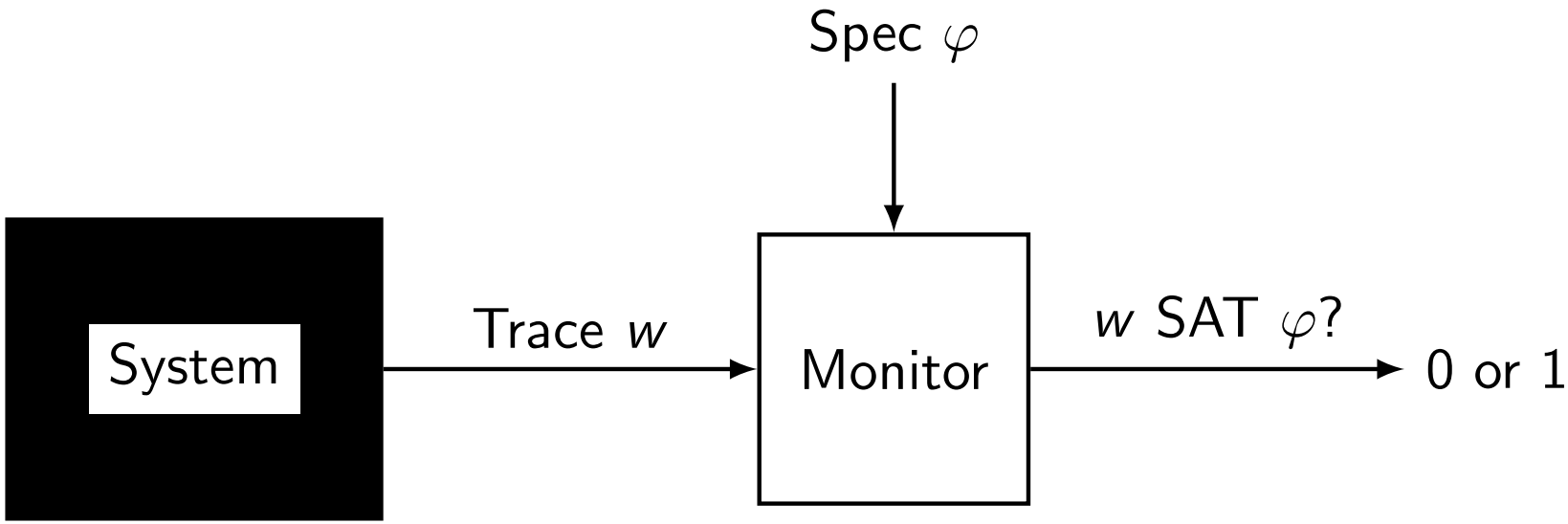
Towards Efficient Parametric Identification for STL

Alexey Bakhirkin Thomas Ferrère Oded Maler

Monitoring vs. Parametric Identification

Monitoring

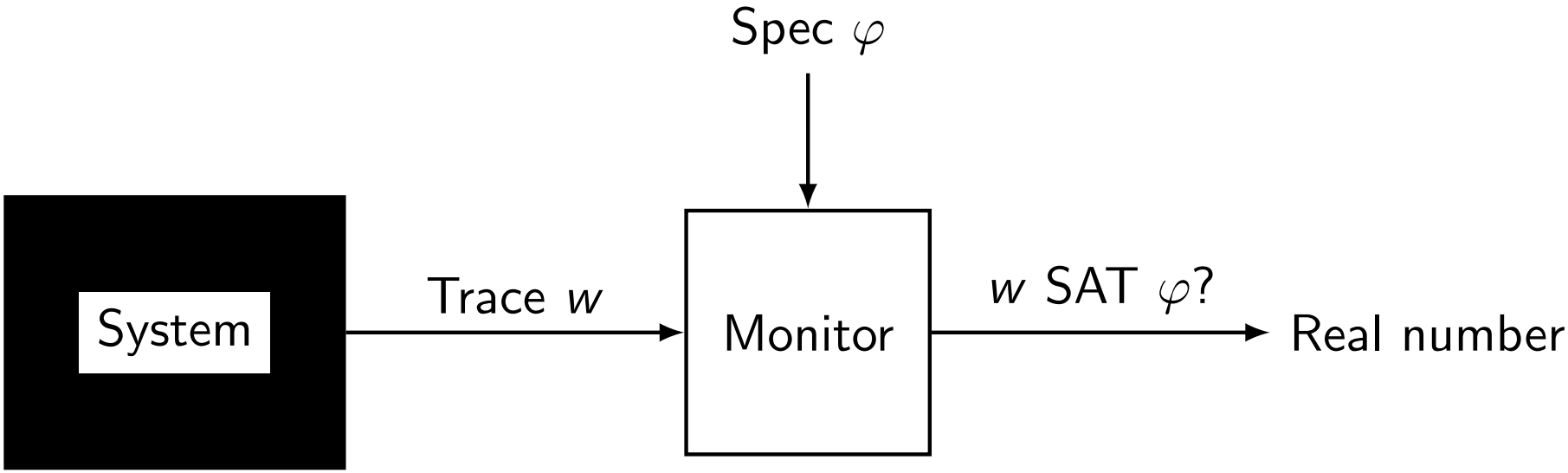
Do traces of a black box satisfy a property?



Monitoring vs. Parametric Identification

Monitoring

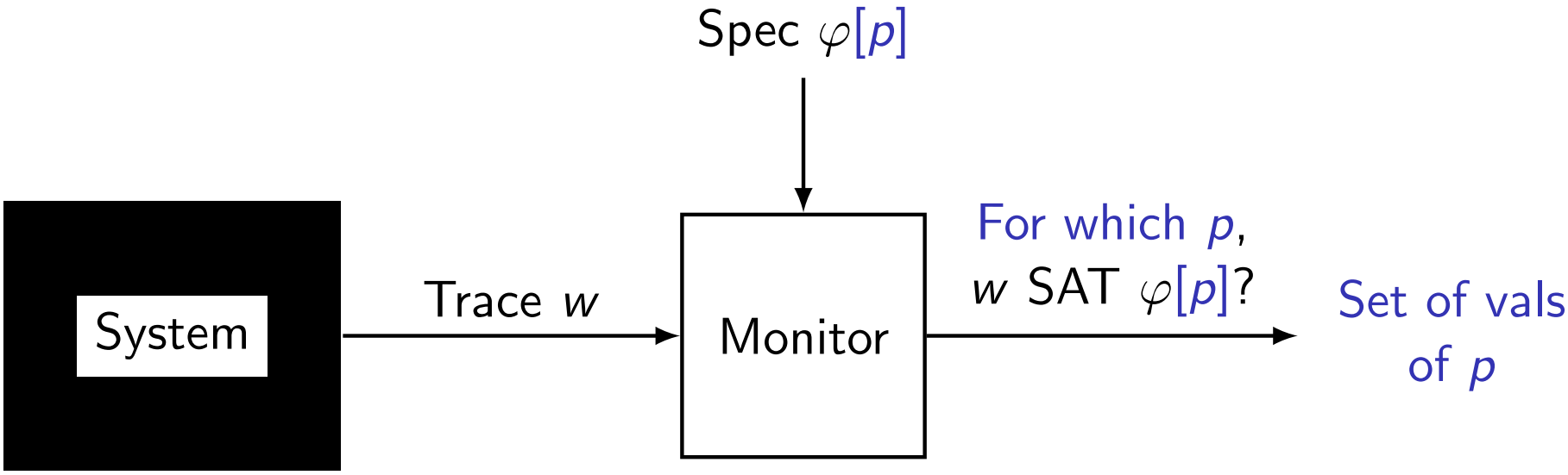
Do traces of a black box satisfy a property, and how well?



Monitoring vs. Parametric Identification

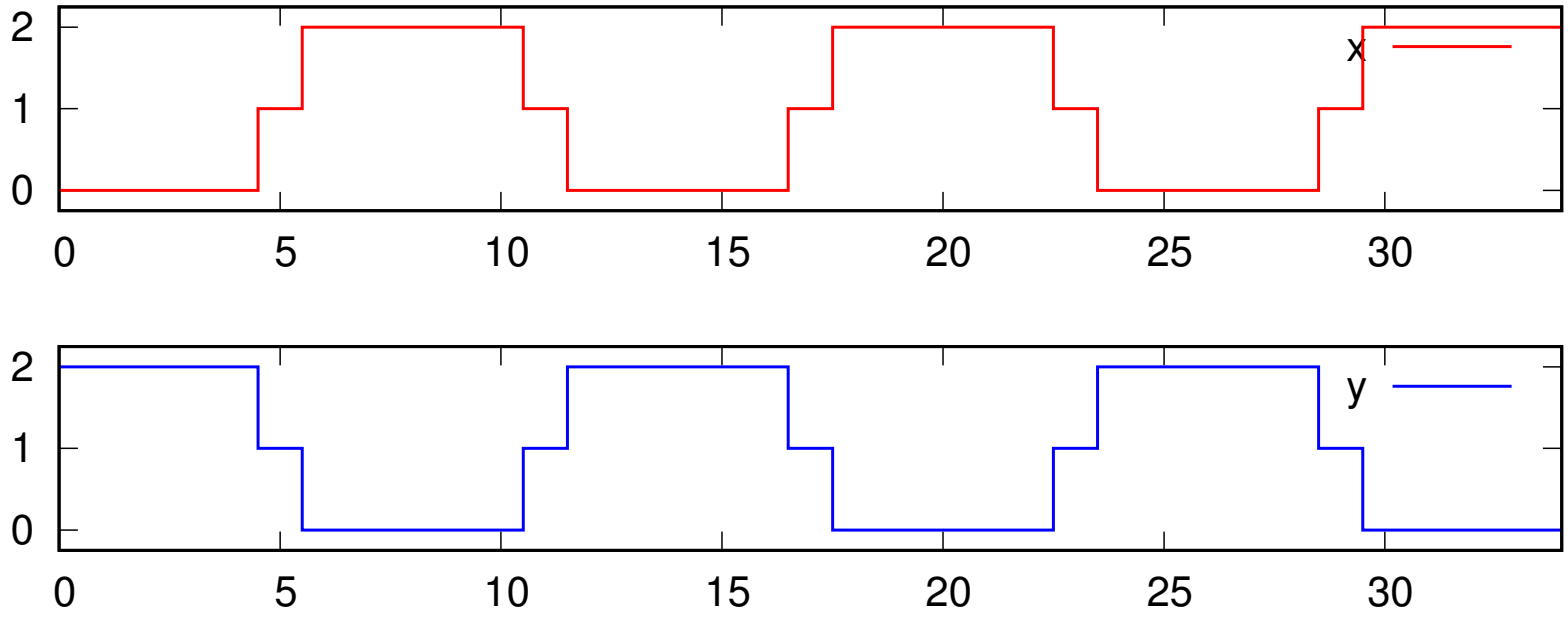
Parametric Identification

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 $[t, t']$.

- $G_{[a,b]}\,\varphi = \neg\, F_{[a,b]}\, \neg\varphi$.
- $\varphi_1\, U_{[a,b]}\, \varphi_2$ can be expressed as U and $F_{[a,b]}$.

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.

No time parameters yet – ongoing work.

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.

What PSTL Can Do?

Motivation

1. Find system parameters from system traces; applications in biology, automotive, etc.

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

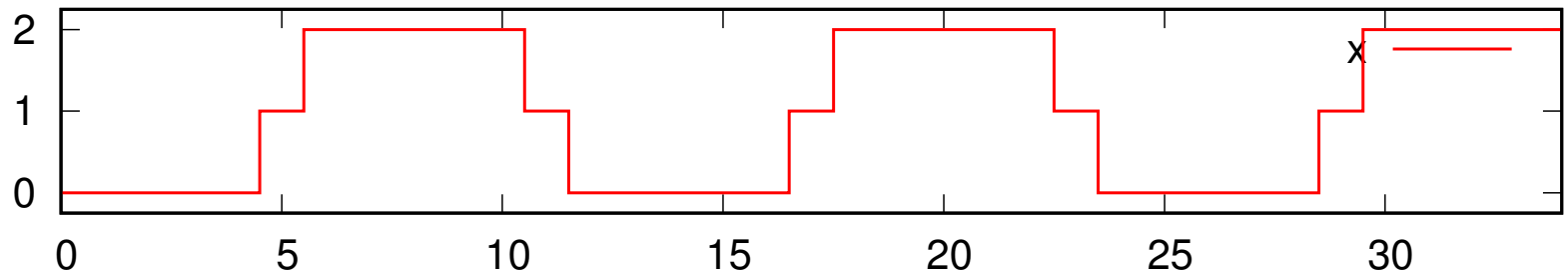
   For example,

   - $G(x \leq p_1 \wedge x \geq p_2)$ – 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. A way to compute robustness – turn constants in the formula into parameters, compute distance from original constants to the validity domain, see A. Rizk et al., CMSB 2008.
3. Evaluate formulas with universal/existential quantifiers. Not this work, see K. Havelund et al., FMCAD 2017.

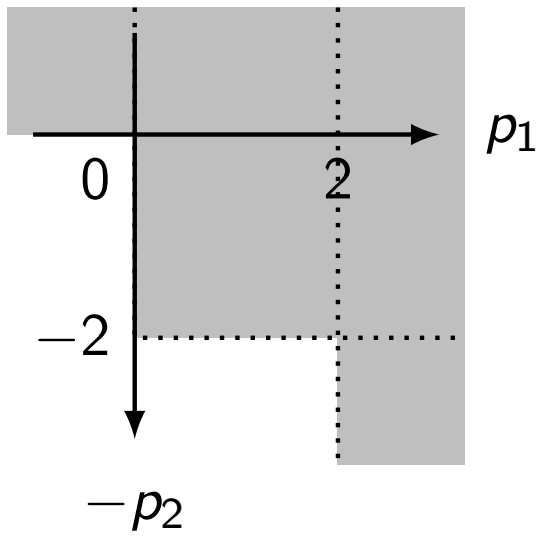
Example

Extract high and low thresholds



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

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

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

F. Fages, A. Rizk, Theor. Comp. Sci. 2008

Directly compute validity domains for LTL formulas.

Our Approach

Setting

- Piecewise-constant approximation.
- Single polarity.
- Time parameters – ongoing work.
- 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.

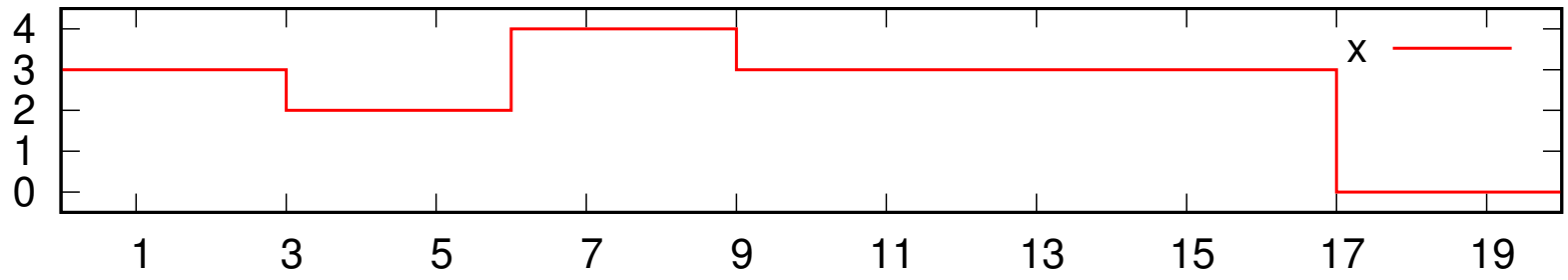
Compute validity signals directly as maps from time segments to sets of boxes.

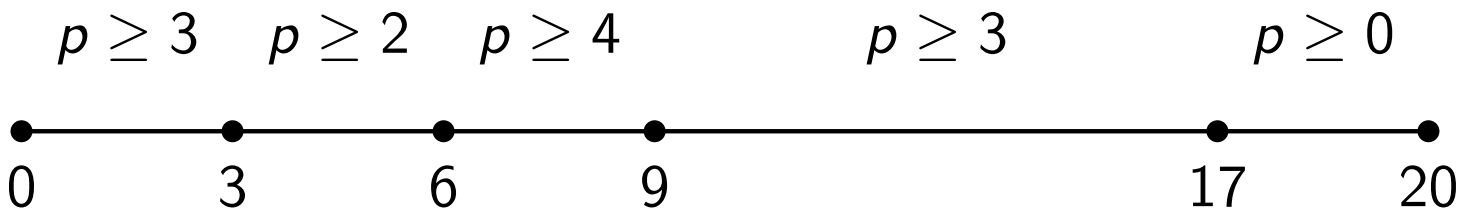
Bottom-up over the formula structure:

- $x \leq p$: 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

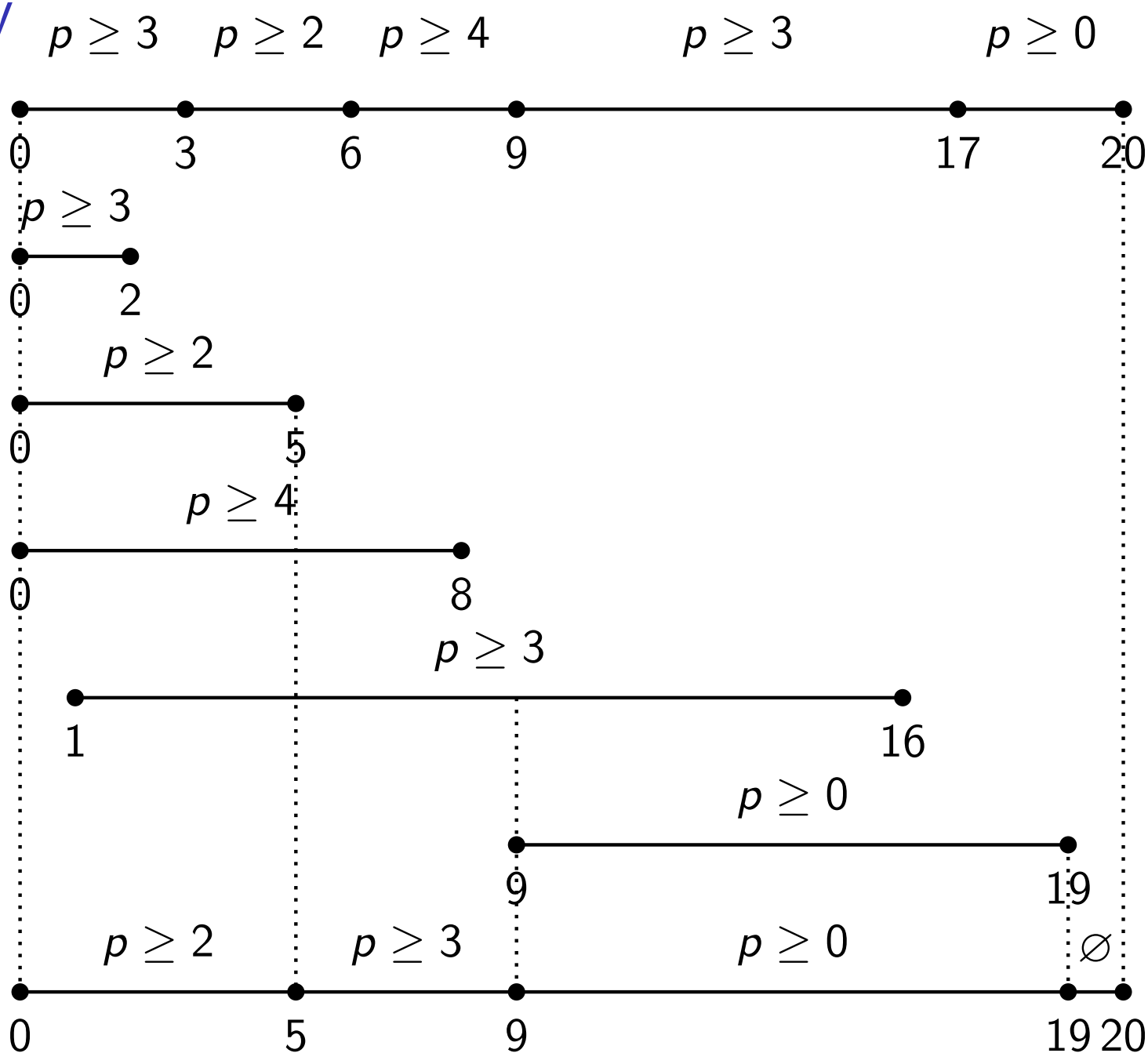
$x \leq p$





Eventually

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



Eventually

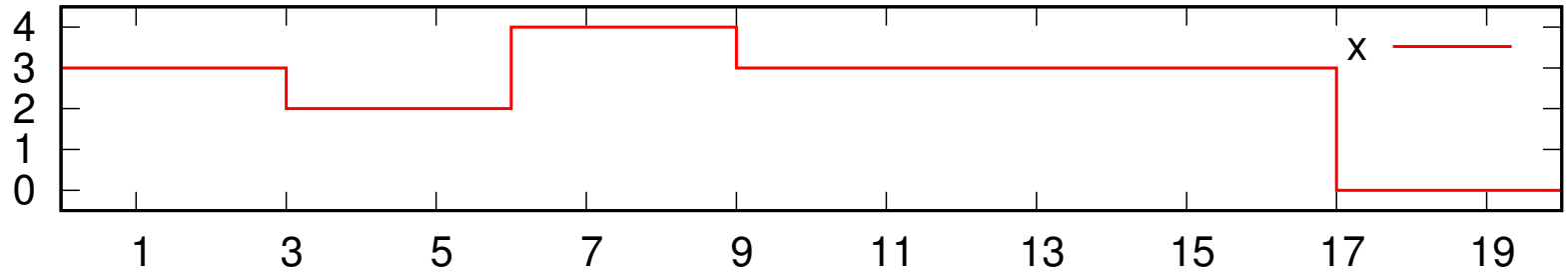
$F_{[1,8]}\, x \leq p$

- 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.

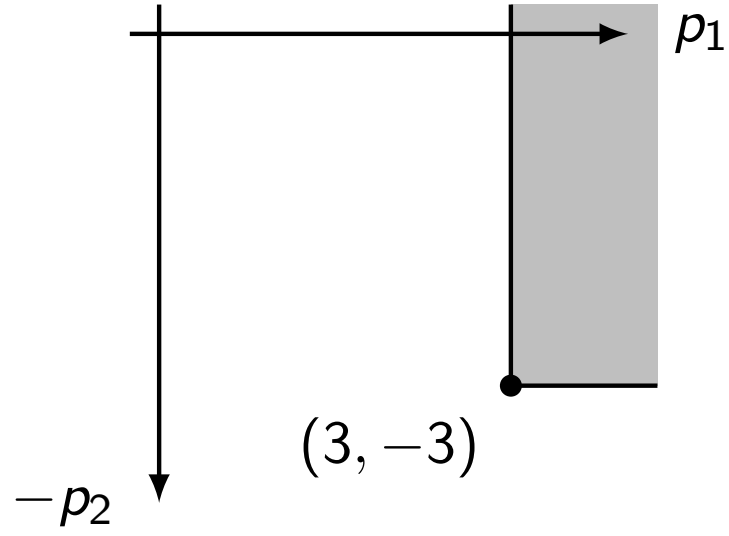
More Pessimistic Example

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

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



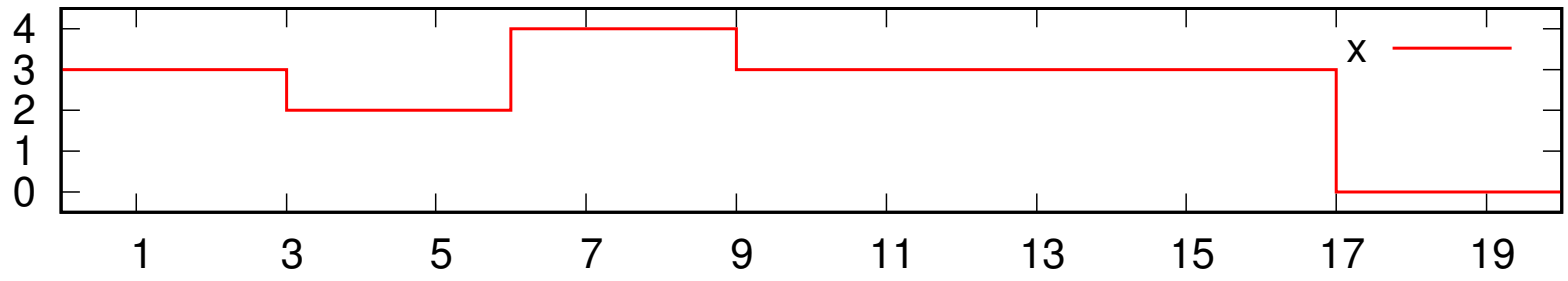
On $[0, 3]$ the validity domain is $p_1 \geq 3 \wedge -p_2 \geq -3$.



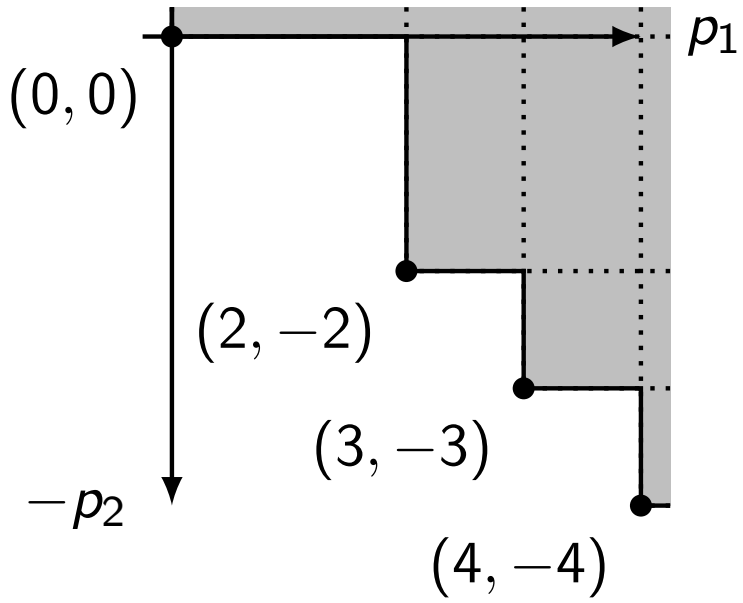
More Pessimistic Example

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

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



Validity domain at time 0 enumerates signal values.



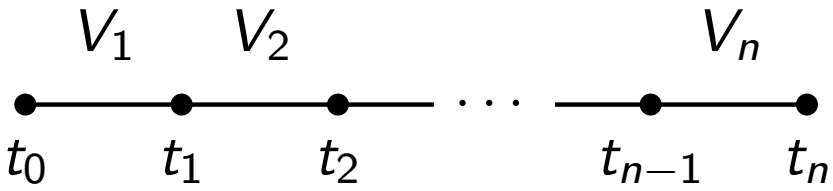
Evaluation

- 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.

Towards Time Parameters

Parametric Identification as Quantifier Elimination

A validity signal (for some formula φ):



The corresponding formula:

$$\bigvee_{i=1}^{n} (t_{i-1} \leq t < t_i) \wedge V_i$$

Validity signal of $F_{[a,b]}\,\varphi$ corresponds to:

$$\exists t'.\ (t + a \leq t' \leq t + b) \wedge \bigvee_{i=1}^{n} (t_{i-1} \leq t' < t_i) \wedge V_i$$

Towards Time Parameters

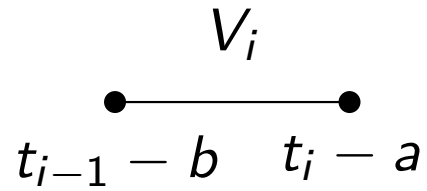
Focus on One Disjunct

$$\exists t'.\ (t + a \leq t' \leq t + b) \wedge (t_{i-1} \leq t' < t_i) \wedge V_i$$

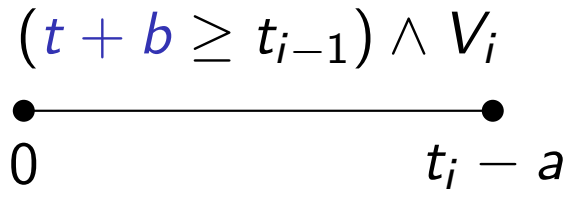
After elimination:

$$(t_{i-1} - b \leq t < t_i - a) \wedge V_i$$

Which is a back-shifted segment:



Or if b is not a constant:



Towards Time Parameters

- A validity domain is not a rectangle but a template polyhedron.
- Some dimensions correspond to linear expressions over absolute time and time parameters.
- Every temporal operator adds 1-2 dimensions and changes existing dimensions.

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.
- Especially important for time parameters.

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.
- Especially important for time parameters.

Thanks