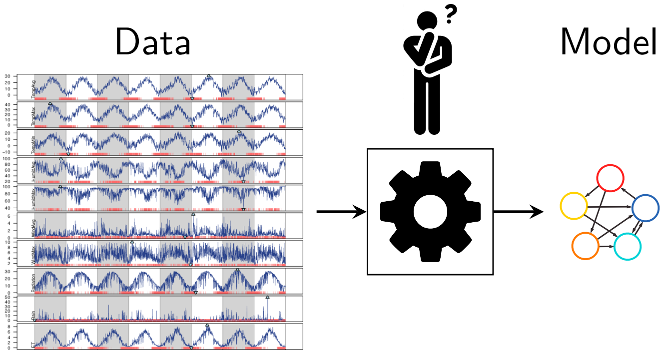
Synthesis of hybrid automata from time-series data

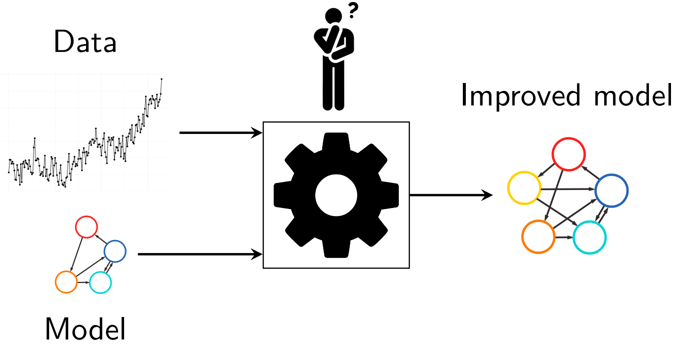
Christian Schilling

September 6, 2022

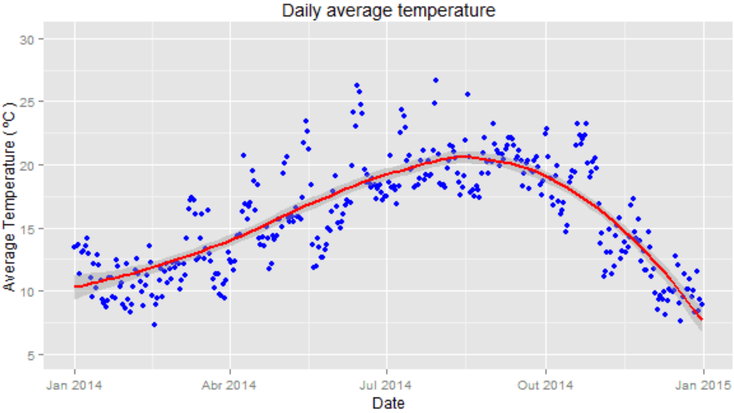
Model synthesis from data



Model synthesis from data

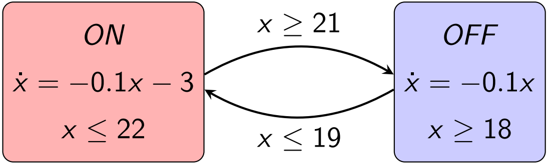


Data: Time series



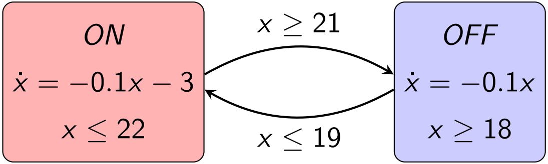
Model: Hybrid automata

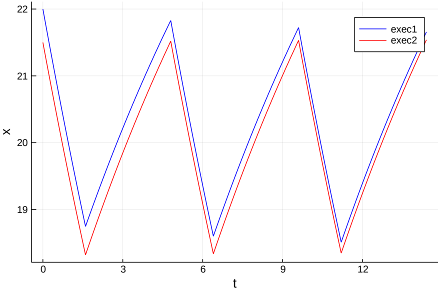
Heater:



Model: Hybrid automata

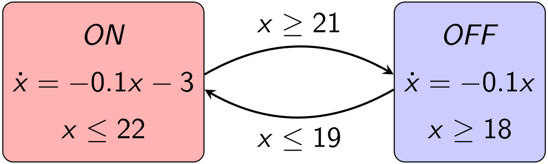
Heater:



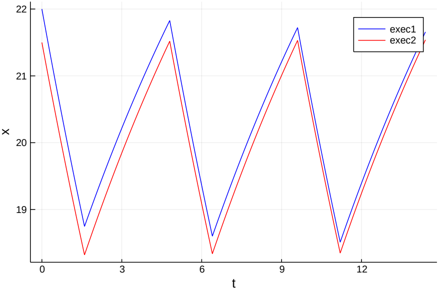


Model: Hybrid automata

Heater:

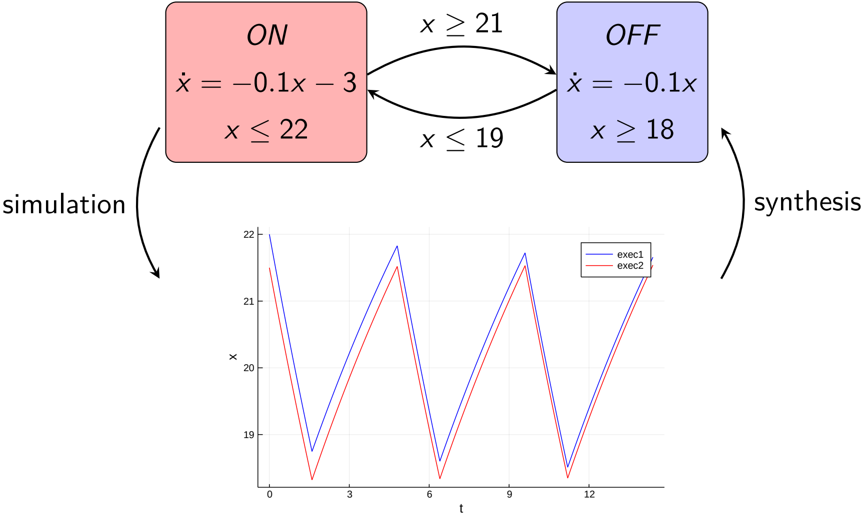


simulation



Model: Hybrid automata

Heater:



Works covered in this presentation

- Offline and online synthesis of linear hybrid automata[1]
- Online synthesis of hybrid automata with affine dynamics[2]
- Offline synthesis of parametric linear hybrid automata[3]

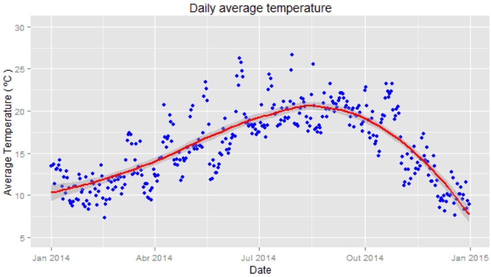
[1] M. García Soto, T. A. Henzinger, C. Schilling, and L. Zeleznik. *CAV*. 2019.
[2] M. García Soto, T. A. Henzinger, and C. Schilling. *HSCC*. 2021.
[3] M. García Soto, T. A. Henzinger, and C. Schilling. *ATVA*. 2022.

Problem statement

- Find a model that is *close* to the data
- How to formalize that?

Problem statement

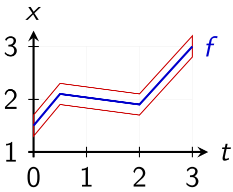
- Find a model that is *close* to the data
- How to formalize that?
- Our answer:
  Require that there is an execution σ that is ε-close to the data

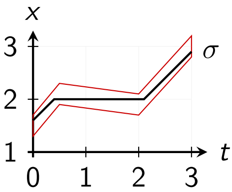


Problem statement

- Require that there is an execution σ that is ε-close to the data
- Sometimes easier to find reference function[1] f that is δ-close to data[2] and then require execution σ that is $(\varepsilon - \delta)$-close to function f

$$\mathrm{d}(f, t) \leq \delta \quad \wedge \quad \mathrm{d}(\sigma, f) \leq \varepsilon - \delta \quad \Longrightarrow \quad \mathrm{d}(\sigma, t) \leq \varepsilon$$





[1] For example the linear interpolation

[2] Distance $\mathrm{d}(f, t)$ defined in the obvious way

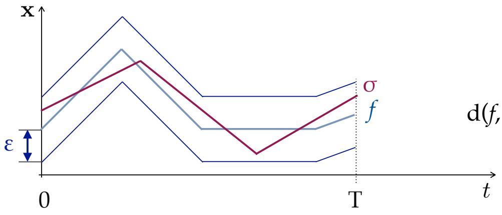
Problem statement

ε-capturing

A model ε-*captures* a function f if there exists an execution σ with $\mathrm{d}(f, \sigma) \leq \varepsilon$

Synthesis problem

Given a finite set of functions $\mathcal{F}$ and $\varepsilon \in \mathbb{R}_{\geq 0}$, construct a model that ε-captures each $f \in \mathcal{F}$



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

Problem statement

ε-capturing

A model ε-*captures* a function f if there exists an execution σ with $\mathrm{d}(f, \sigma) \leq \varepsilon$

Synthesis problem

Given a finite set of functions $\mathcal{F}$ and $\varepsilon \in \mathbb{R}_{\geq 0}$, construct a model that ε-captures each $f \in \mathcal{F}$

- Trivial problem: just have automaton with ε-close behavior for every $f \in \mathcal{F}$
- Additional constraints, e.g., as few locations as possible

Some underlying ideas

- A set of executions annotated with location names induces a unique minimal automaton

- Forget about automaton; instead synthesize executions with maximal sharing of locations
- Forget about guard and invariant constraints; only synthesize dynamics and then choose constraints as tightly as possible

Online synthesis algorithm

$\mathcal{H}_0 :=$ dummy automaton
for each $f_i \in \mathcal{F}$:
  if f_i is ε-captured by $\mathcal{H}_i$
    $\mathcal{H}_{i+1} := \mathcal{H}_i$
  else
    $\mathcal{H}_{i+1} :=$ modify($\mathcal{H}_i$, f_i)

Online synthesis algorithm

$\mathcal{H}_0 :=$ dummy automaton
for each $f_i \in \mathcal{F}$:
  if f_i is ε-captured by $\mathcal{H}_i$
    $\mathcal{H}_{i+1} := \mathcal{H}_i$
  else
    $\mathcal{H}_{i+1} :=$ modify($\mathcal{H}_i$, f_i)

Overview

Model synthesis from data

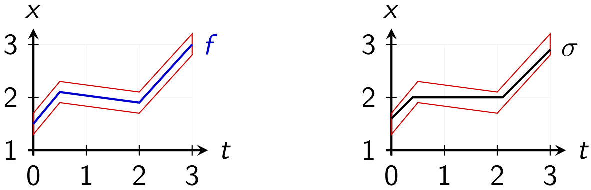
Offline and online synthesis of linear hybrid automata

Online synthesis of hybrid automata with affine dynamics

Offline synthesis of parametric linear hybrid automata

Synthesis of linear hybrid automata[1]

- Continuous dynamics: $\dot{x} = c$
- Constraints (invariants and guards): $\bigwedge_i a_i x \leq b_i$
- ε is given
- Synchronous vs. asynchronous switching



[1] M. García Soto, T. A. Henzinger, C. Schilling, and L. Zeleznik. *CAV*. 2019.

Synthesis of linear hybrid automata[1]

- SMT-based offline synthesis with synchronous switching
- Size constraint: model has minimal number of locations
- Let f be a piecewise-linear function with m pieces switching at times $(t_j)_j$ and points $\mathbf{x}_j = f(t_j)$
- Formula $\phi_{f,\varepsilon}(\ell)$ satisfiable iff there exists ε-close execution with ℓ locations

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

- Lift to set $\mathcal{F}$ (i.e., offline synthesis) via

$$\phi_{\mathcal{F},\varepsilon}(\ell) = \bigwedge_{f \in \mathcal{F}} \phi_{f,\varepsilon}(\ell)$$

[1] M. García Soto, T. A. Henzinger, C. Schilling, and L. Zeleznik. *CAV*. 2019.

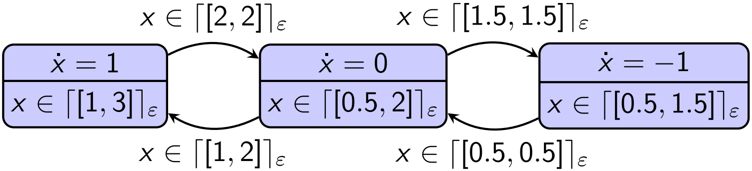
Synthesis of linear hybrid automata[1]

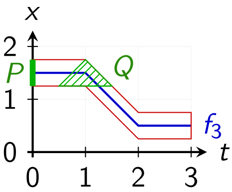
- Membership- and reachability-based online synthesis with asynchronous switching
- Size constraints: minimal number of locations and for each vertex of invariants/guards there is ε-close witness in $\mathcal{F}$
- Algorithm is complete for a class of automata (i.e., there is data to synthesize corresponding automaton)

- Algorithm to modify automaton
  - Try to find execution in existing model
  - If not found, try to extend constraints
  - If not successful, try to add transitions
  - If not successful, add locations

[1] M. García Soto, T. A. Henzinger, C. Schilling, and L. Zeleznik. *CAV*. 2019.

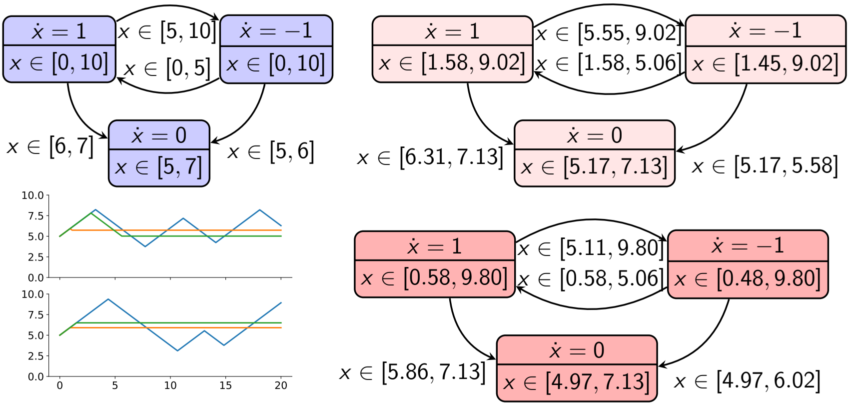
Synthesis of linear hybrid automata[1]





[1] M. García Soto, T. A. Henzinger, C. Schilling, and L. Zeleznik. *CAV*. 2019.

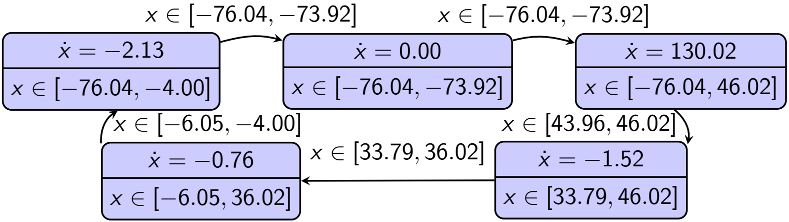
Synthesis of linear hybrid automata[1]

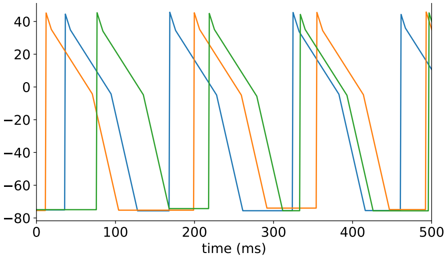


- Original model (top left)
- Synthesis results ($\varepsilon = 0.2$) after 10 resp. 100 traces (right)
- Sample input and output traces (bottom left)

[1] M. García Soto, T. A. Henzinger, C. Schilling, and L. Zeleznik. *CAV*. 2019.

Synthesis of linear hybrid automata[1]





[1] M. García Soto, T. A. Henzinger, C. Schilling, and L. Zeleznik. *CAV*. 2019.

Overview

Model synthesis from data

Offline and online synthesis of linear hybrid automata

Online synthesis of hybrid automata with affine dynamics

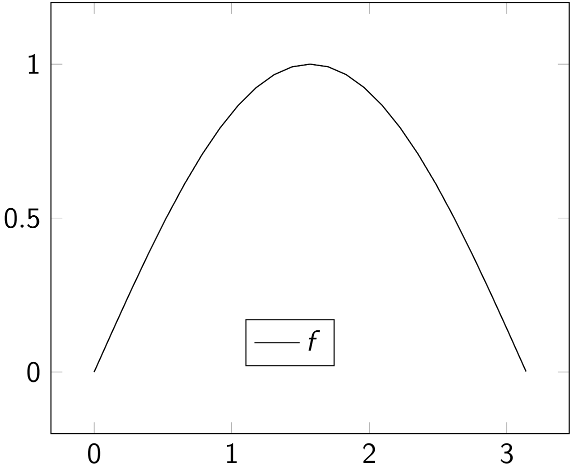
Offline synthesis of parametric linear hybrid automata

Synthesis of hybrid automata with affine dynamics[1]

- Continuous dynamics: $\dot{x} = Ax + b$
- Constraints (invariants and guards): $\bigwedge_i a_i x \leq b_i$ (as before)
- ε is given
- Synchronous switching
- Hierarchical search for minimal model modifications
- More complex because reachability is undecidable
  So we cannot always answer "is f ε-captured by $\mathcal{H}$?"

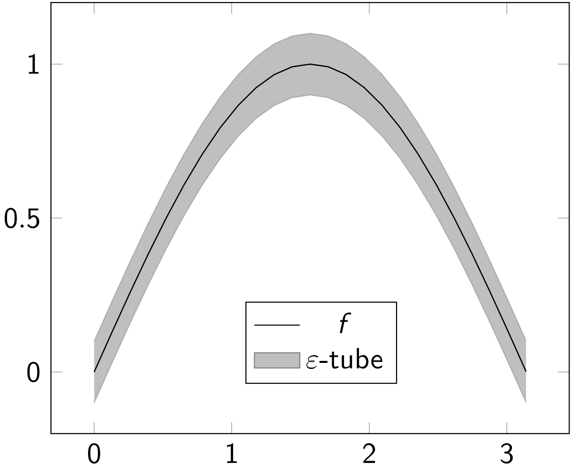
[1] M. García Soto, T. A. Henzinger, and C. Schilling. *HSCC*. 2021.

Synthesis of hybrid automata with affine dynamics[1]



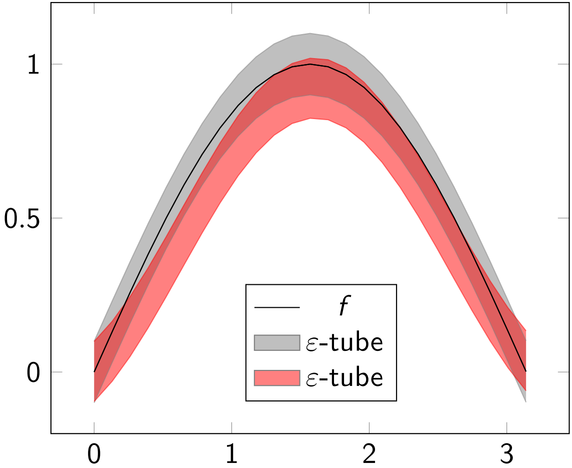
[1] M. García Soto, T. A. Henzinger, and C. Schilling. *HSCC*. 2021.

Synthesis of hybrid automata with affine dynamics[1]



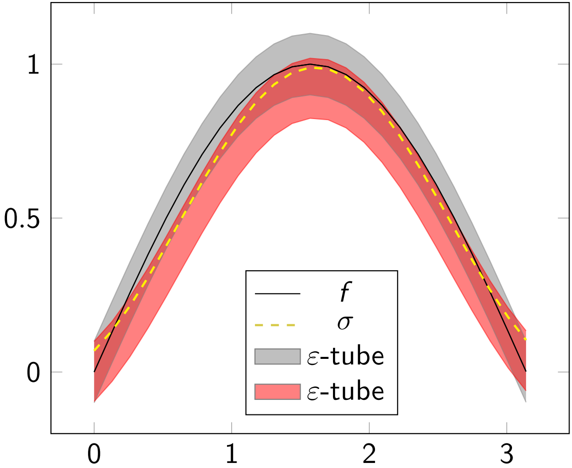
[1] M. García Soto, T. A. Henzinger, and C. Schilling. *HSCC*. 2021.

Synthesis of hybrid automata with affine dynamics[1]



[1] M. García Soto, T. A. Henzinger, and C. Schilling. *HSCC*. 2021.

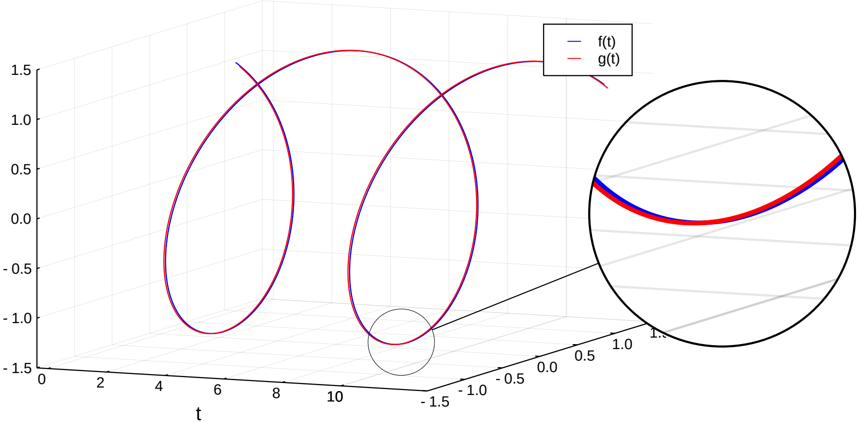
Synthesis of hybrid automata with affine dynamics[1]



[1] M. García Soto, T. A. Henzinger, and C. Schilling. *HSCC*. 2021.

Synthesis of hybrid automata with affine dynamics[1]

$$A = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \ \dot{f}(t) = Af(t), f(0) = \begin{pmatrix} 1 \\ 1 \end{pmatrix}, \ \dot{g}(t) = Ag(t), g(0) = \begin{pmatrix} 1.05 \\ 0.95 \end{pmatrix}$$

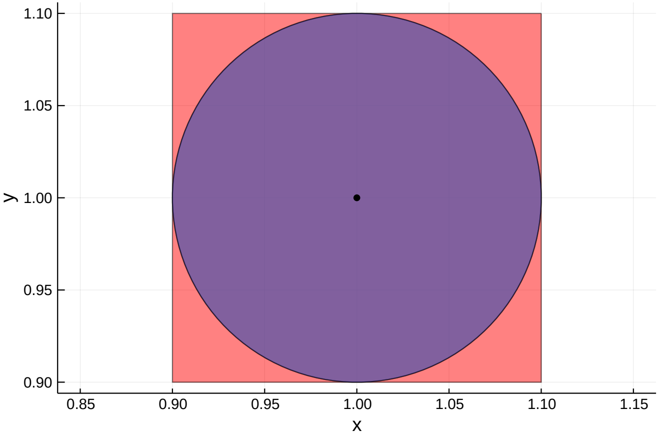


[1] M. García Soto, T. A. Henzinger, and C. Schilling. *HSCC*. 2021.

Synthesis of hybrid automata with affine dynamics[1]

$$\dot{f}(t) = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} f(t), f(0) = \begin{pmatrix} 1 \\ 1 \end{pmatrix}$$

Initial states with ε-captured executions

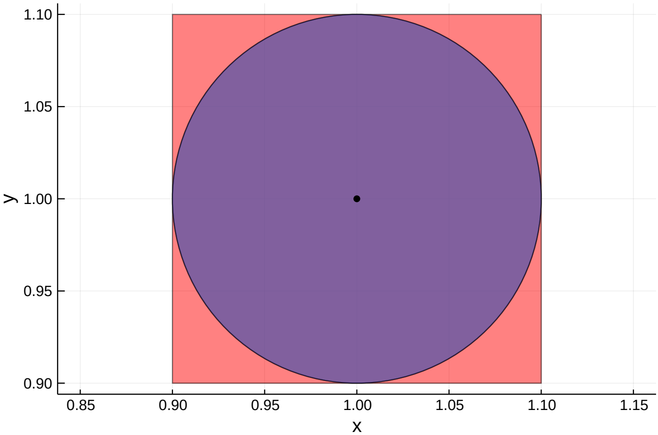


[1] M. García Soto, T. A. Henzinger, and C. Schilling. *HSCC*. 2021.

Synthesis of hybrid automata with affine dynamics[1]

Theorem

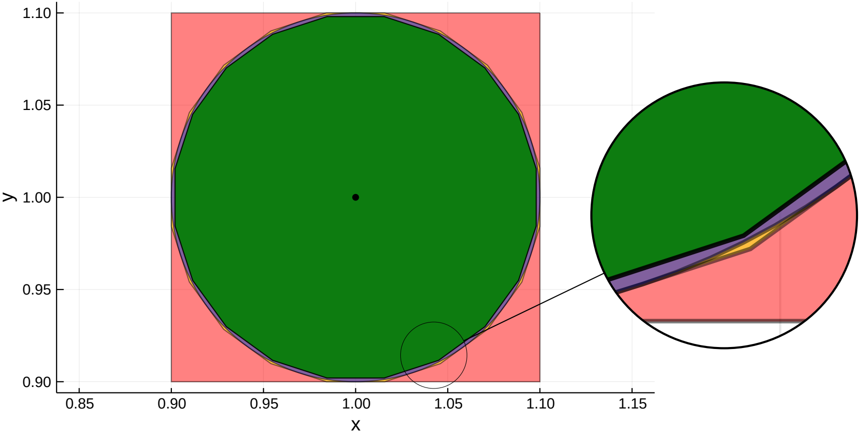
The set of initial states of ε-close executions (purple) is convex



[1] M. García Soto, T. A. Henzinger, and C. Schilling. *HSCC*. 2021.

Synthesis of hybrid automata with affine dynamics[1]

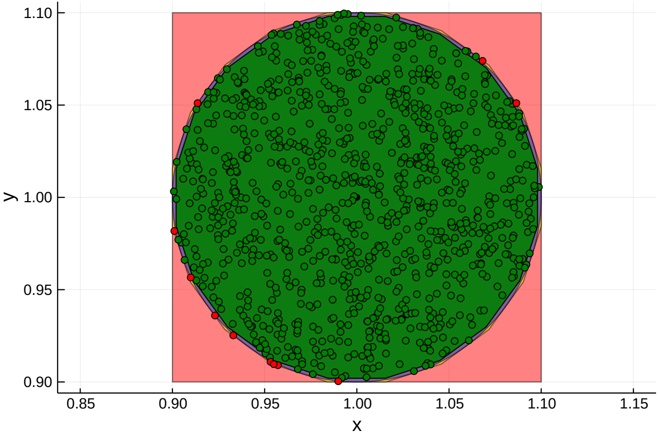
- Over-/underapproximation obtained with refinement procedure



[1] M. García Soto, T. A. Henzinger, and C. Schilling. *HSCC*. 2021.

Synthesis of hybrid automata with affine dynamics[1]

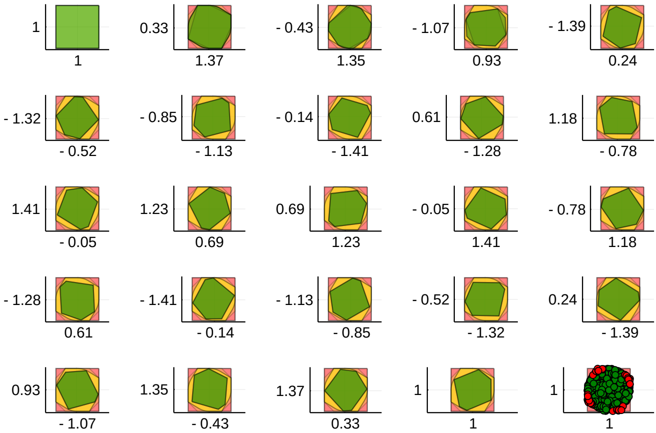
- Sampling from overapproximation



[1] M. García Soto, T. A. Henzinger, and C. Schilling. *HSCC*. 2021.

Synthesis of hybrid automata with affine dynamics[1]

- Precision after many location switches



[1] M. García Soto, T. A. Henzinger, and C. Schilling. *HSCC*. 2021.

Synthesis of hybrid automata with affine dynamics[1]

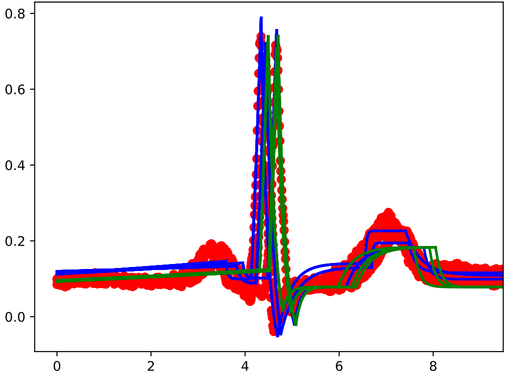
- Hierarchical search for model modifications

- $([\,], \mathcal{H}, (0,0,0), 2)$
  - $(q_1, \mathcal{H}_1, (0,0,1), 3)$
  - $(q_2, \mathcal{H}_2, (0,0,0), 2)$
    - $\boxed{(q_2 q_1, \mathcal{H}_4, (0,1,1), 2)}$
    - $(q_2 q_2, \mathcal{H}_5, (0,1,0), 3)$
    - $(q_2 q_3, \mathcal{H}_6, (1,1,0), 1)$
  - $(q_3, \mathcal{H}_3, (1,0,0), 1)$

[1] M. García Soto, T. A. Henzinger, and C. Schilling. *HSCC*. 2021.

Synthesis of hybrid automata with affine dynamics[1]

Electrocardiogram example



[1] M. García Soto, T. A. Henzinger, and C. Schilling. *HSCC*. 2021.

Overview

Model synthesis from data

Offline and online synthesis of linear hybrid automata

Online synthesis of hybrid automata with affine dynamics

Offline synthesis of parametric linear hybrid automata

Synthesis of parametric linear hybrid automata[1]

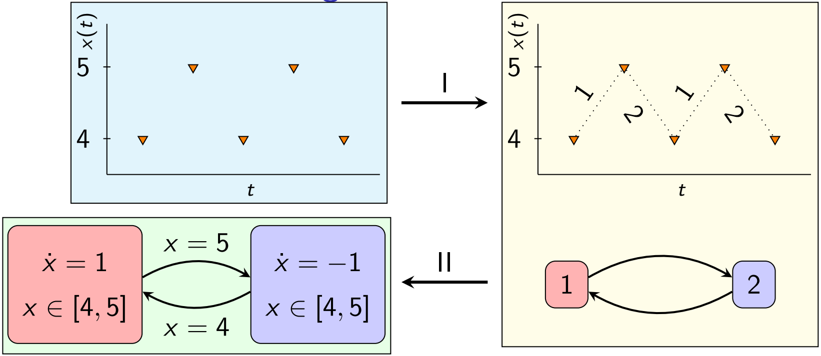
- Back to linear hybrid automata ($\dot{x} = c$)
- Synchronous switching
- Offline algorithm
- ε is not given

Problem statement

Given a finite set of time series and a discrete structure, find the minimal value $\varepsilon \in \mathbb{R}_{\geq 0}$ and an instantiated model $\mathcal{H}$ such that $\mathcal{H}$ ε-captures each time series

[1] M. García Soto, T. A. Henzinger, and C. Schilling. *ATVA*. 2022.

Algorithm sketch

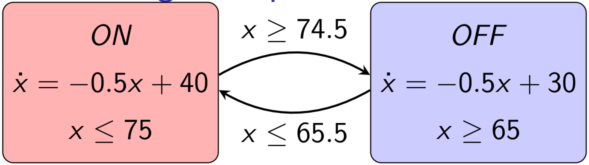


1. Fix discrete structure (by assigning location name to each piece in time series)

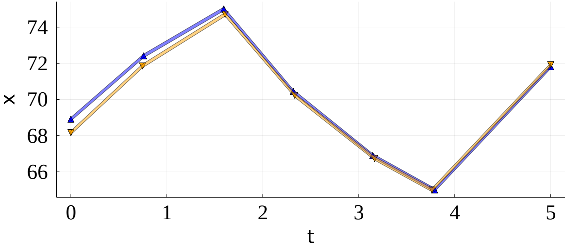
2. Construct family of models (parameter polyhedron) such that any instantiated model ε-captures data

   Here ε is parameter itself $\rightsquigarrow$ choose some minimizer

Running example - Initialization



We obtain two time series from simulations

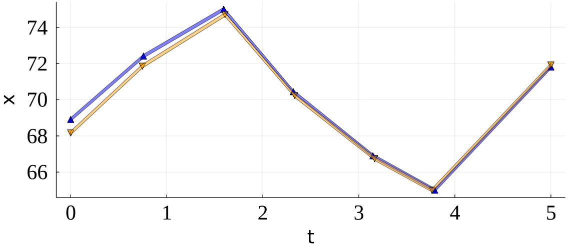


Running example - Initialization

$$
\begin{aligned}
t_1 &= (0.00, \; 0.76, \; 1.59, \; 2.32, \; 3.15, \; 3.79, \; 5.00)\\
d_1 &= (68.91, \; 72.41, \; 75.00, \; 70.44, \; 66.90, \; 65.00, \; 71.81)
\end{aligned}
$$

$$
\begin{aligned}
t_2 &= (0.0, \; 0.75, \; 1.61, \; 2.33, \; 3.16, \; 3.76, \; 5.00)\\
d_2 &= (68.16, \; 71.85, \; 74.70, \; 70.22, \; 66.75, \; 65.00, \; 71.92)
\end{aligned}
$$

We obtain two time series from simulations



Running example - Phase 1

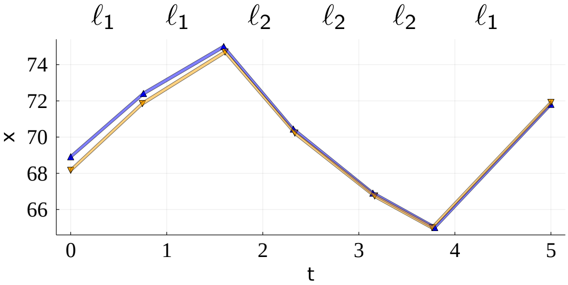
- Send slope vectors of pieces to clustering algorithm
- Clustering cost for different numbers of clusters k, together with relative improvement compared to $k - 1$

clusters	1	2	3	4	5	6	7	8
cost	259.76	17.07	11.80	2.46	0.78	0.09	0.04	0.01
rel. [%]	–	0.93	0.31	0.79	0.68	0.89	0.60	0.61

- Good values for k: 2, 4, 6 (we choose $k = 2$)
- Associated (one-dimensional) cluster centers (representing slopes): 4.53 and -4.46
- For both time series, assigned clusters are $(1, 1, 2, 2, 2, 1)$ (i.e., symbolic location ℓ_1 for pieces 1, 2, 6 and ℓ_2 for others)

Running example - Result of Phase 1

- Only retain symbolic locations from clustering, i.e., each piece in time series has associated location
- Induces discrete structure of automaton



Running example - Phase 2 (one time series)

- $\ell(k)$ yields symbolic location of piece k
- Construct linear program with slopes of symbolic locations $\mathbf{m}_1, \ldots, \mathbf{m}_\lambda$ as parameters
- Initial position $\mathbf{x}_0$ is another parameter
- Executions must be ε-close, where ε is another parameter

$$
\begin{aligned}
\{(\mathbf{m}_1, \ldots, \mathbf{m}_\lambda, \mathbf{x}_0, \varepsilon) \mid \mathbf{x}_0 &\in \mathcal{B}_\varepsilon(d(t_0)), \\
\mathbf{x}_0 + (t_1 - t_0)\mathbf{m}_{\ell(1)} &\in \mathcal{B}_\varepsilon(d(t_1)), \\
\mathbf{x}_0 + (t_1 - t_0)\mathbf{m}_{\ell(1)} + (t_2 - t_1)\mathbf{m}_{\ell(2)} &\in \mathcal{B}_\varepsilon(d(t_2)), \\
&\vdots \\
\mathbf{x}_0 + (t_1 - t_0)\mathbf{m}_{\ell(1)} + \ldots + (t_p - t_{p-1})\mathbf{m}_{\ell(p)} &\in \mathcal{B}_\varepsilon(d(t_p))\}.
\end{aligned}
$$

Running example - Result of Phase 1

- First half of (symmetric) constraints

$$
\begin{array}{ccccccccc}
 & & & & x_0^{(1)} & - & \varepsilon & \leq & 68.91 \\
0.76m_1 & & & + & x_0^{(1)} & - & \varepsilon & \leq & 72.41 \\
1.59m_1 & & & + & x_0^{(1)} & - & \varepsilon & \leq & 75.00 \\
1.59m_1 & + & 0.72m_2 & + & x_0^{(1)} & - & \varepsilon & \leq & 70.44 \\
1.59m_1 & + & 1.55m_2 & + & x_0^{(1)} & - & \varepsilon & \leq & 66.90 \\
1.59m_1 & + & 2.20m_2 & + & x_0^{(1)} & - & \varepsilon & \leq & 65.00 \\
2.80m_1 & + & 2.20m_2 & + & x_0^{(1)} & - & \varepsilon & \leq & 71.81
\end{array}
$$

Phase 2 (multiple time series)

- Intersect parameter polyhedra of different time series
- Translated to LP: concatenate constraints
- Technical detail: need new $\mathbf{x}_0$ dimensions

Correctness

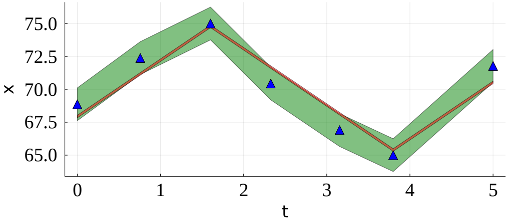
Problem statement (recalled)

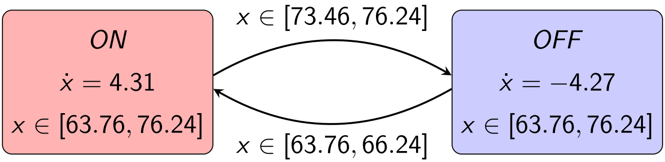
Given a finite set of time series and a discrete structure, find the minimal value $\varepsilon \in \mathbb{R}_{\geq 0}$ and an instantiated model $\mathcal{H}$ such that $\mathcal{H}$ ε-captures each time series

Theorem

Phase 2 solves problem in polynomial time

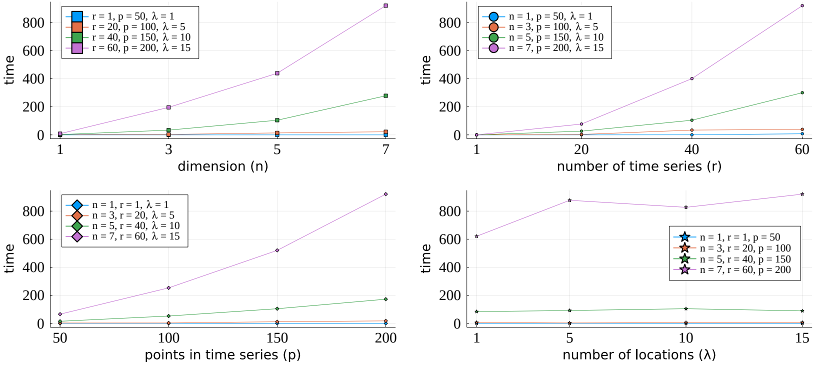
Running example - Result of Phase 2



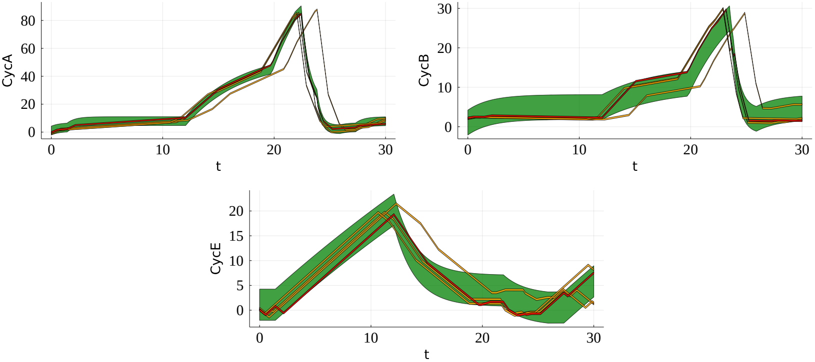


Evaluation: Scalability

- r time series with p data points, n dimensions and λ locations



Evaluation: Cell-cycle regulation



Summary

- Algorithmic synthesis of hybrid automata from time series
- Common idea: $\mathcal{H}$ ε-captures (has execution ε-close to) data
- Minimality guarantees
- Features
  - Synthesis: online[1,2] vs. offline[1,3]
  - Dynamics: constant[1,3] vs. affine[2]
  - Switching: synchronous[1,2,3] vs. asynchronous[1]
  - ε: given[1,2] vs. not given[3]
  - Scalability: low[1,2] vs. medium[3]

---

[2] M. García Soto, T. A. Henzinger, C. Schilling, and L. Zeleznik. *CAV*. 2019.

[3] M. García Soto, T. A. Henzinger, and C. Schilling. *HSCC*. 2021.

[4] M. García Soto, T. A. Henzinger, and C. Schilling. *ATVA*. 2022.

Future work

- Nonlinear dynamics
- Updates on transitions (resets)
- Better oracle for change points
- Backtracking of decisions
- Learning from negative examples