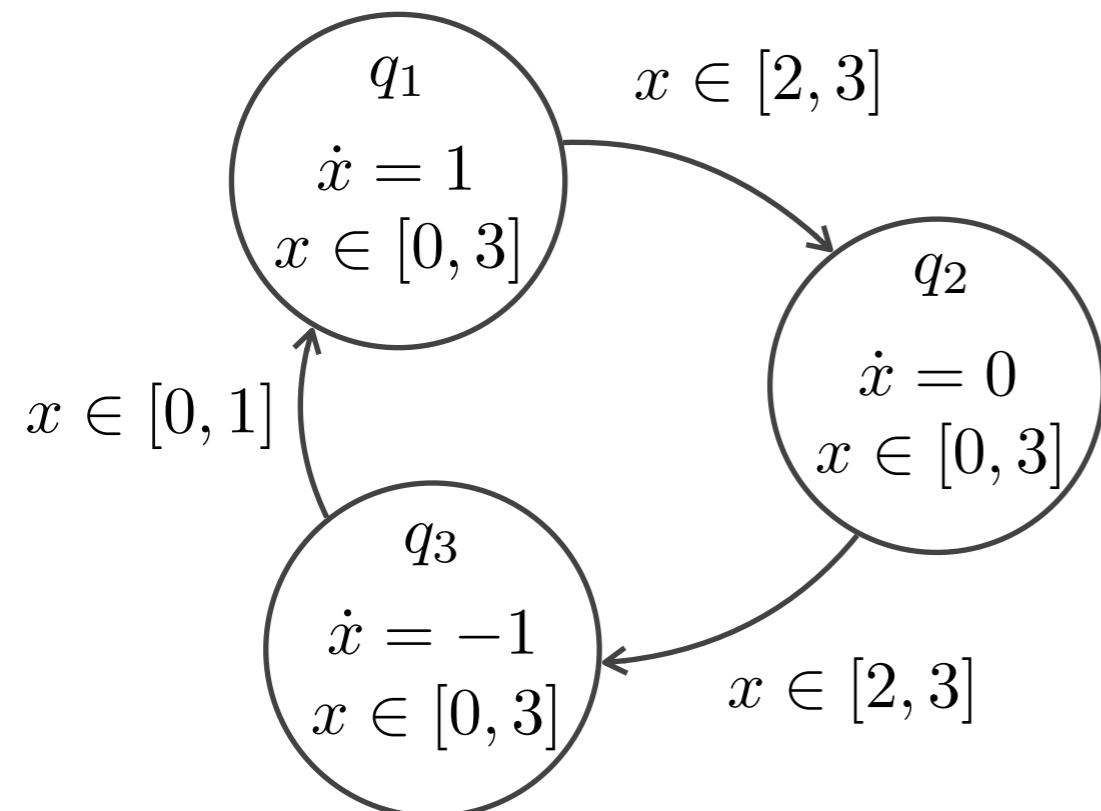


Execution

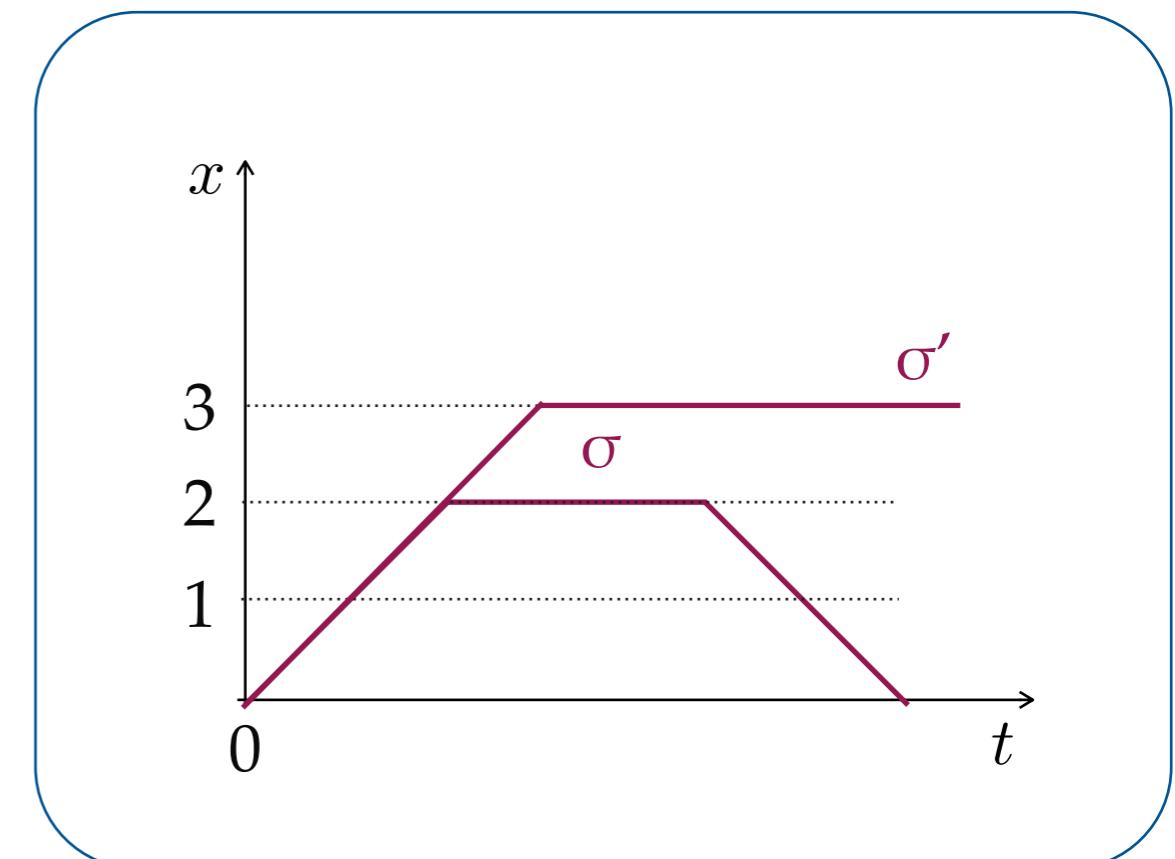


- We consider non deterministic linear hybrid automata
- The LHA features piecewise-linear executions

LHA sample



Execution

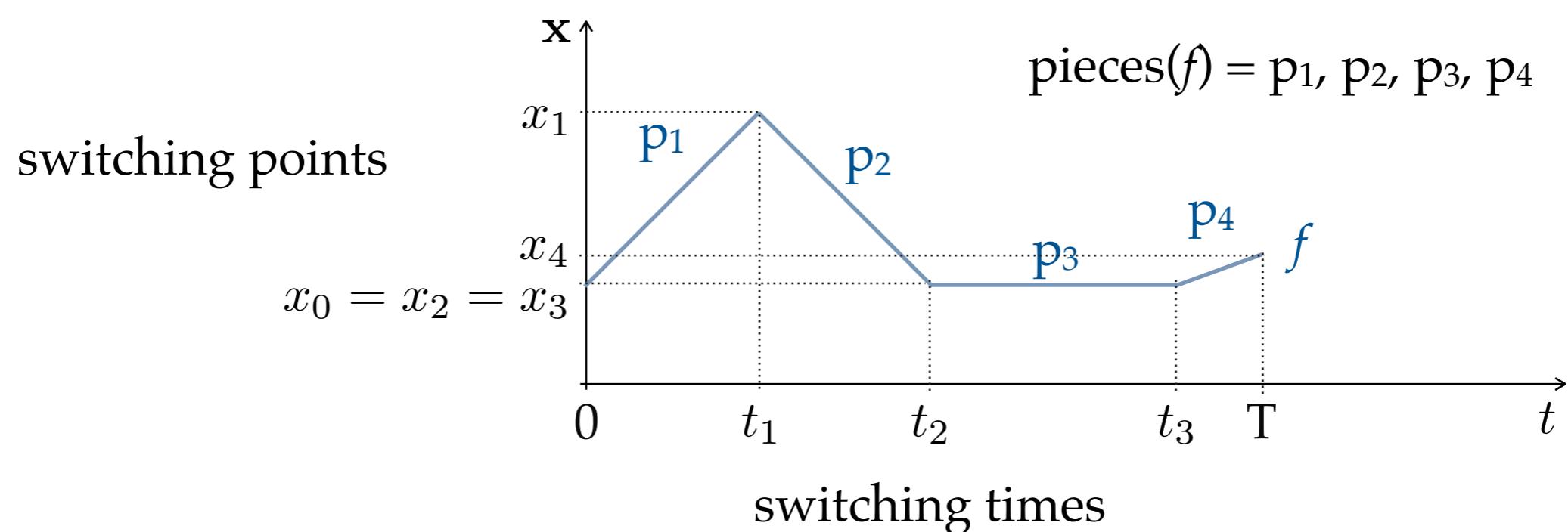


- We consider non deterministic linear hybrid automata
- The LHA features piecewise-linear executions

Piecewise-linear function

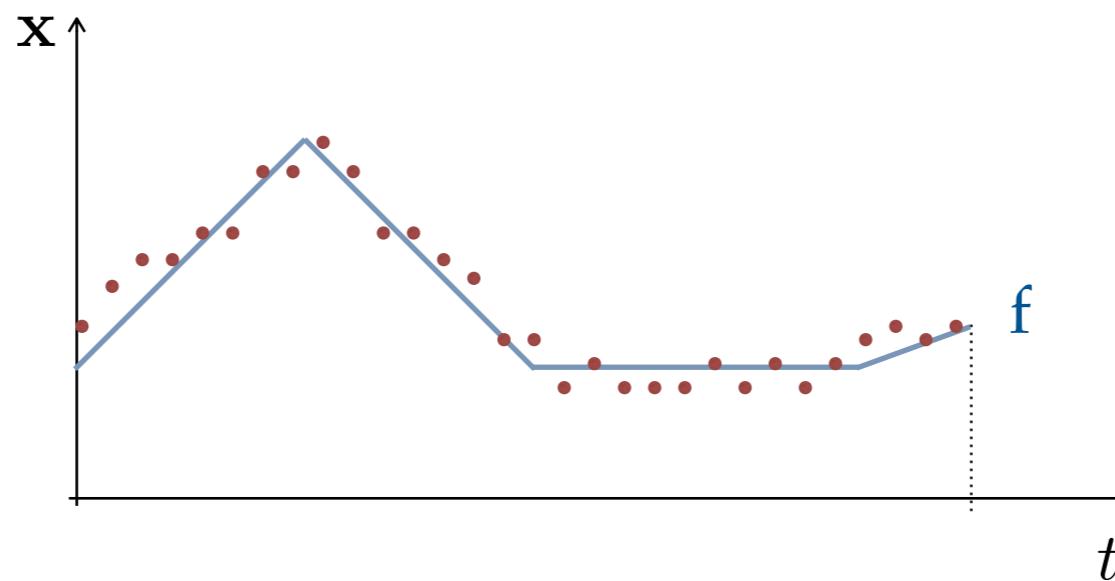
Definition. $f: [0, T] \rightarrow \mathbb{R}^n$ is an **m-piecewise-linear (m-PWL) function** if

- $f \equiv p_1, p_2, \dots, p_m$ 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

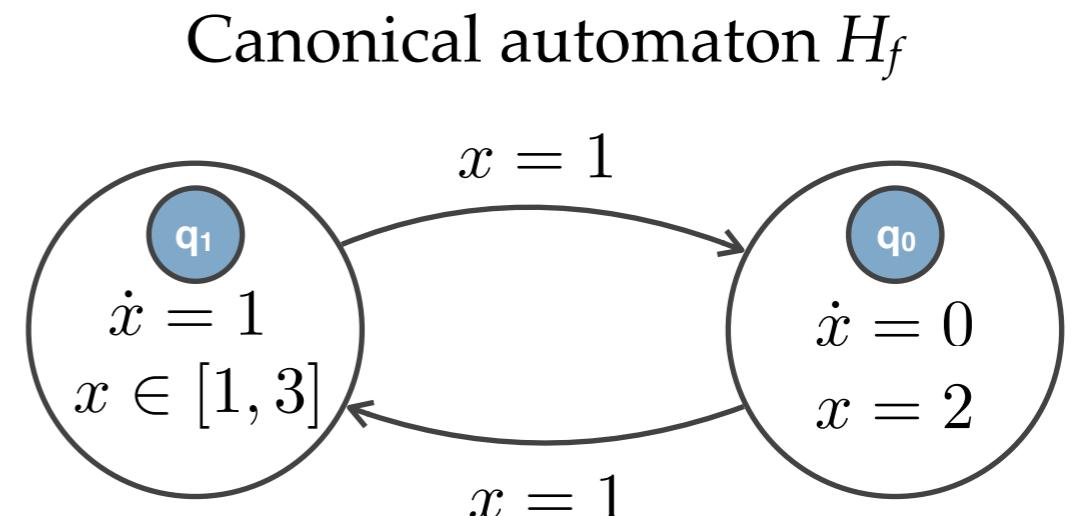
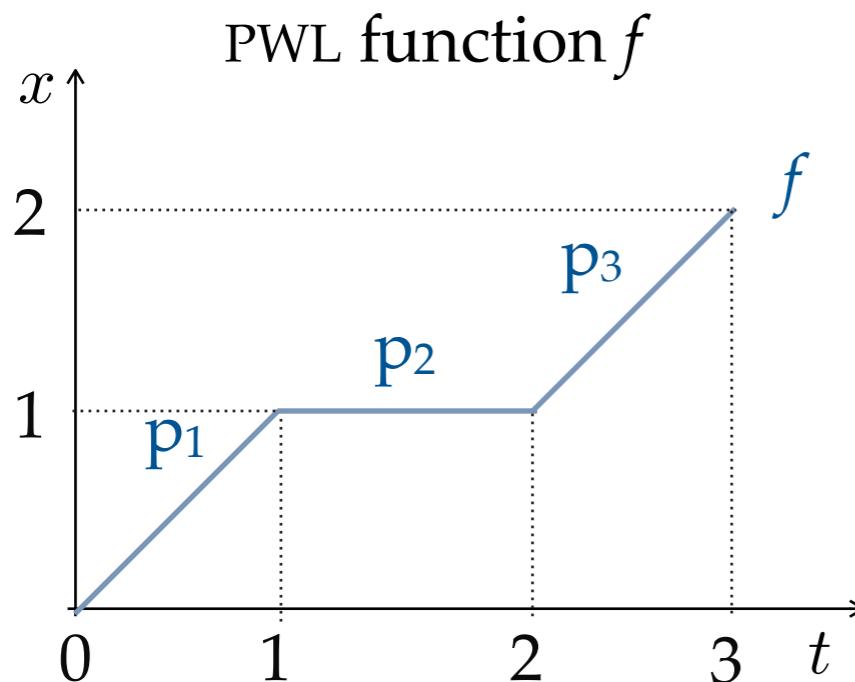


- Douglas-Peucker line simplification algorithm
- Linear regression
- Hakimi-Schmeichel algorithm

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
- $\text{Grd}(q_a, q_{a'}) = \text{convex_hull} (\{\text{end_point}(p) : \exists p, p' \in \text{pieces}(f) \text{ adjacent, } \text{slope}(p) = a, \text{slope}(p') = a'\})$

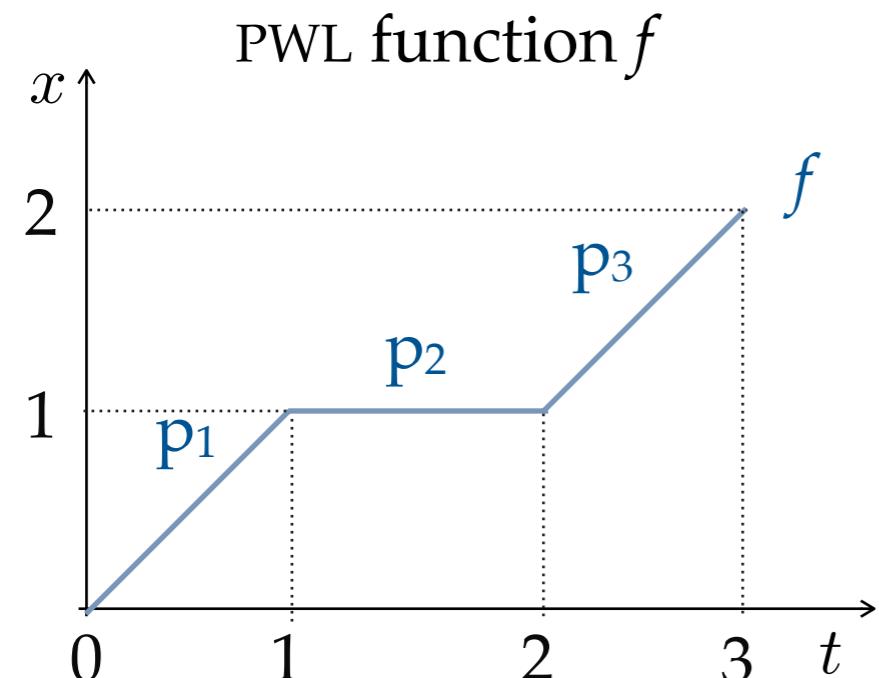


$$\text{slope}(p_1) = 1, \text{slope}(p_2) = 0, \text{slope}(p_3) = 1$$

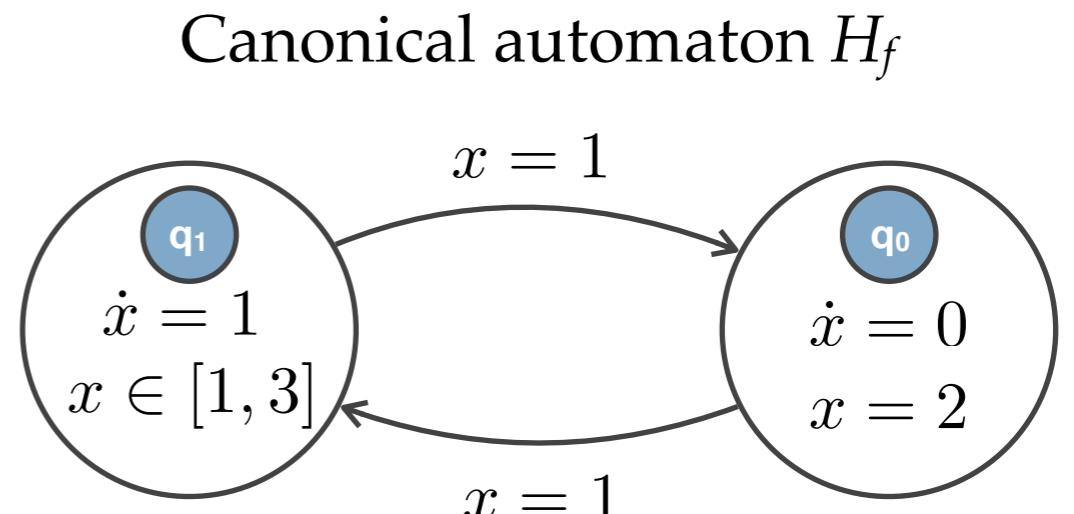
Canonical linear hybrid automaton

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
- $\text{Grd}(q_a, q_{a'}) = \text{convex_hull} (\{\text{end_point}(p) : \exists p, p' \in \text{pieces}(f) \text{ adjacent, } \text{slope}(p) = a, \text{slope}(p') = a'\})$



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



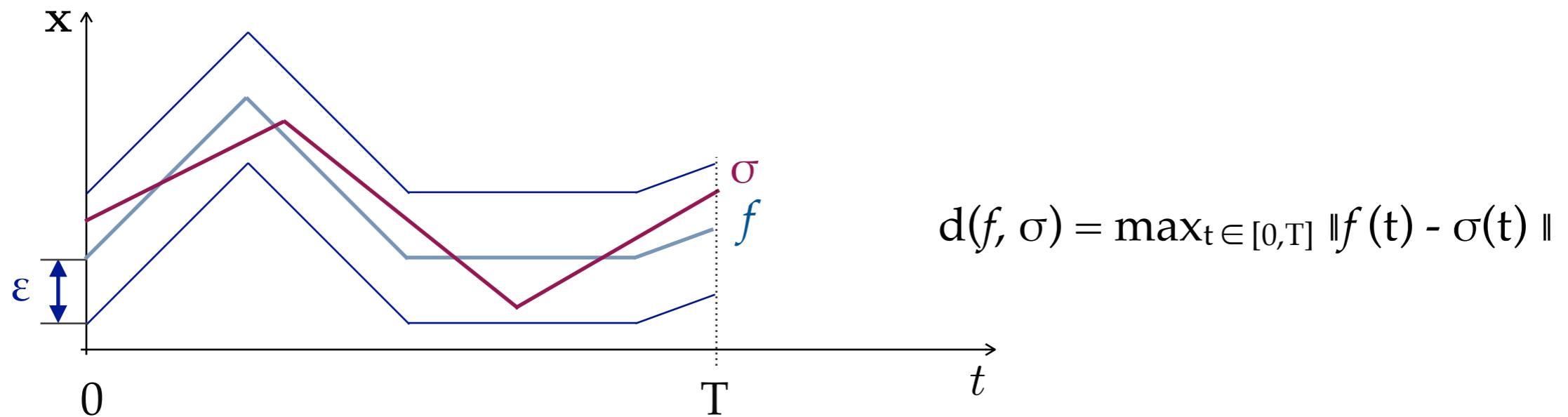
Given a PWL function f ,
 f is an execution of H_f

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 $d(f, \sigma) \leq \varepsilon$.



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

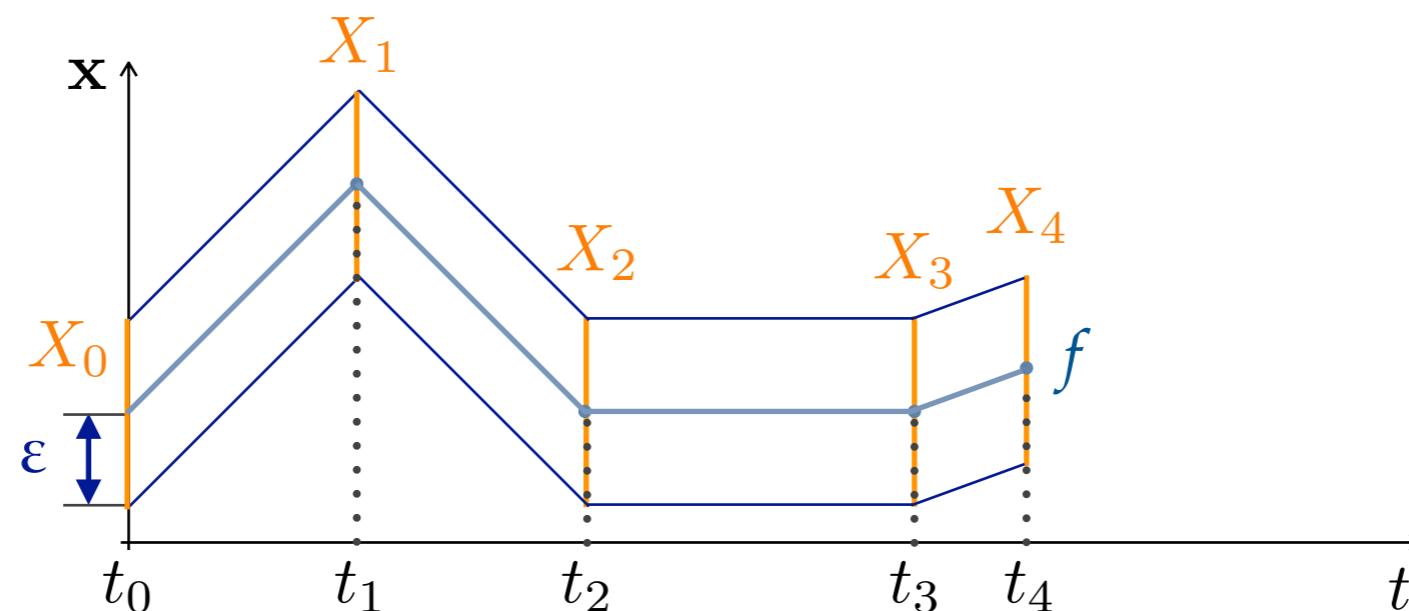
Asynchronous specification

Synchronous specification

Synchronous specification

Specification

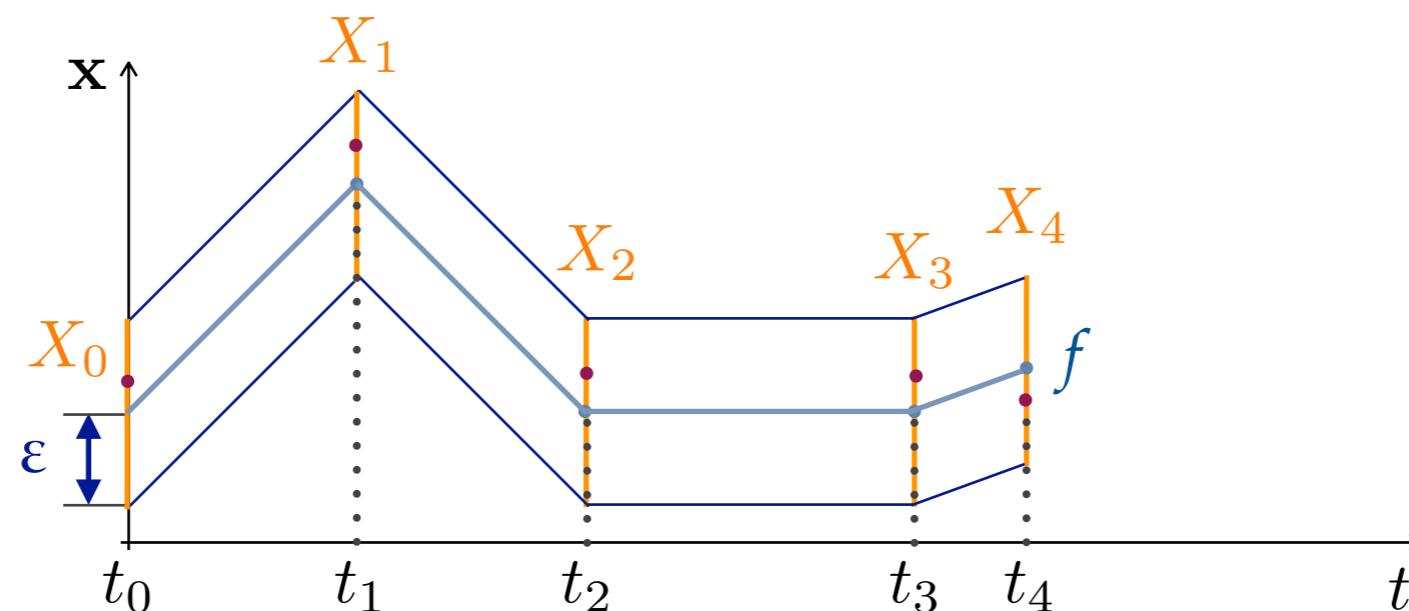
- $H \varepsilon$ -captures every function f in F
- H switches synchronously with the functions in F



Synchronous specification

Specification

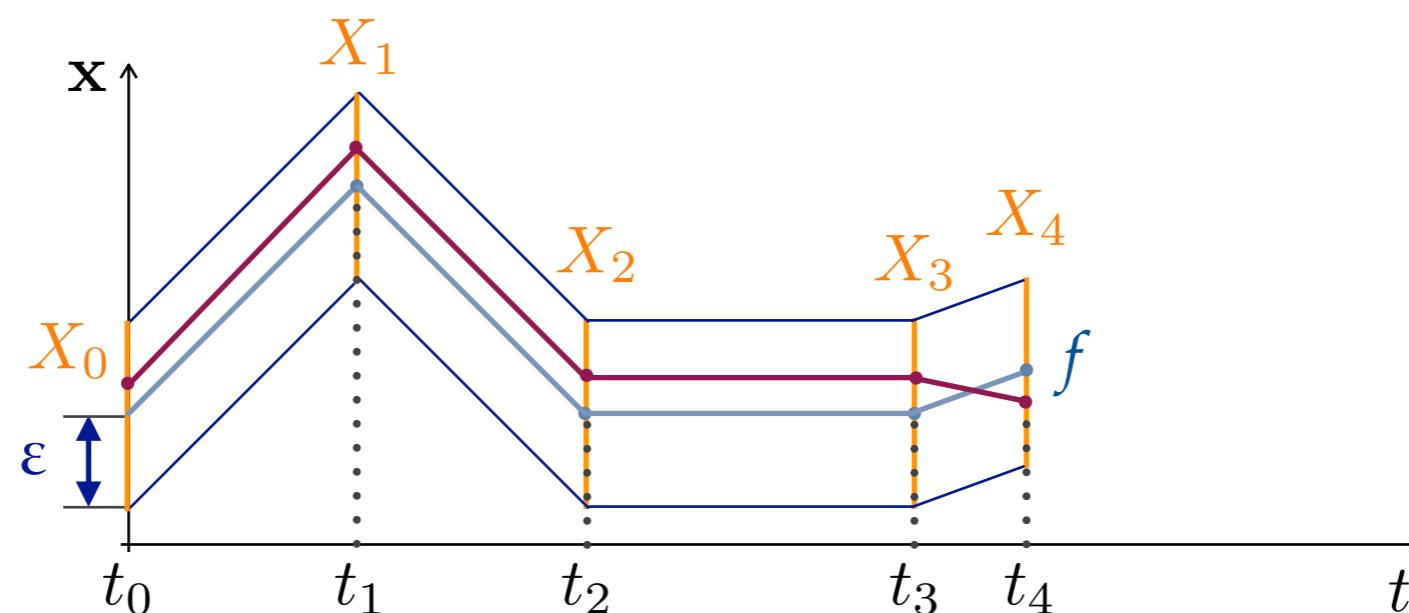
- $H \varepsilon$ -captures every function f in F
- H switches synchronously with the functions in F



Synchronous specification

Specification

- $H \varepsilon$ -captures every function f in F
- H switches synchronously with the functions in F

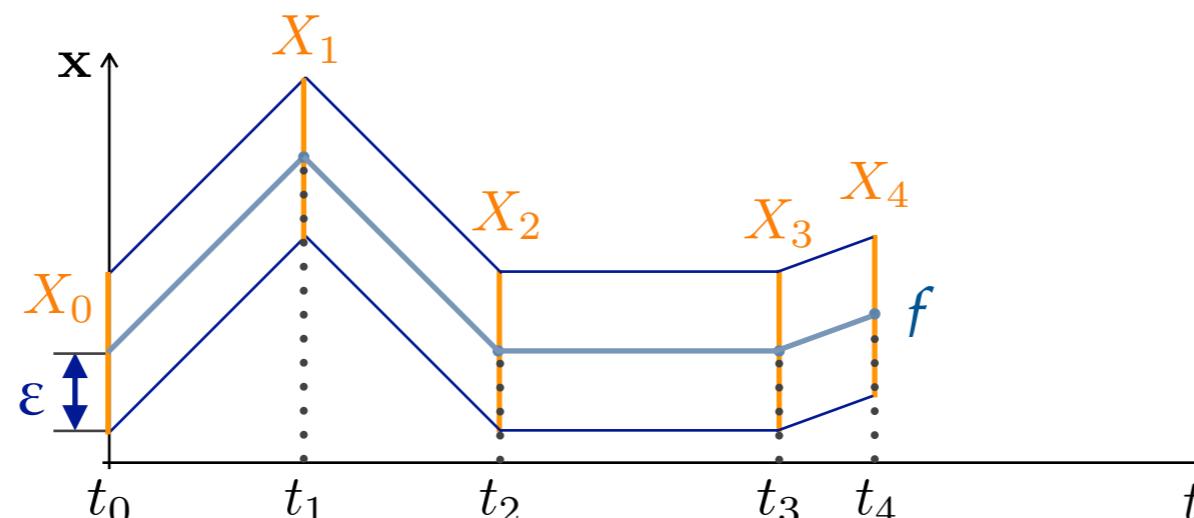


SMT-based synthesis approach

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 , $x_0 = f(t_0), \dots, x_m = f(t_m) \rightarrow$ Expressed as $y_j \in X_j$
 - $b_1 = \text{slope}(q_1), \dots, b_m = \text{slope}(q_m)$ with ℓ different values (c_1, \dots, c_ℓ)

$$\phi_{f,\varepsilon}(\ell) \equiv \bigwedge_{j=1}^m y_j = y_{j-1} + b_j(t_j - t_{j-1}) \wedge \bigwedge_{j=0}^m y_j \in X_j \wedge \bigwedge_{j=1}^m \bigvee_{k=1}^\ell b_j = c_k$$

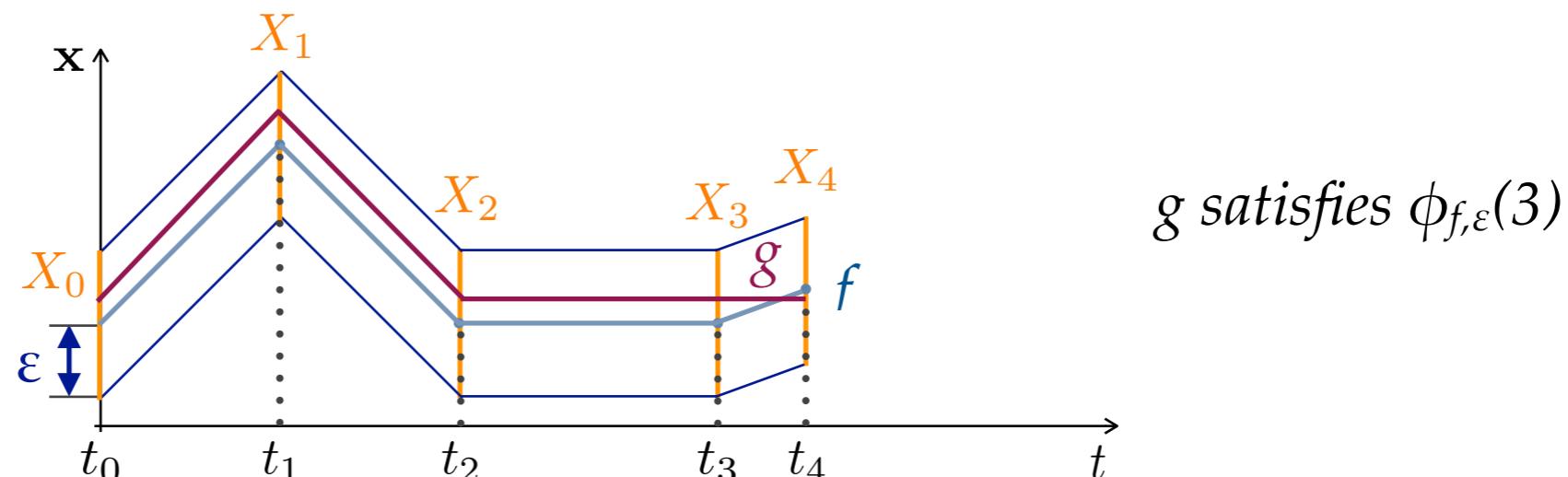


SMT-based synthesis approach

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 , $x_0 = f(t_0), \dots, x_m = f(t_m) \rightarrow$ Expressed as $y_j \in X_j$
 - $b_1 = \text{slope}(q_1), \dots, b_m = \text{slope}(q_m)$ with ℓ different values (c_1, \dots, c_ℓ)

$$\phi_{f,\varepsilon}(\ell) \equiv \bigwedge_{j=1}^m y_j = y_{j-1} + b_j(t_j - t_{j-1}) \wedge \bigwedge_{j=0}^m y_j \in X_j \wedge \bigwedge_{j=1}^m \bigvee_{k=1}^\ell b_j = c_k$$



SMT-based synthesis approach

SMT-based synthesis approach

- Global linear arithmetic formula for F

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

- Minimization of the number of different slopes

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

- Canonical automaton of G

$$\Rightarrow H_G$$

SMT-based synthesis approach

SMT-based synthesis approach

- Global linear arithmetic formula for F

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

- Minimization of the number of different slopes

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

- 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

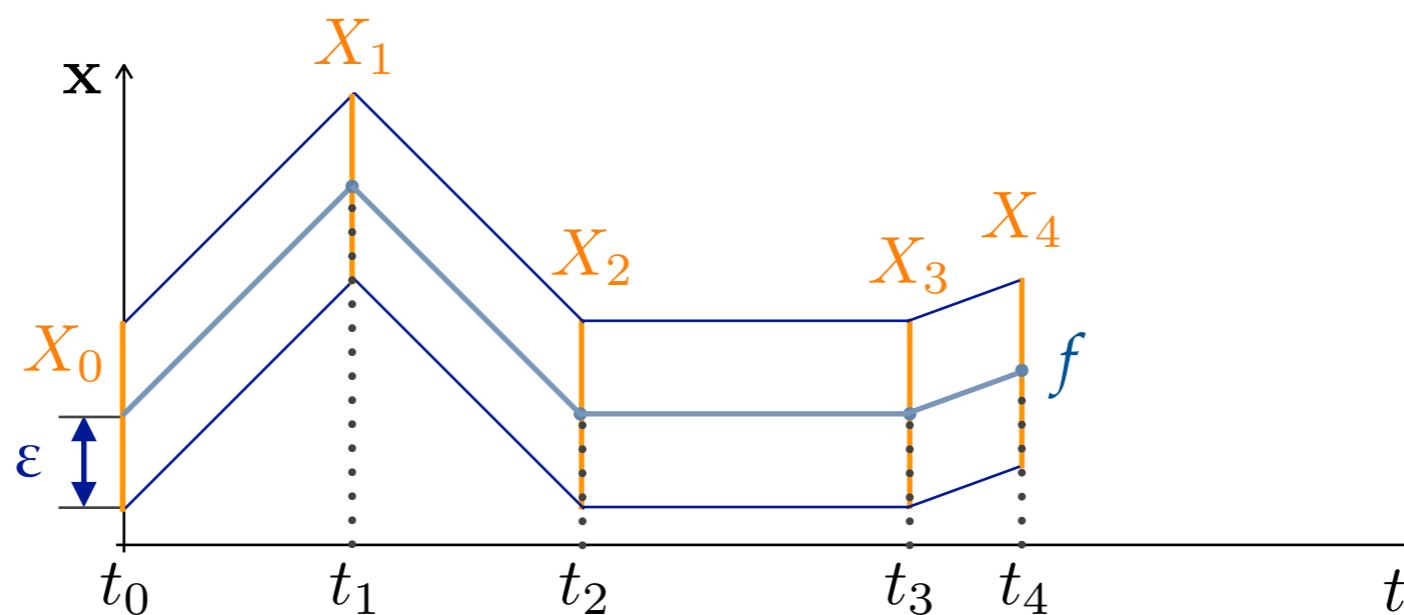
Membership-based synthesis approach

Asynchronous specification

Asynchronous specification

Specification

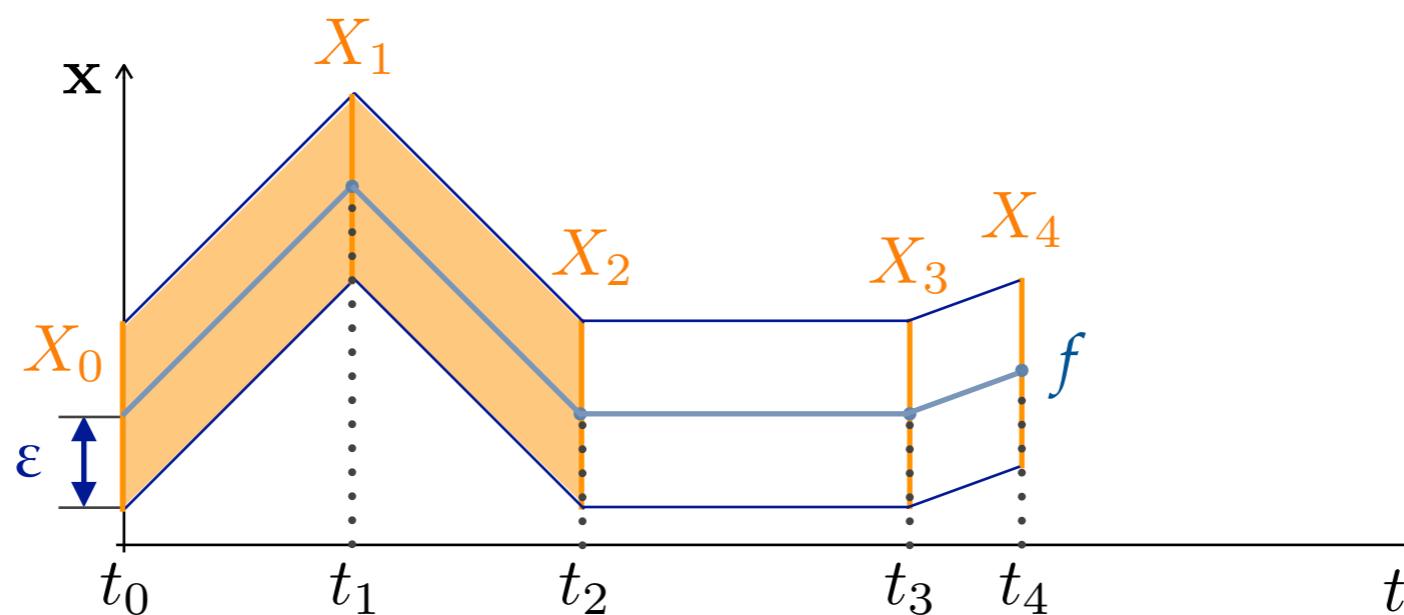
- $H \varepsilon$ -captures every function f in F
- H switches in the time interval determined by the previous and posterior time instants in the functions f in F



Asynchronous specification

Specification

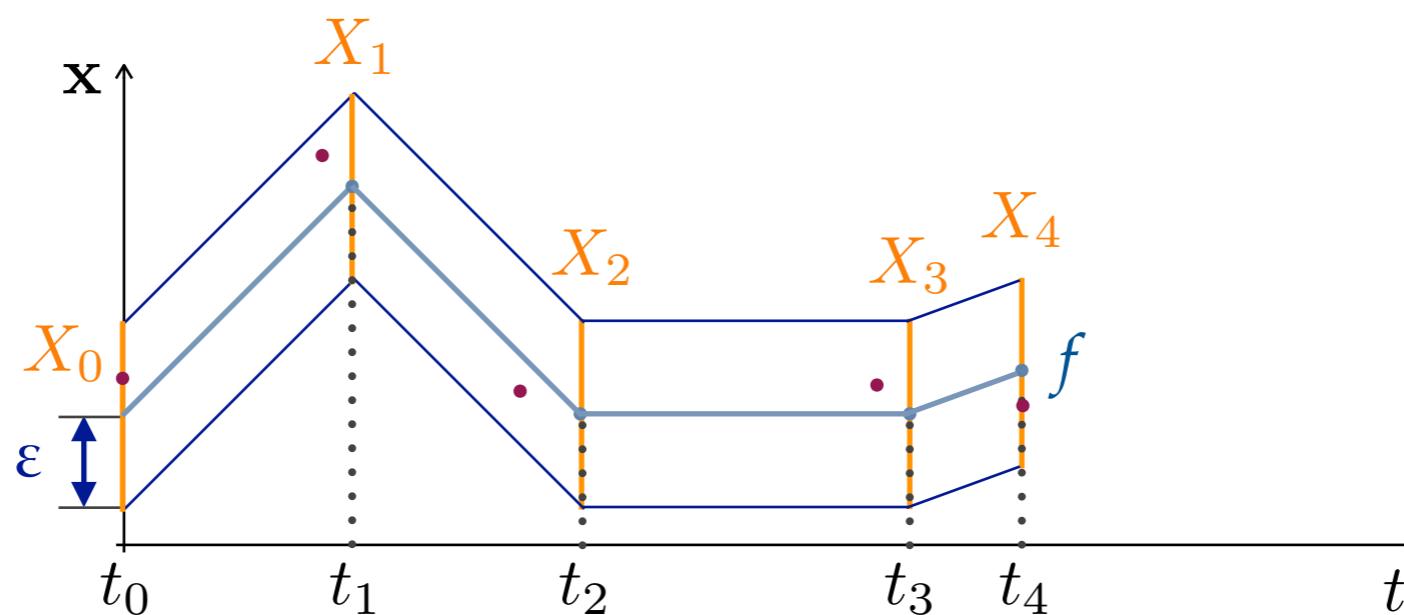
- $H \varepsilon$ -captures every function f in F
- H switches in the time interval determined by the previous and posterior time instants in the functions f in F



Asynchronous specification

Specification

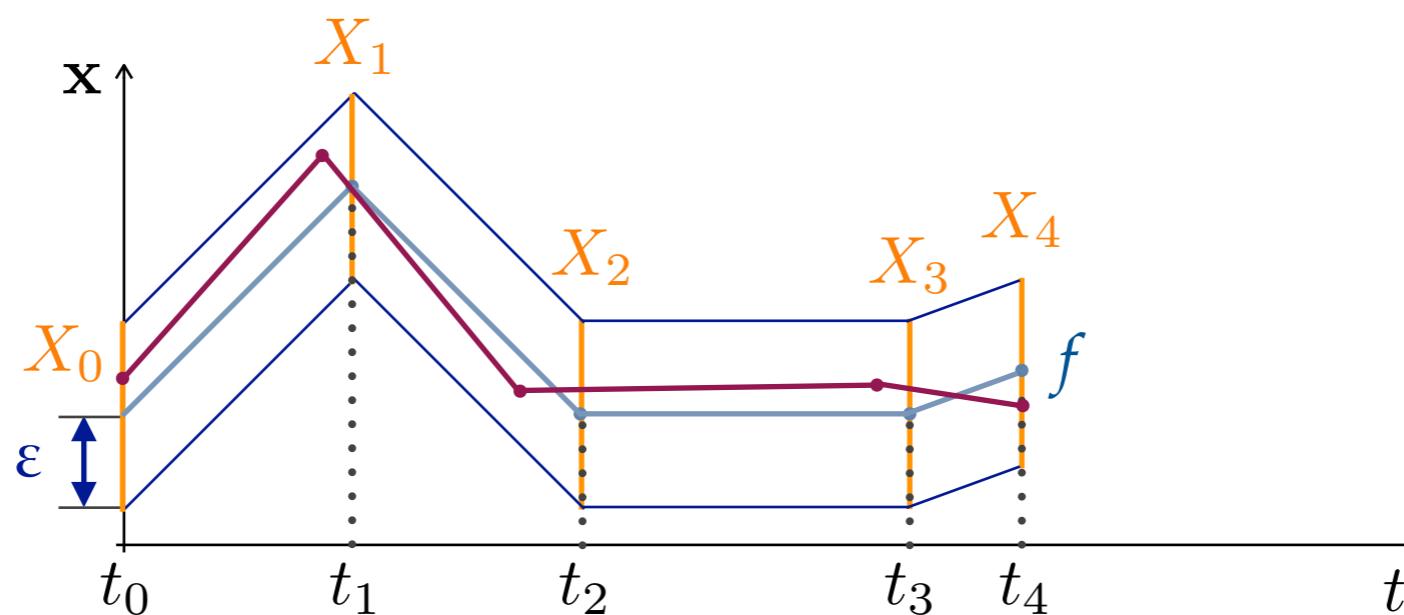
- $H \varepsilon$ -captures every function f in F
- H switches in the time interval determined by the previous and posterior time instants in the functions f in F



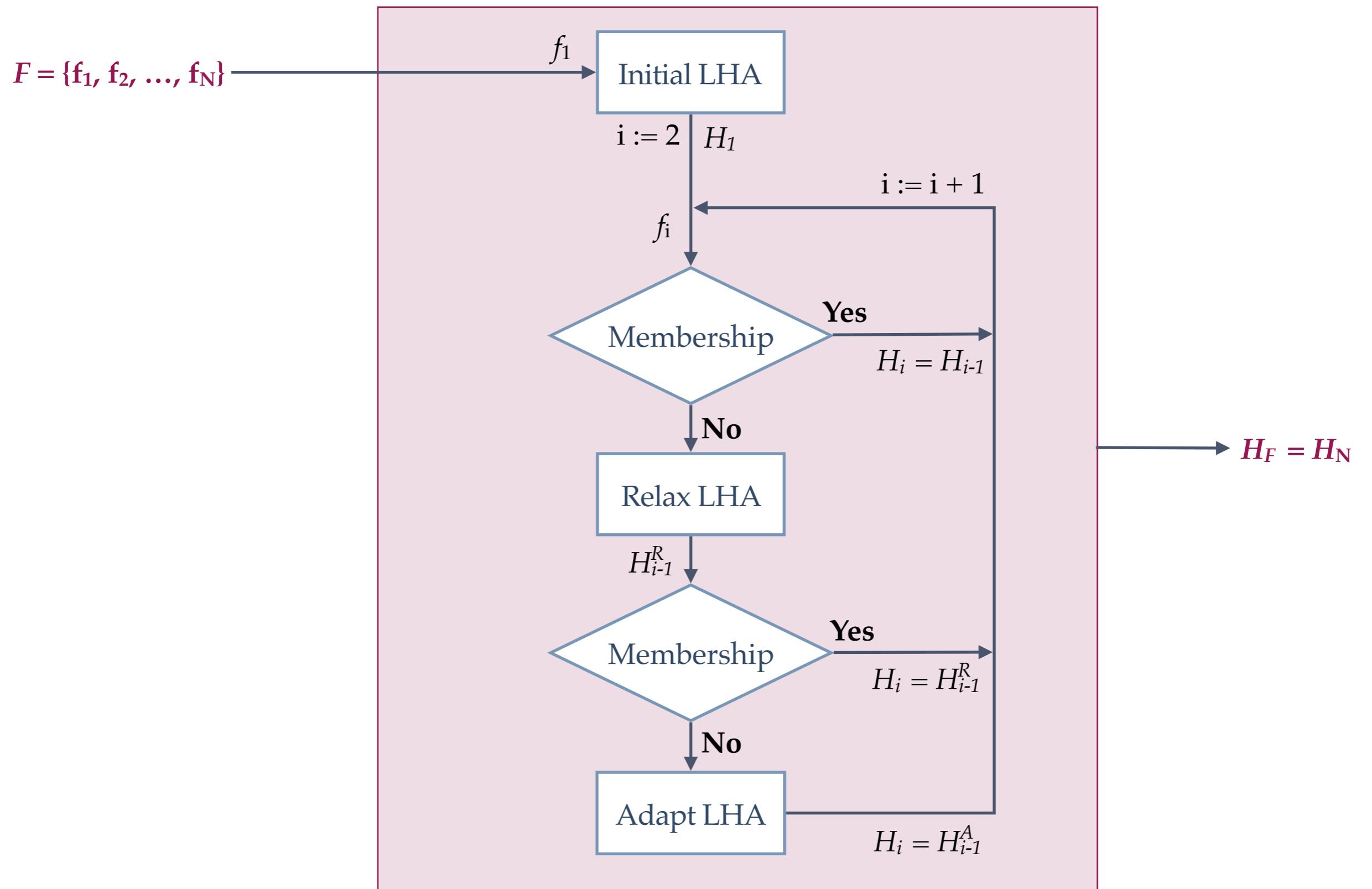
Asynchronous specification

Specification

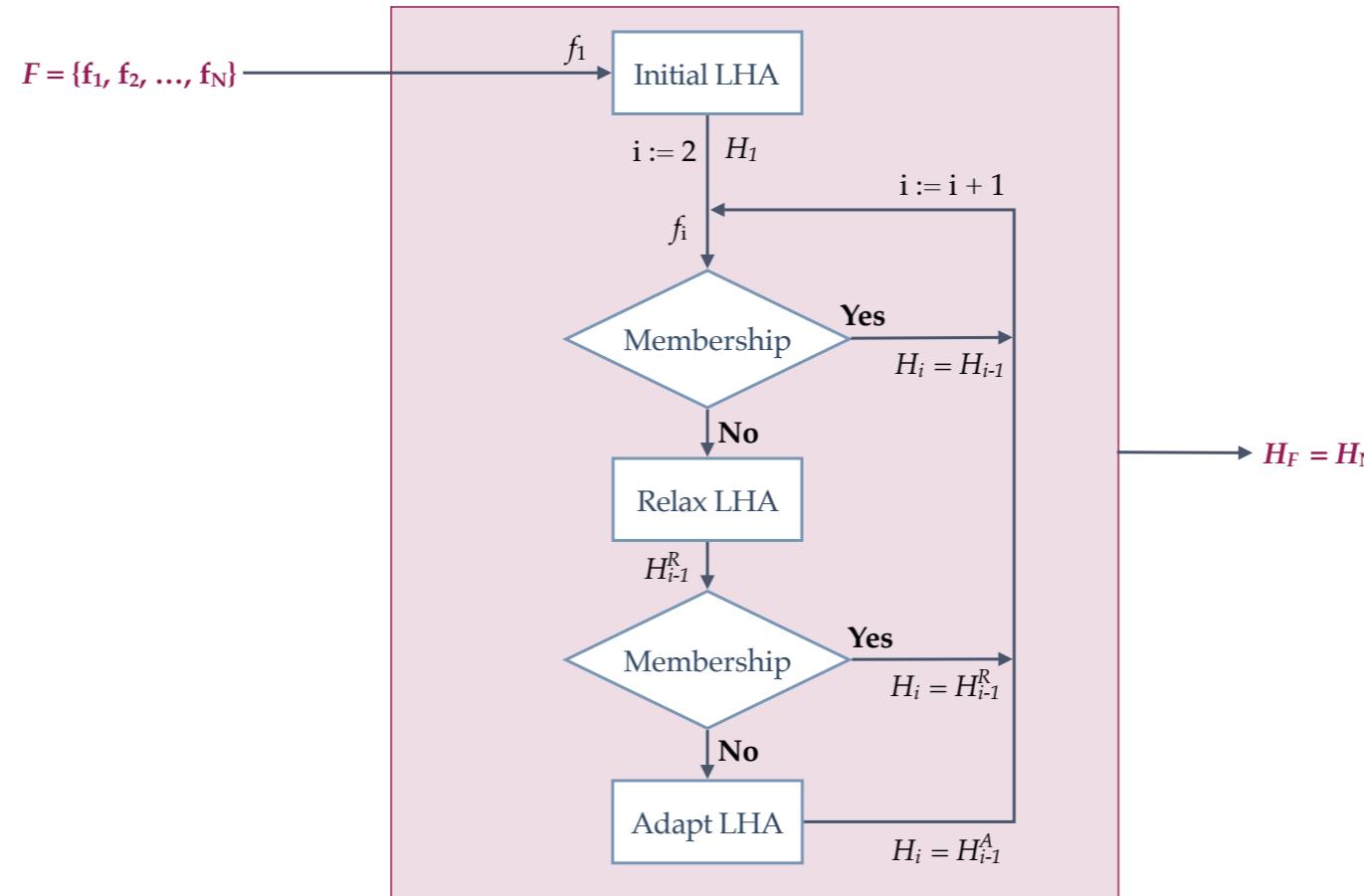
- $H \varepsilon$ -captures every function f in F
- H switches in the time interval determined by the previous and posterior time instants in the functions f in F



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 $t_i \in \text{domain}(p_i \cup p_{i+1})$

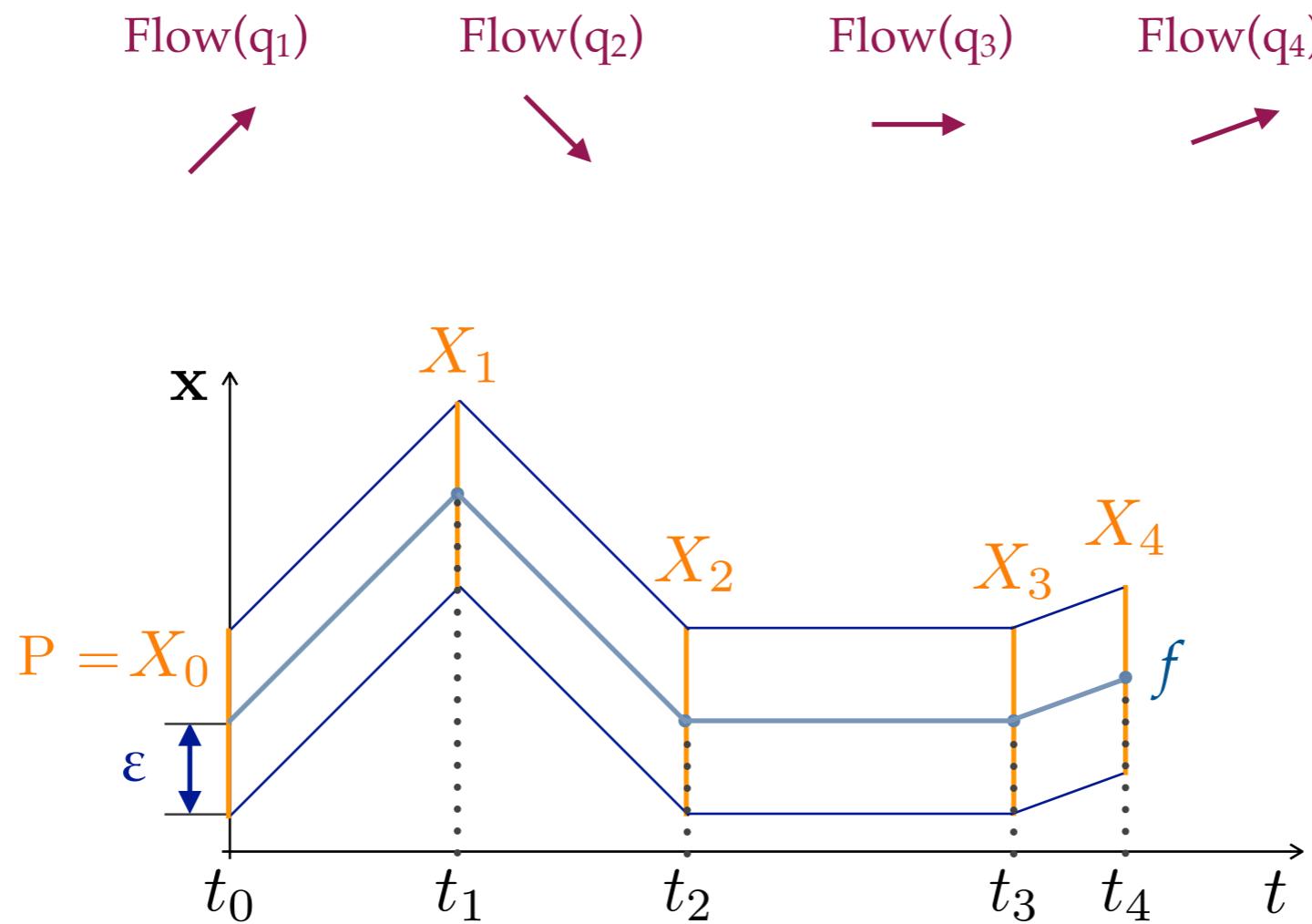
Membership procedure

- 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

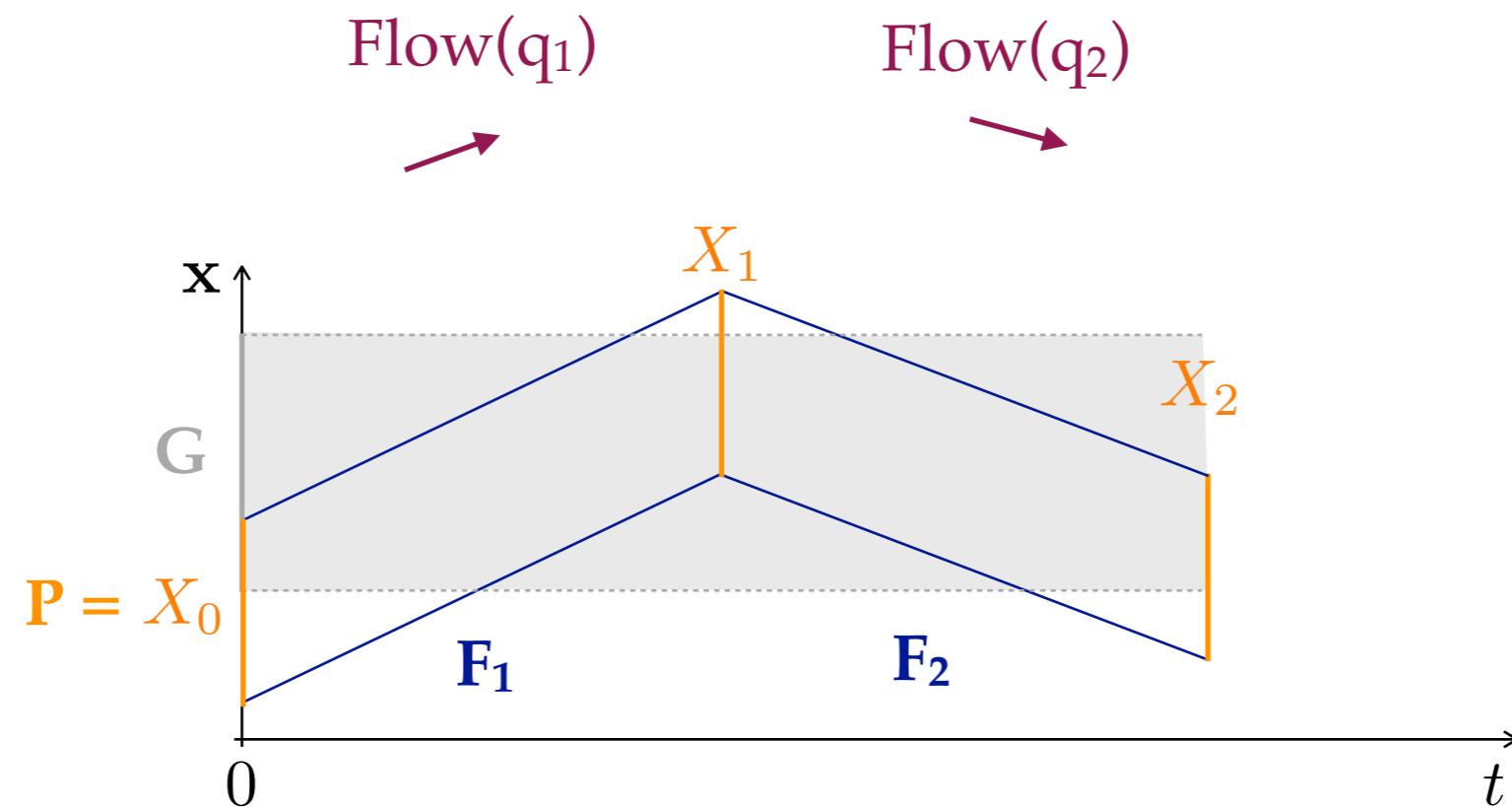
- Linear hybrid automaton H and PWL function f
- $\pi \equiv q_1, q_2, q_3, q_4$ mode path in H



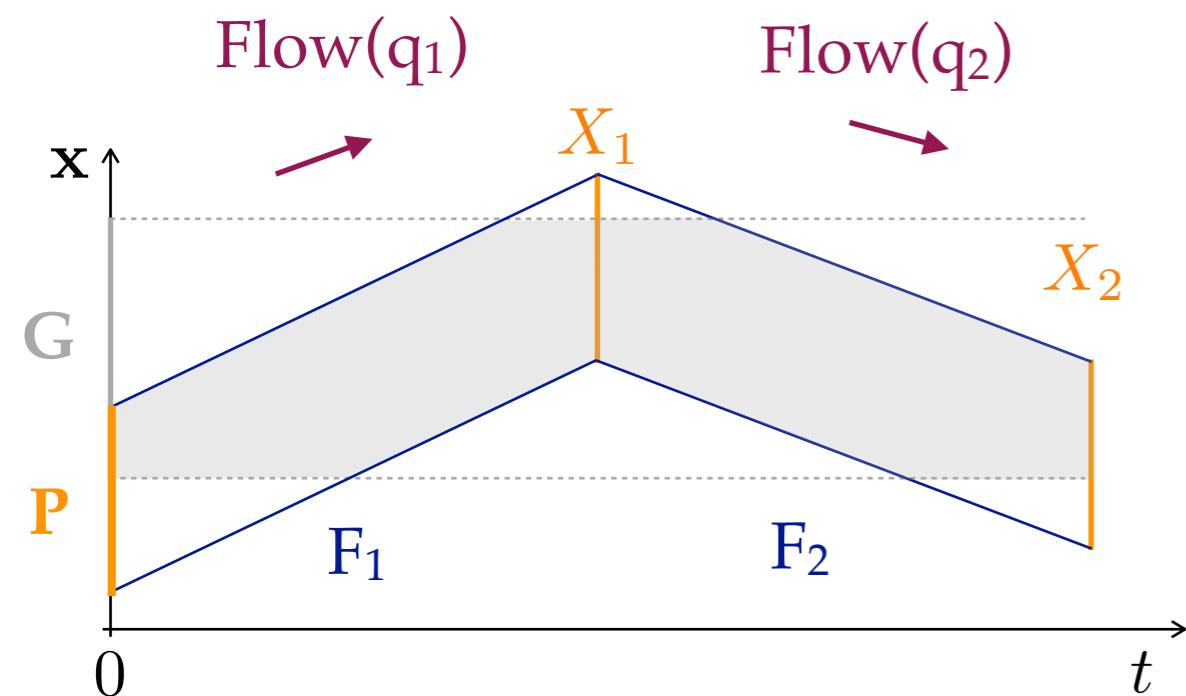
Reachability computation

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

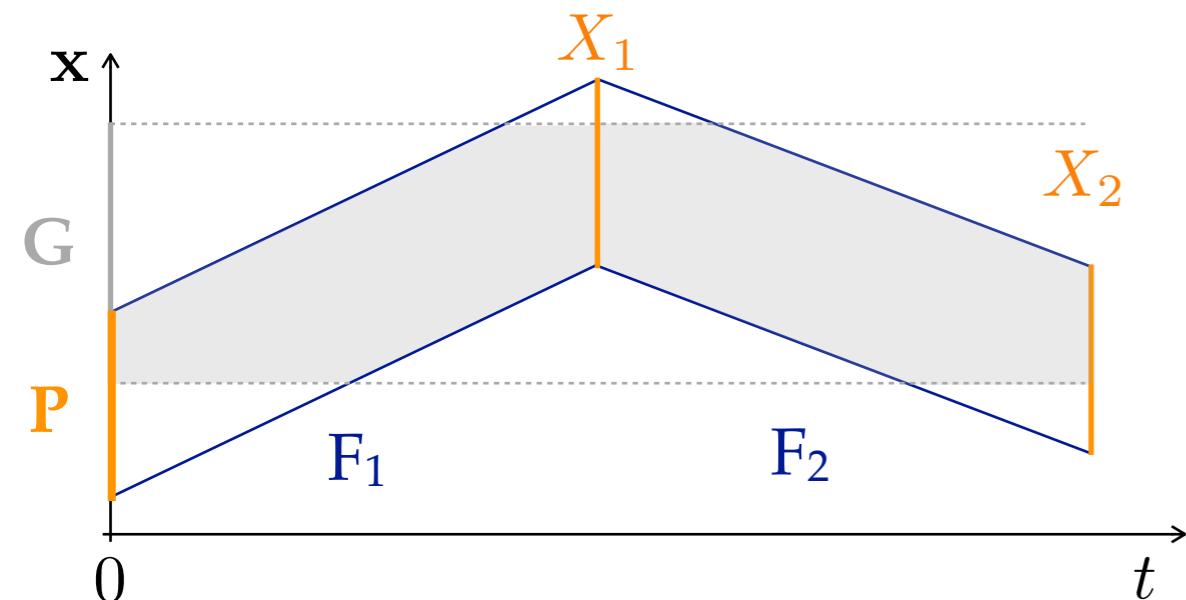


$$G_1 = F_1 \cap G$$

$$G_2 = F_2 \cap G$$

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

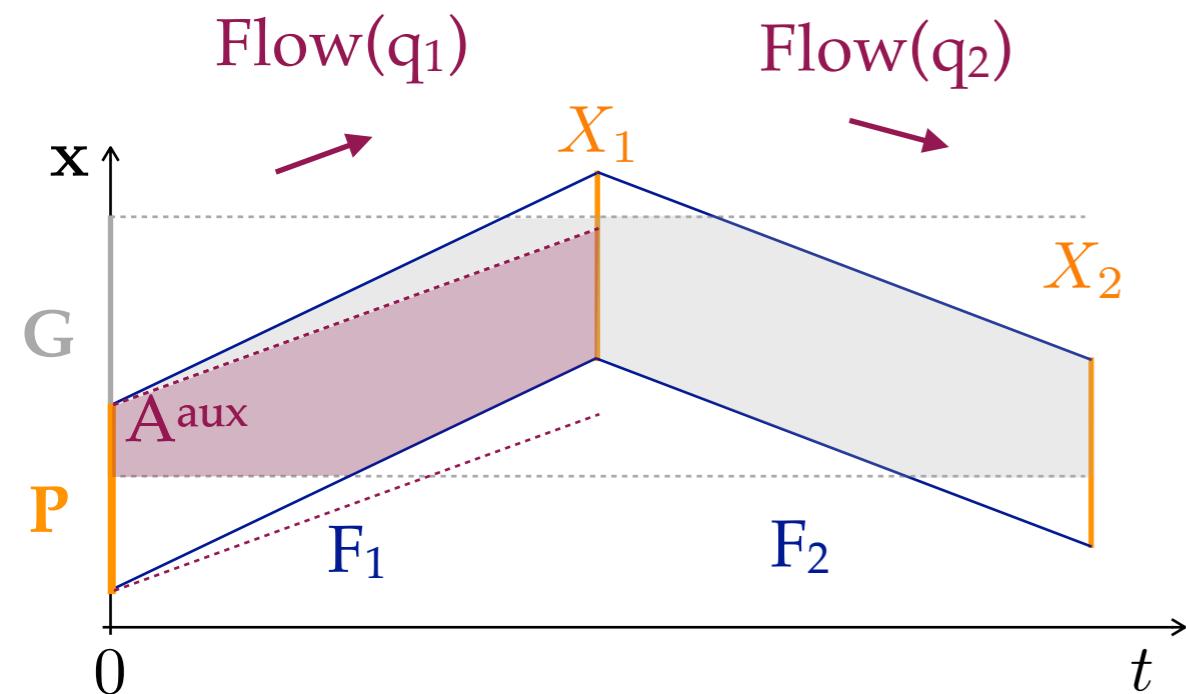
$A = \text{PRE}(X_1, A^{\text{aux}}, \text{Flow}(q_2))$



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

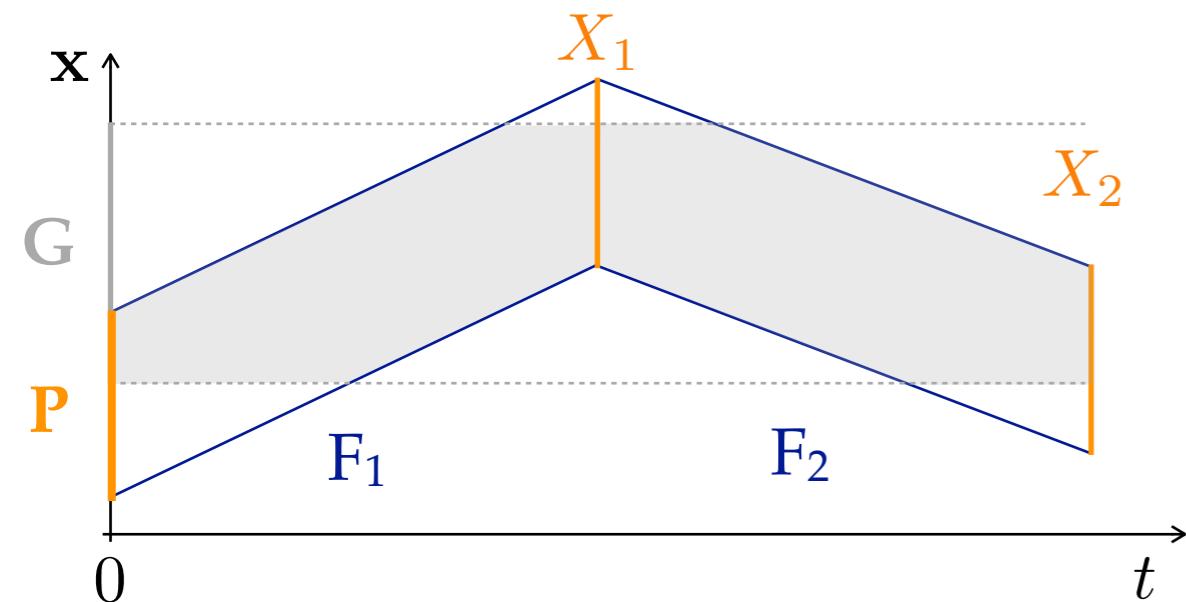
$B = \text{POST}(B^{\text{aux}}, G_2, \text{Flow}(q_1))$

Reachability computation



$$G_1 = F_1 \cap G$$

$$G_2 = F_2 \cap G$$



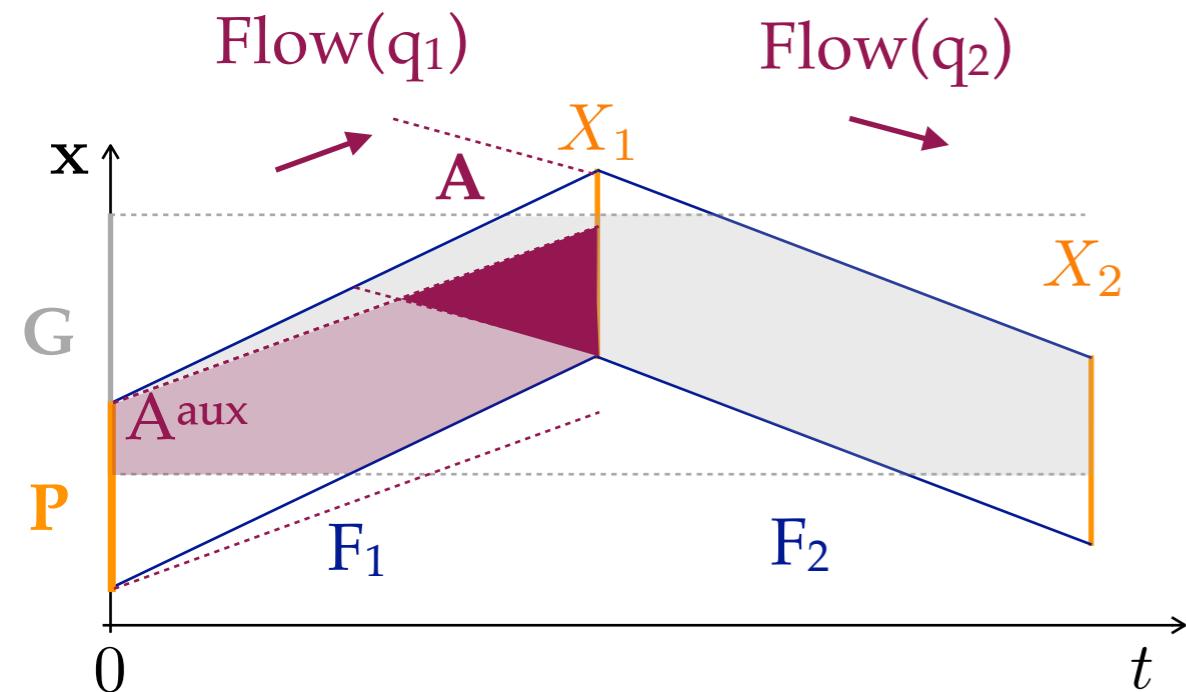
$$A^{aux} = \text{POST}(P, G_1, \text{Flow}(q_1))$$

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

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

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

Reachability computation

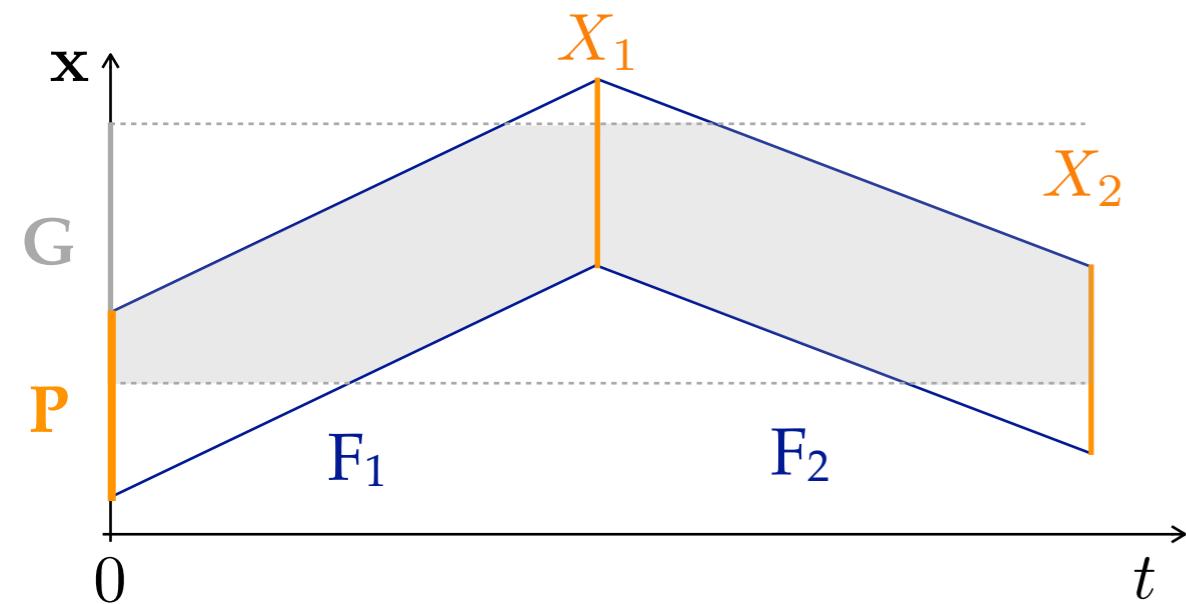


$$G_1 = F_1 \cap G$$

$$G_2 = F_2 \cap G$$

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

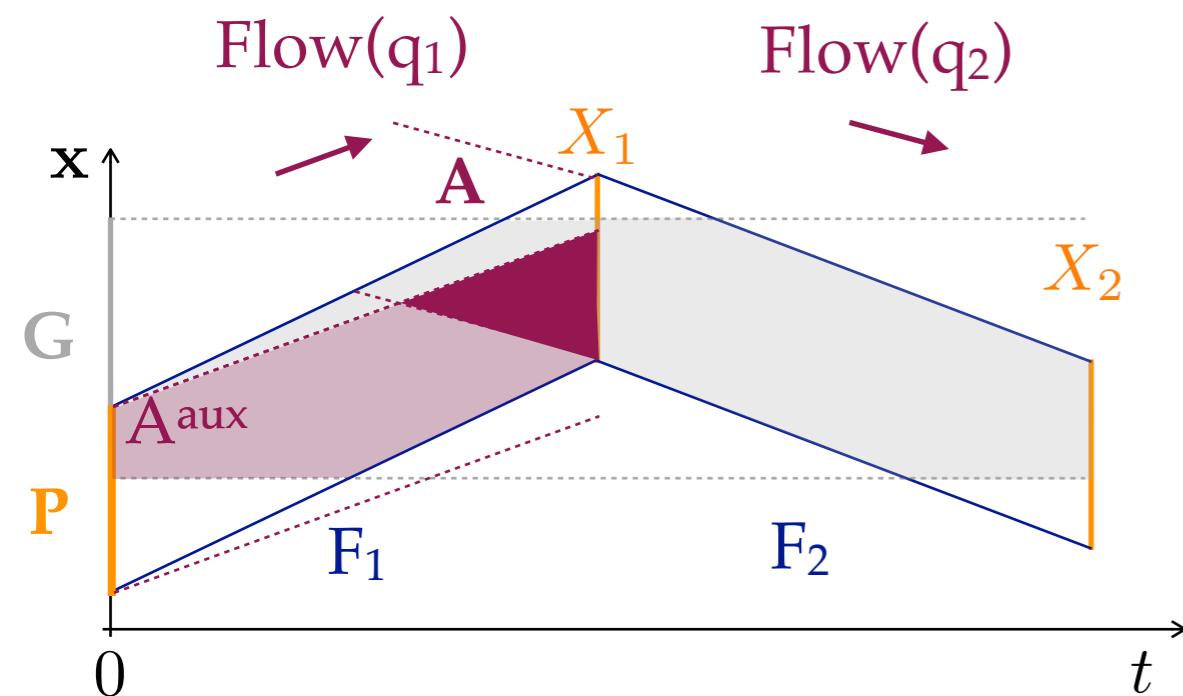
$A = \text{PRE}(X_1, A^{aux}, \text{Flow}(q_2))$



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

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

Reachability computation

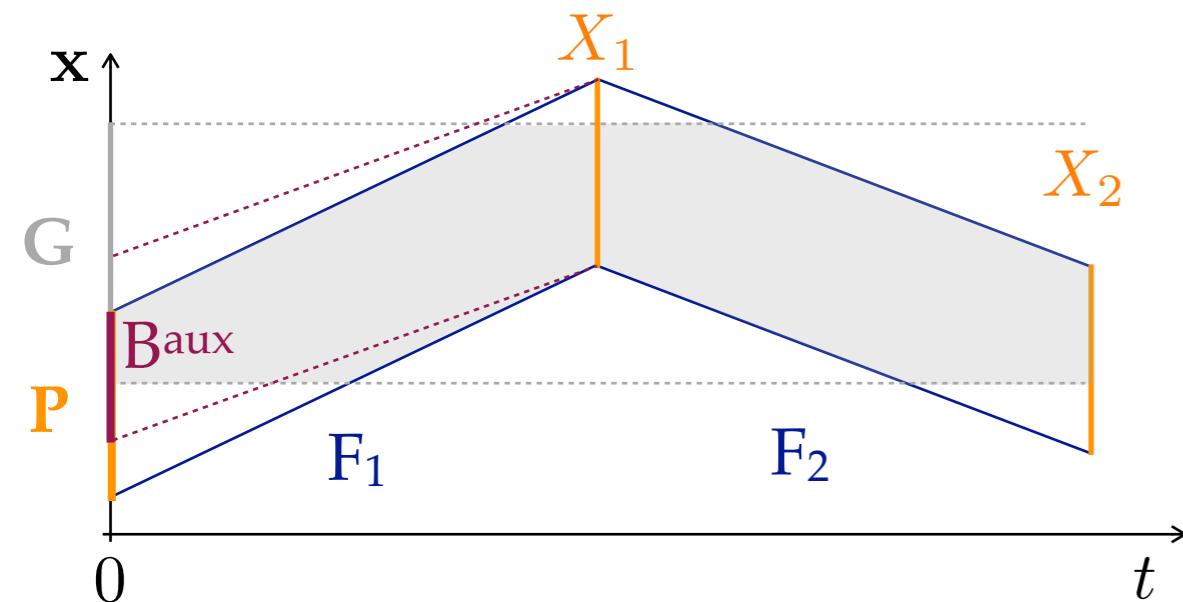


$$G_1 = F_1 \cap G$$

$$G_2 = F_2 \cap G$$

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

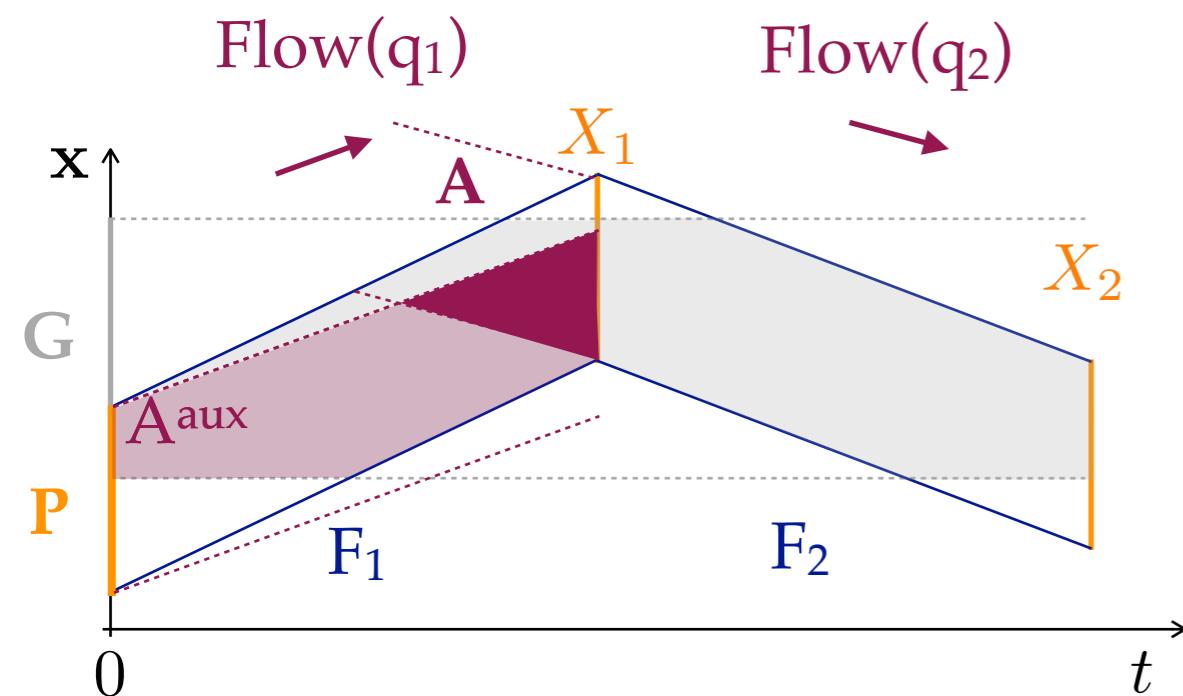
$A = \text{PRE}(X_1, A^{aux}, \text{Flow}(q_2))$



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

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

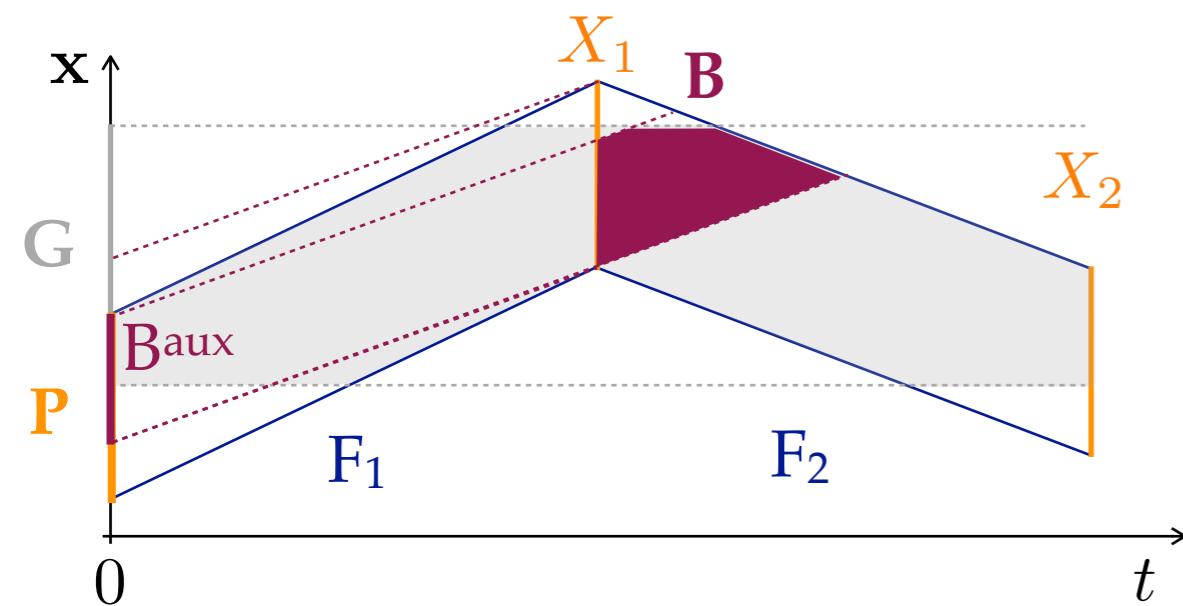
Reachability computation



$$G_1 = F_1 \cap G$$

$$G_2 = F_2 \cap G$$

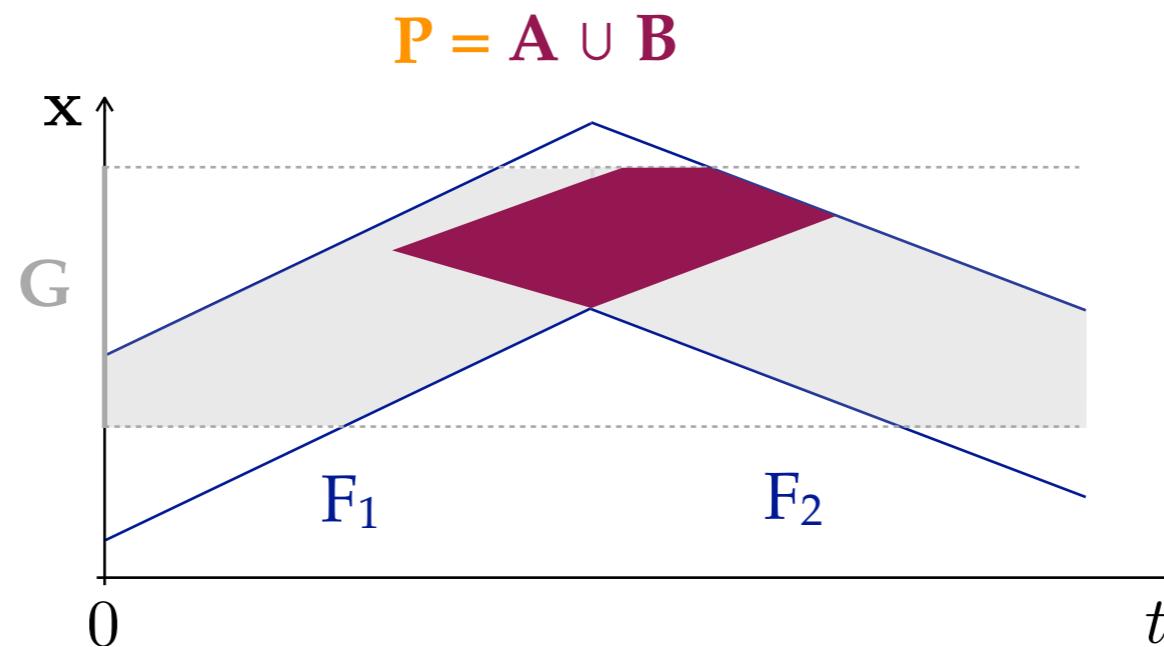
```
Aaux = POST (P, G1, Flow(q1))  
A = PRE (X1, Aaux, Flow(q2))
```



```
Baux = PRE (X1, P, Flow(q1))  
B = POST (Baux, G2, Flow(q1))
```

Reachability computation

Reachable switching set



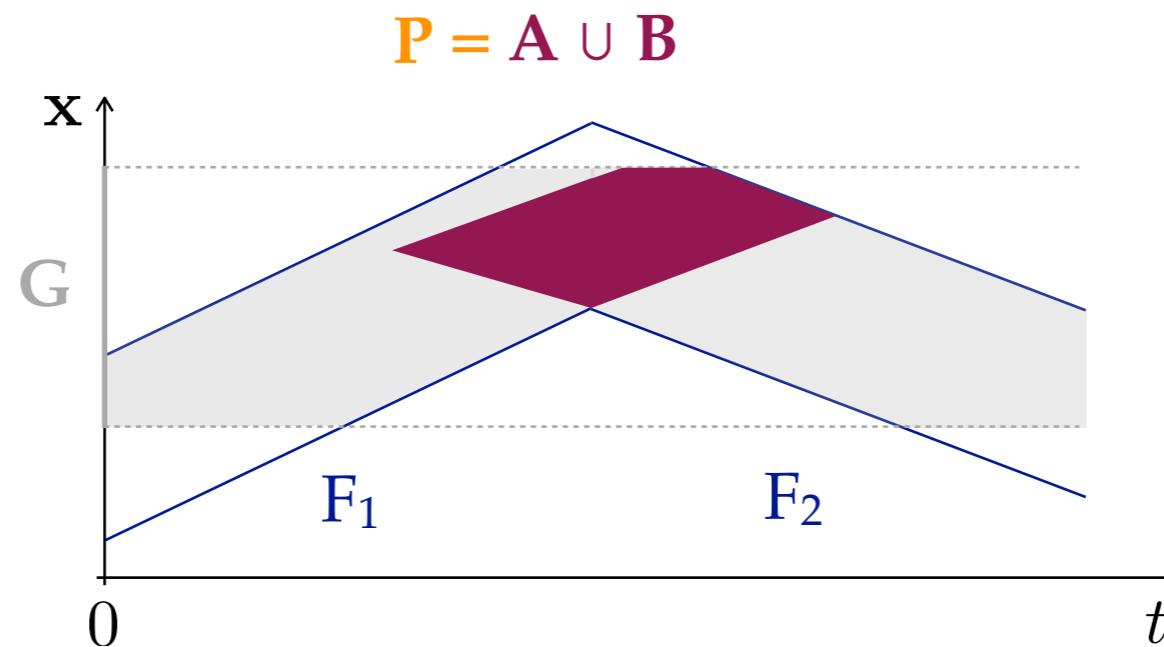
- ❖ Compute iteratively the reachable switching set P until the last ε -tube piece:

Last $P \neq \emptyset \Rightarrow \text{membership True}$

Last $P = \emptyset \Rightarrow \text{membership False}$

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

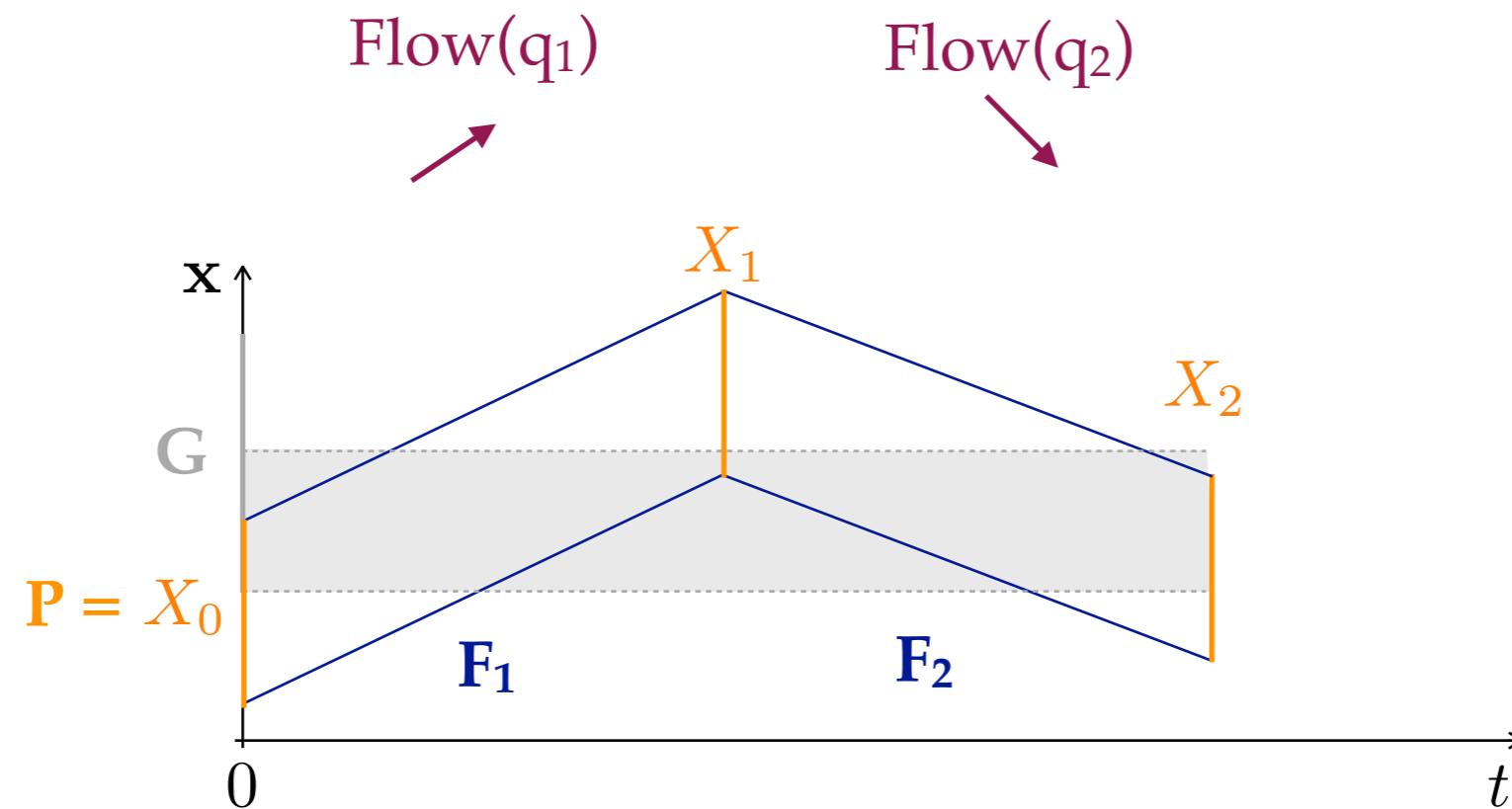
Last $P = \emptyset \Rightarrow$ membership False

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

Relaxation

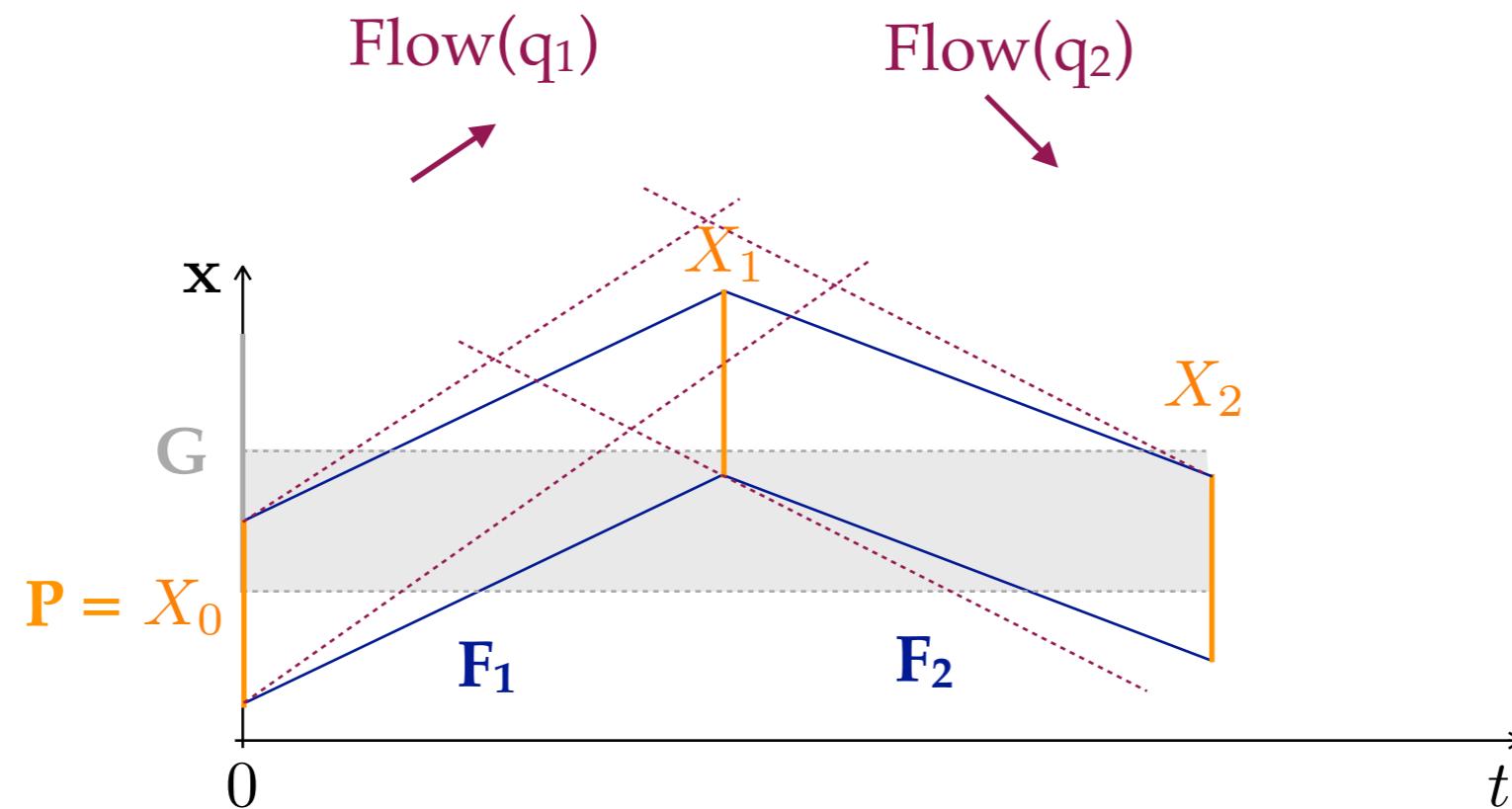
Relaxation problem

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 ?



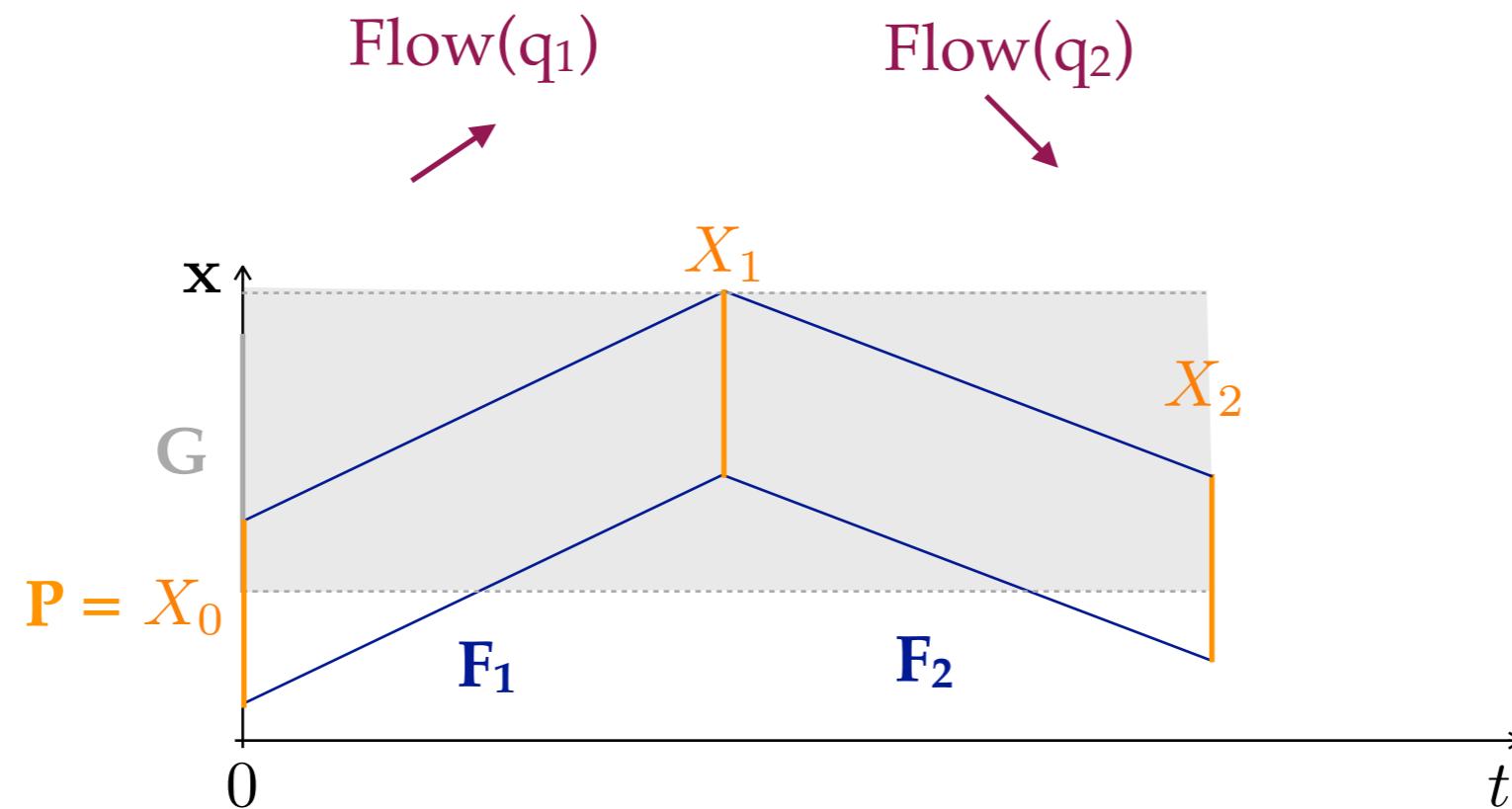
Relaxation problem

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 ?



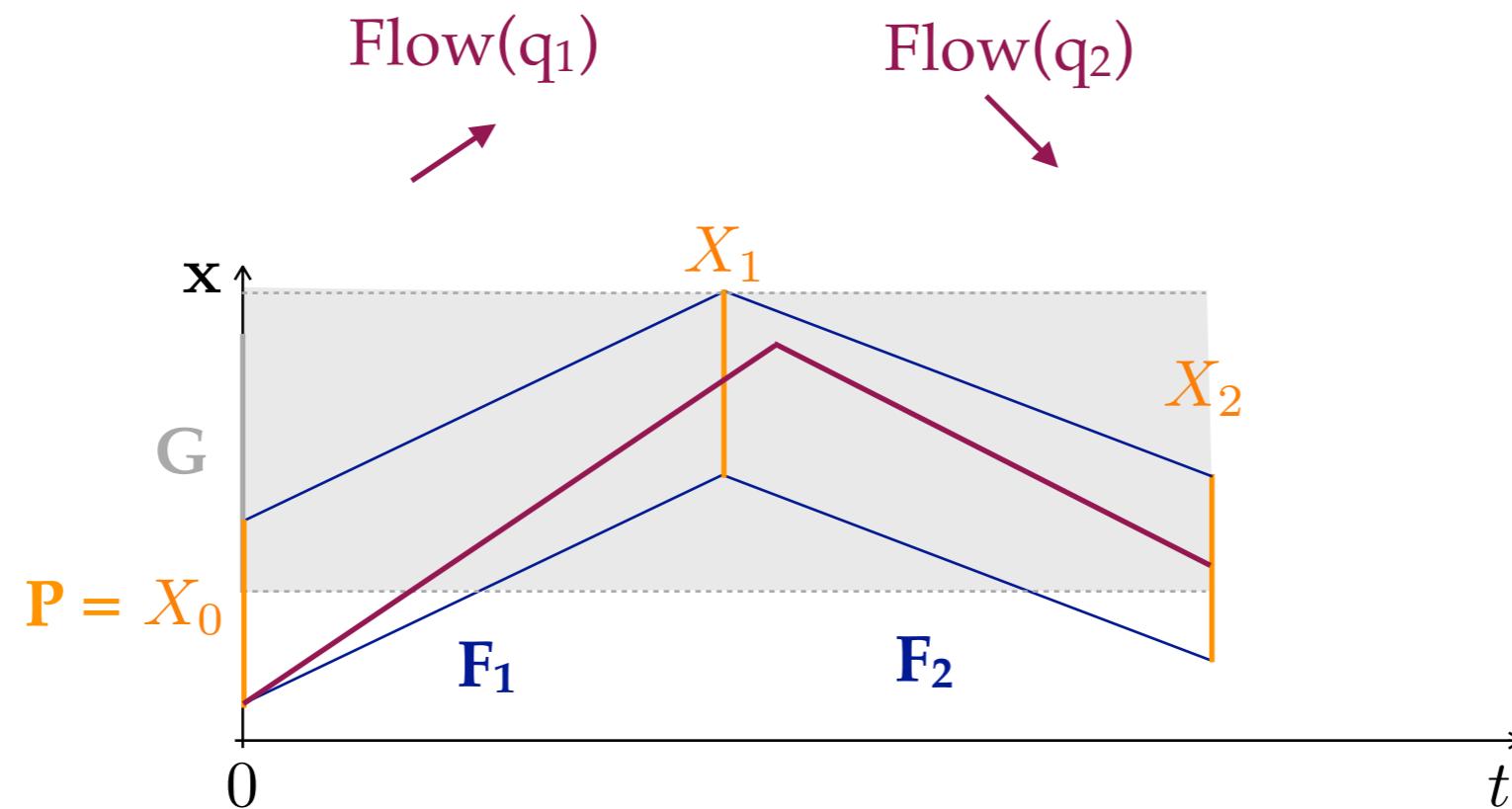
Relaxation problem

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 ?



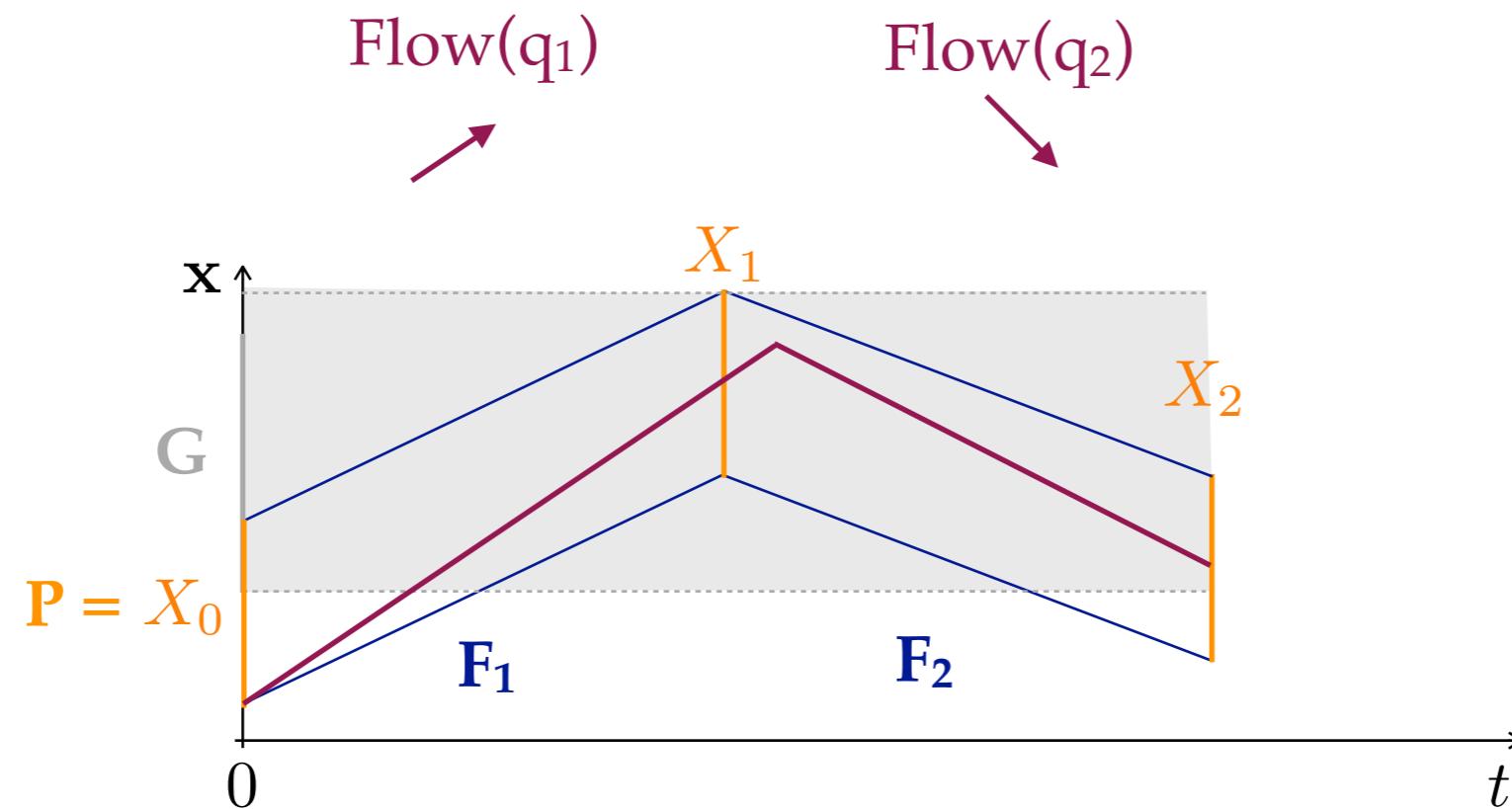
Relaxation problem

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 $\text{Flow}(q'_i) \approx \text{slope}(p_i)$
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:

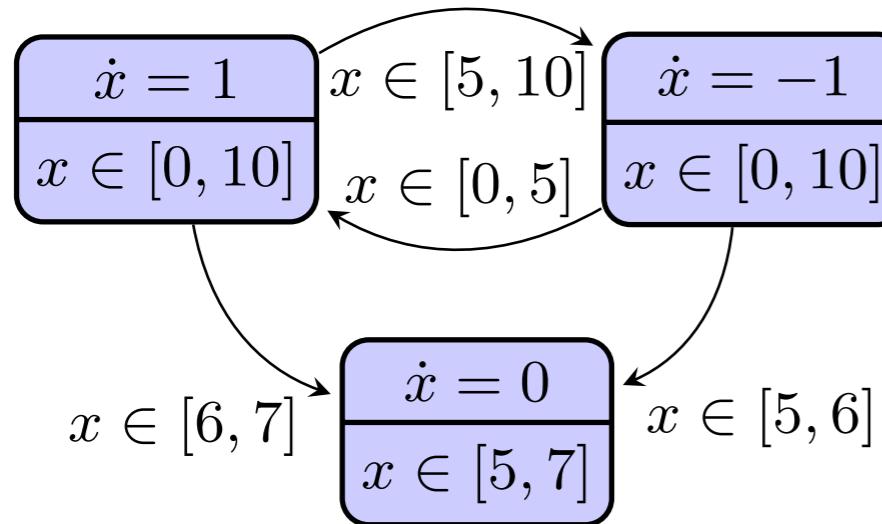
1. Existing q'_i with $\text{Flow}(q'_i) \approx \text{slope}(p_i)$
2. New q'_i with $\text{Flow}(q'_i) = \text{slope}(p_i)$

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

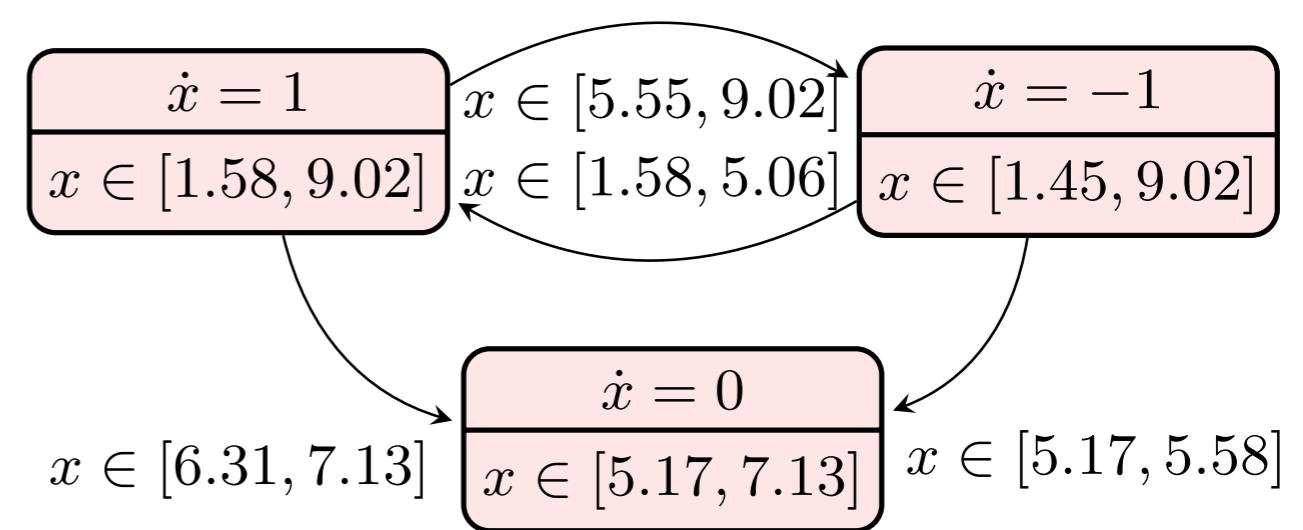
Experiments

Synthetic data

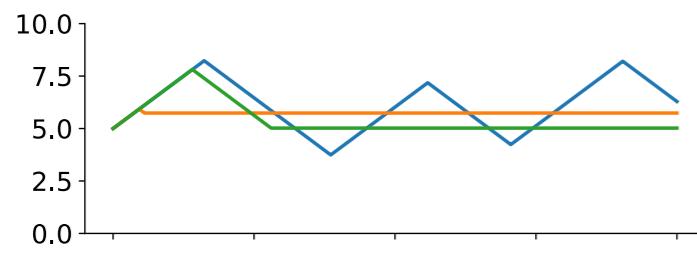
Original LHA



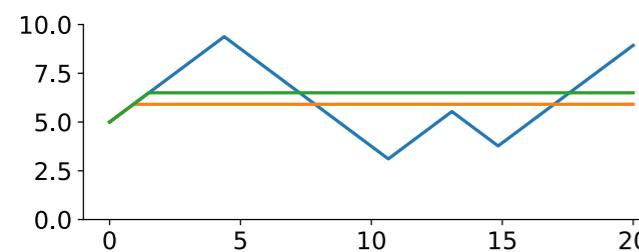
Synthesized LHA after 10 iterations



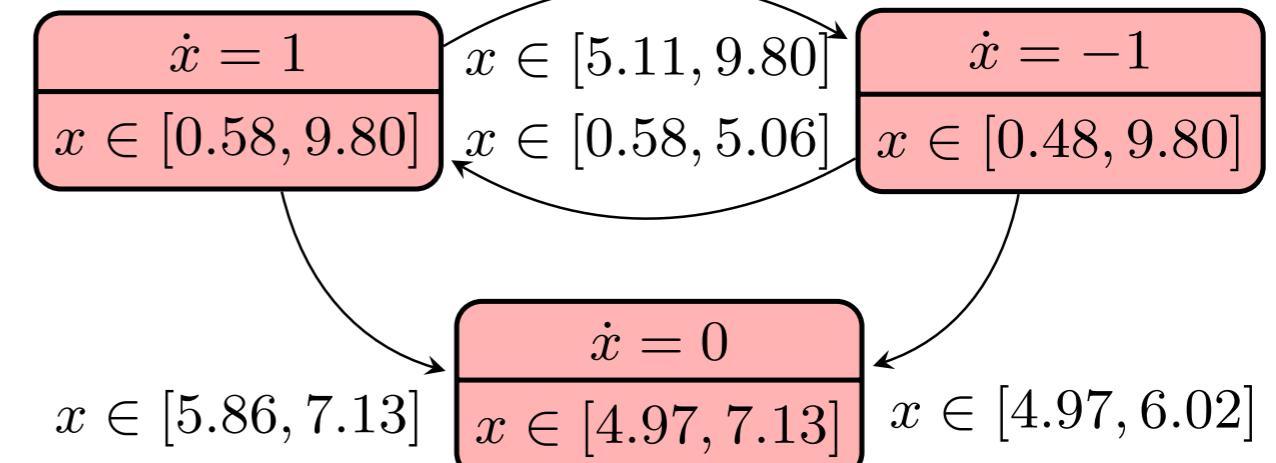
Executions of original LHA



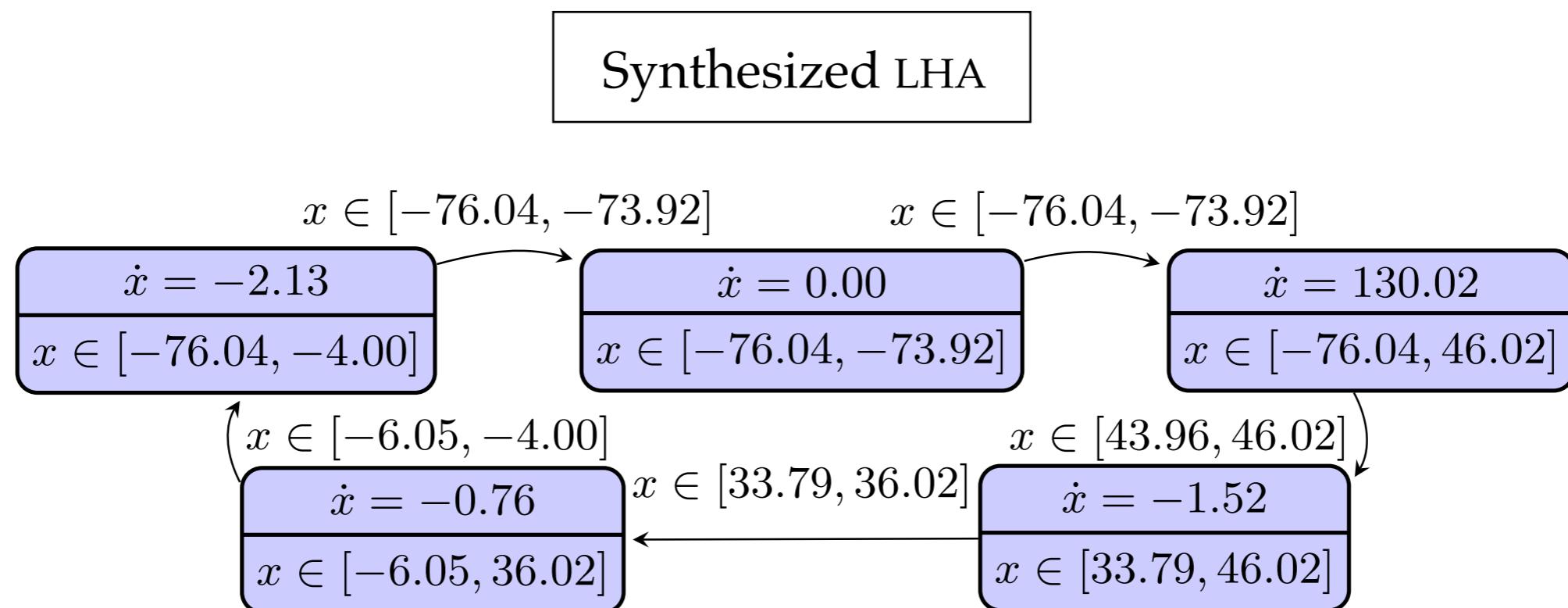
Executions of synthesized LHA



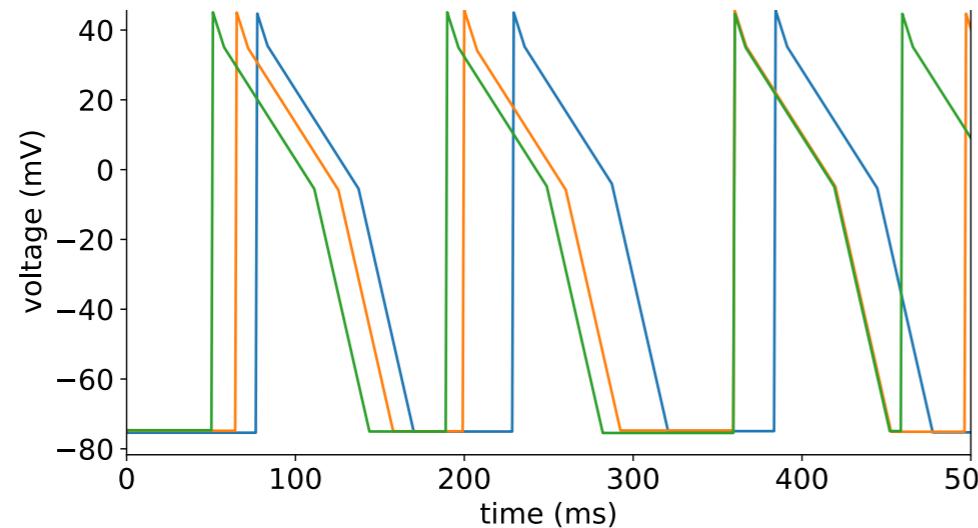
Synthesized LHA after 100 iterations



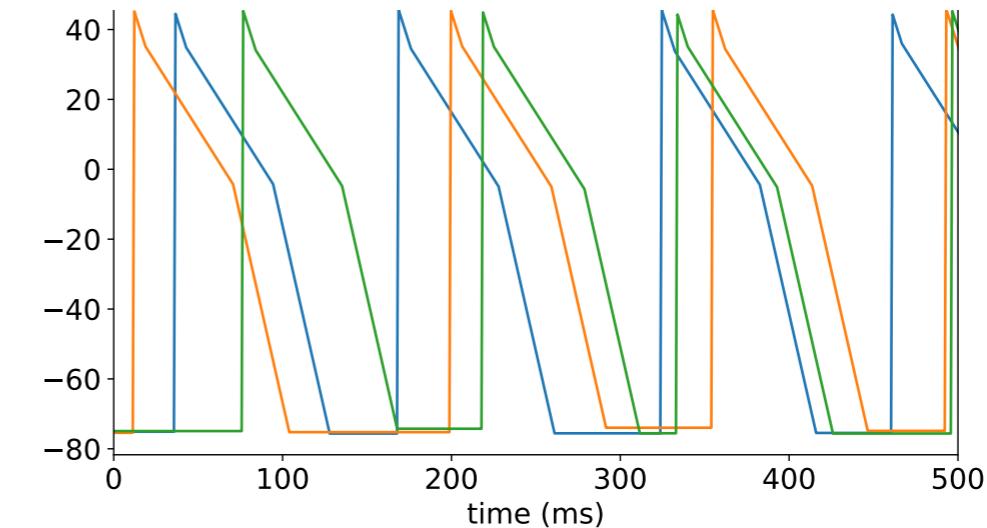
Cell model



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?