Efficient Parametric Identification for STL

Alexey Bakhirkin Thomas Ferrère Oded Maler

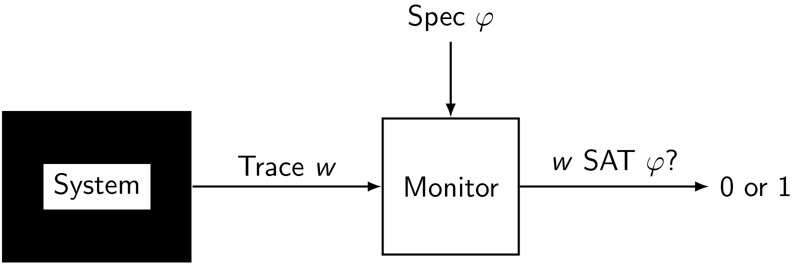
Verimag

IST AUSTRIA
Institute of Science and Technology

Monitoring vs. Parametric Identification

Monitoring

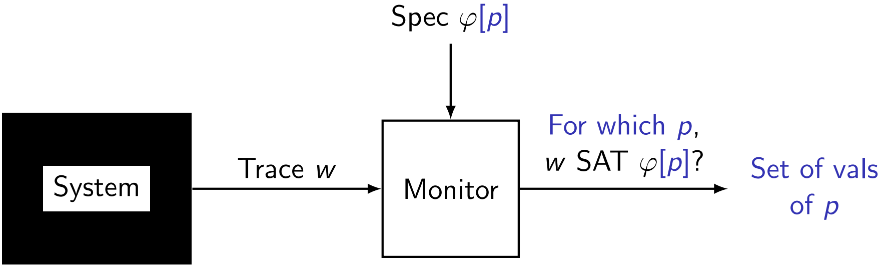
Do traces of a black box satisfy a property?



Monitoring vs. Parametric Identification

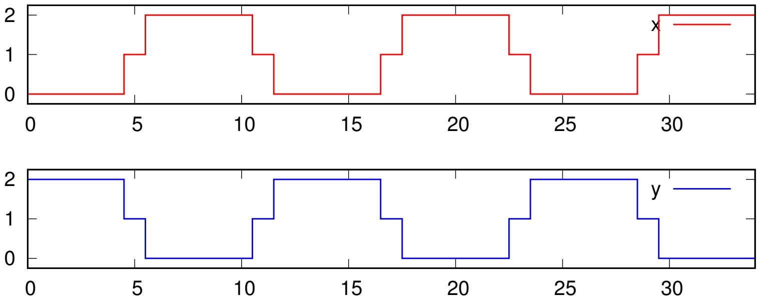
Parametric Identification

What is the value of a parameter of a black box?



- Find the set of all or tightest or etc values of parameters.
- From it, find the parameter of the black box.

Our Setting



- Real-valued.
- Piecewise-constant interpolation.
- Time is bounded.
- Offline computation.
- Specification language – Signal Temporal Logic

Signal Temporal Logic

Standard Semantics for Monitoring

$$\varphi ::= x \geq c \mid x \leq c \mid F_{[a,b]}\, \varphi \mid \varphi_1 U \varphi_2 \mid$$
$$\mathit{true} \mid \mathit{false} \mid \neg\varphi \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \wedge \varphi_2 \mid$$

A formula evaluates to *true* or *false* at a time point t.

- $x \leq c$, if $x(t) \leq c$.
- $F_{[a,b]}\, \varphi$, if φ holds for some $t' \in [t + a, t + b]$.
- $\varphi_1 U \varphi_2$, if φ_2 holds at some $t' \geq t$, and φ_1 holds on $[t, t']$.

Parameterized STL

Semantics for Parametric Identification

$$\varphi ::= x \leq c \mid x \geq c \mid x \leq p \mid x \geq p \mid F_{[a,b]}\,\varphi \mid \varphi_1 U \varphi_2 \mid$$
$$\mathit{true} \mid \mathit{false} \mid \neg\varphi \mid \varphi_1 \vee \varphi_2 \mid \varphi_1 \wedge \varphi_2 \mid$$

For every time point t, we want to find the validity domain – for which parameter values the formula evaluates to *true*.

- $x \leq p$: $p \geq x(t)$.
- $F_{[a,b]}\,\varphi$: union of the validity domains on $[t + a, t + b]$.
- $\varphi_1 U \varphi_2$: see paper.

Single polarity – we want that a given parameter appears only in $\leq$ or only in $\geq$ expressions.

Every validity domain is upward/downward-closed set of rectangles. There is a finite number of tightest parameter combinations.

What PSTL Can Do?

Motivation

1. Find system parameters from system traces.

   S. Jha et al., RV 2017 – extracting parameters from car sensor traces.

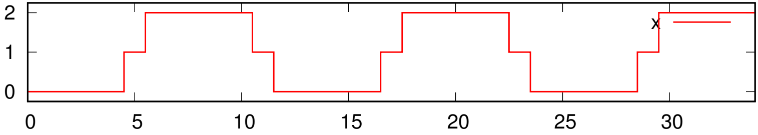
   For example,

   - $G\,(x \leq p_1 \wedge x \geq p_2)$ – finds the range of x – between min p_1 and max p_2.
   - $F\,(x \leq p_1 \wedge x \geq p_2)$ – enumerates the possible values of x. The domain has the form $(p_1 \geq x_1 \wedge p_2 \leq x_1) \vee (p_1 \geq x_1 \wedge p_2 \leq x_1) \vee \cdots$

2. Evaluate formulas with universal/existential quantifiers. Not this paper, see K. Havelund et al., FMCAD 2017.

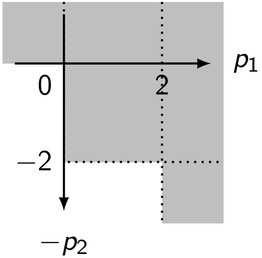
Example

Extract high and low thresholds



Formula: $G\, F_{[0,t_{\mathrm{edge}}+t_{\mathrm{stab}}]}((G_{[0,t_{\mathrm{stab}}]}\, x \leq p_1) \vee (G_{[0,t_{\mathrm{stab}}]}\, x \geq p_2))$

Validity domain at time 0: $(p_1 \geq 2) \vee (p_1 \geq 0 \wedge p_2 \leq 2) \vee (p_2 \leq 0)$



Previous Approaches

E. Asarin, A. Donzé, O. Maler, D. Nickovic, RV 2011

B. Hoxha, A. Dokhanchi, G. Fainekos, STTT 2017

- Allow time parameters.
- Search in parameter space (single polarity helps).
- Quantifier elimination in logical encoding.

Our Approach

Setting

- Piecewise-constant approximation.
- Single polarity.
- No time parameters.
- The validity signal (validity domain over time) is piecewise-constant.
- A single validity domain is an upward-closed set of boxes, representable as a set of points.

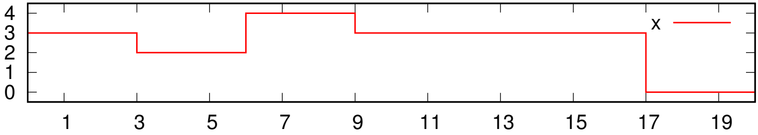
Compute validity signals directly as maps from time segments to sets of boxes.

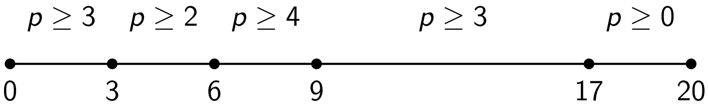
Bottom-up over the formula structure:

- $x \leq p$: directly, see next slide.
- $F_{[a,b]}\,\varphi$: running union of validity signal of φ over the window $[t+a, t+b]$.
- $\varphi_1 U \varphi_2$: see paper.

Atomic Comparison

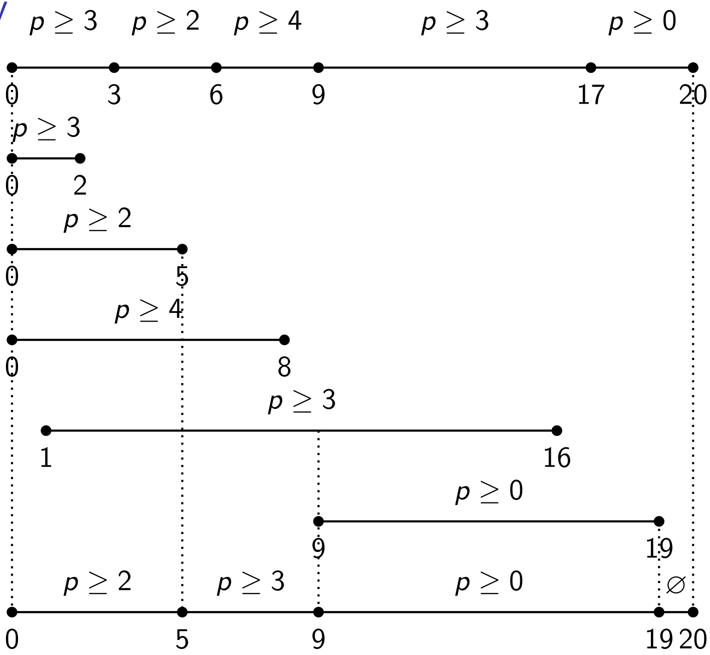
$x \leq p$





Eventually

$F_{[1,8]}\, x \leq p$



Eventually

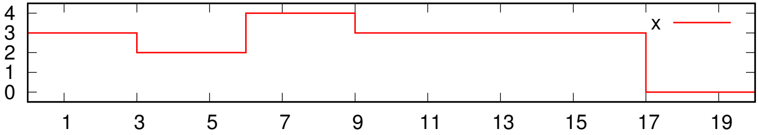
$F_{[1,8]}\, x \leq p$

- We adapt an algorithm by D. Lemire; originally linear in the length of the input.
- Fast in 1 dimension (1 parameter).
- Often reasonably fast in multiple dimensions; linear in the length of the input for a given formula.
- Not linear in general.

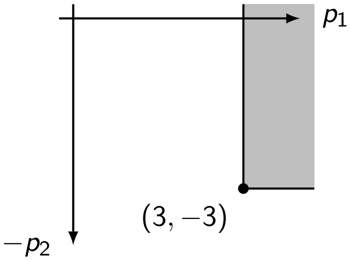
More Pessimistic Example

$F\,(x \le p_1 \wedge x \ge p_2)$

Start with $x \le p_1 \wedge x \ge p_2$



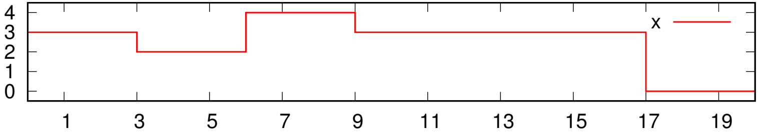
On $[0,3]$ the validity domain is $p_1 \ge 3 \wedge -p_2 \ge -3$.

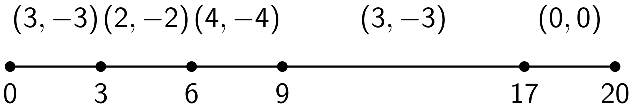


More Pessimistic Example

$F\left(x \leq p_1 \wedge x \geq p_2\right)$

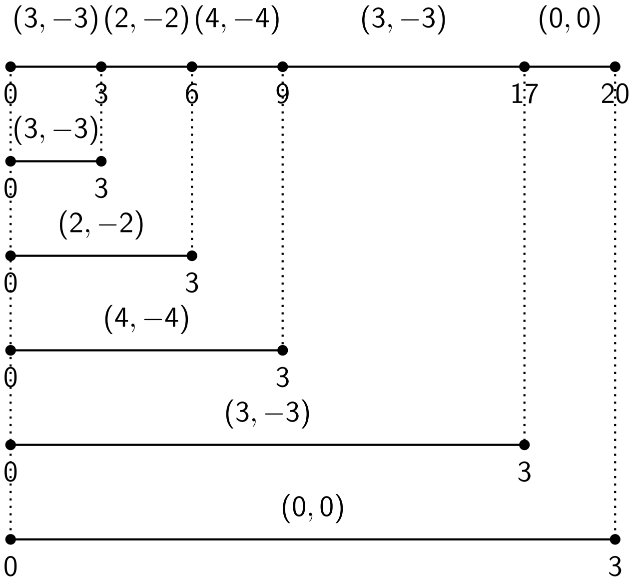
Start with $x \leq p_1 \wedge x \geq p_2$





More Pessimistic Example

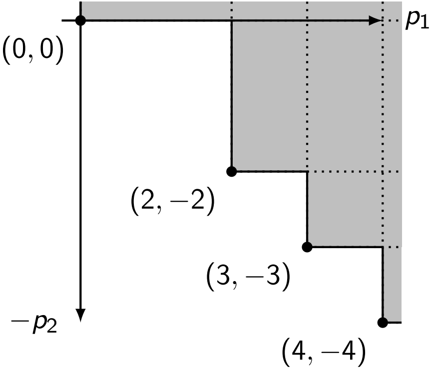
$F\left(x \leq p_1 \wedge x \geq p_2\right)$



More Pessimistic Example

$F\,(x \leq p_1 \wedge x \geq p_2)$

Validity domain at time 0 enumerates signal values.



Evaluation

- Prototype implementation in OCaml.
- Synthetic signals and output of a simulink model.
- Focus on examples where the validity domains have small number of boxes.
- In most examples, handle signals with 1M samples under a minute. For simple formulas, under 10 seconds.

Evaluation

Formula	Signal	Length / time, s 10^5	10^6
$\varphi_1 = G(x \leq p_1 \wedge x \geq p2)$	w_{sincos}	0.36	3.9
	w_{square}	0.36	3.75
$\varphi_2 = G(x \geq p \wedge y \geq p)$	w_{sincos}	0.28	3.4
$\varphi_3 = G(y \geq p \, U \, x \geq p)$	w_{sincos}	0.31	3.4
$\varphi_4 = G(x \leq 6 \rightarrow F_{[0,50]}(x \geq 6 \vee x \leq p))$	w_{square}	0.12	1.4
$\varphi_{5,1} = F_{[0,5K]}(x \geq p_1 \vee G_{[0,250]} y \geq p_2)$	w_{sincos}	0.41	4.4
$\varphi_{5,2} = F_{[0,50K]}(x \geq p_1 \vee G[0,250] y \geq p_2)$		0.4	4.3
$\varphi_{6,1} = G_{[0,5K]} F_{[0,250]}((G_{[0,200]} x \leq p_1) \vee (G_{[0,200]} x \geq p_2))$	w_{sincos}	4	44
	w_{square}	0.75	8
$\varphi_{6,2} = G[0,50K] F_{[0,250]}((G_{[0,200]} x \leq p_1) \vee (G_{[0,200]} x \geq p_2))$	w_{sincos}	2.5	42
	w_{square}	0.67	8.7
$\varphi_{6,3} = G_{[0,50K]} F_{[0,125]}((G[0,200] x \leq p_1) \vee (G_{[0,200]} x \geq p_2))$	w_{sincos}	TO	TO
	w_{square}	0.66	8.1
$\varphi_7 = G \, F_{[0,45]} G_{[0,30]}(x_{cmd-resp} \leq p_1 \wedge x_{resp-cmd} \leq p_2)$	w_{pitch}	0.56	6

Future Work

- Optimized Pareto set implementation.
  - Large sets of points.
  - Operations: union and intersection or complement.
  - Sorted arrays in 2 dimensions, trees in 3 or more dimensions.
- Time parameters
  - Real time is not piecewise-constant; need more than just boxes.
  - Need fast set operations.

Future Work

- Optimized Pareto set implementation.
  - Large sets of points.
  - Operations: union and intersection or complement.
  - Sorted arrays in 2 dimensions, trees in 3 or more dimensions.
- Time parameters
  - Real time is not piecewise-constant; need more than just boxes.
  - Need fast set operations.

Thanks