The MonPoly Monitoring Tool

https://sourceforge.net/projects/monpoly/

Specification Language

Metric First-Order Temporal Logic (MFOTL)

$\varphi,\psi = t_1 < t_2 \mid t_1 = t_2 \mid p(t_1, \ldots, t_n)$
$\mid \neg\varphi \mid \varphi \lor \psi \mid \forall x.\ \varphi$
$\mid \circ_I\ \varphi \mid \varphi\ U_I\ \psi$
$\mid \bullet_I\ \varphi \mid \varphi\ S_I\ \psi$
$\mid [\omega_x \varphi](y;\ g_1, \ldots, g_n)$ –aggregations

ω = AVG | SUM | CNT | MIN | MAX | MED

+the usual syntactic sugar
(always, eventually, once, historically, …)

all future intervals are bounded

Features

implementation language: OCaml

algorithmic ideas:
- translation of temporal operators into incrementally updated auxiliary first-order predicates
- efficient sliding window algorithm
- waiting queue for future dependencies

two versions

finite relations	regular relations
- fast	- at least one order of magnitude slower
- syntactic restrictions on where negations may occur	- negation can occur freely

Example

signature

publish(x:string)
approve(x:string)

formula

publish(r) IMPLIES ONCE[0,7] approve(r)

informal policy

Every published report must have been approved within the last 7 time-units.

log / stream

- @1 approve ("A")
- @2 publish ("A")
- @3 publish ("A") ("B")
- @7 approve ("B")
- @3 (tp 2): ("B")
- @9 publish ("A")
- @10 publish ("A") ("B")
- @11
- @9 (tp 4): ("A")
- @10 (tp 5): ("A")

violations

Industrial Case Studies

NOKIA

- 5 million time-points
- 218 million tuples
- 5 GB of logs
- single core machine
- 20 min – 14 days per policy

monitor usage-control policies of highly sensitive cell phone location, call, and SMS data

example policy: The synchronization scripts must run for at least 1 second and for no longer than 6 hours.

Google

- 77.2 million time-points
- 26 billion tuples
- 400 GB of logs
- cluster of 1000 computers
- log slicing + MapReduce
- 2.5 – 12 hours per policy

monitor authentication policies in a network of 35000 computers used both within Google's corporate network and externally.

example policy: Long-running SSH sessions must not last longer than 24 hours.

The Online Monitoring Problem

START

property in a specification language

event stream →  → verdict stream

monitor

verdicts denote whether the property holds at **every** position in the event stream

not considered: instrumentation
(or how to generate the event stream)

considered setting: integer time-stamped events

important distinction: time-points vs time-stamps:
- time-points (indices in the event-stream)
- time-stamps (real-time information about events)

formulas specify real-time constraints via intervals I with integer bounds (or infinity)

assumptions on the sequence of time-stamps:
- non-decreasing (repeated time-stamps allowed)
- progressing (always eventually increasing)

Specification Language

Metric Dynamic Logic (MDL)

$\varphi,\psi = p \mid \neg\varphi \mid \varphi \lor \psi \mid \langle r \rangle_I\ \varphi \mid \varphi\ _I\langle r \rangle$

$r,s = \star \mid \varphi? \mid r + s \mid rs \mid r^*$

+the usual syntactic sugar
(until, next, since, previous,
always, eventually, once, historically, …)

more expressive than MTL
incomparable to MFOTL (propositional but regex)

future intervals may be unbounded

Features

implementation language: OCaml

algorithmic ideas:
- state update via dynamic programming
- future dependencies treated symbolically as variables in Boolean expressions
- different representations of Boolean expressions (explicit and BDD)
- keep only distinct Boolean expressions in memory

↓

almost event-rate independent memory consumption
(almost = logarithmic in the event-rate; in practice: constant)

Example

formula

$\langle \top^*\ \text{enter}\ \top^* \rangle_{[0,2]}$ exit

informal policy

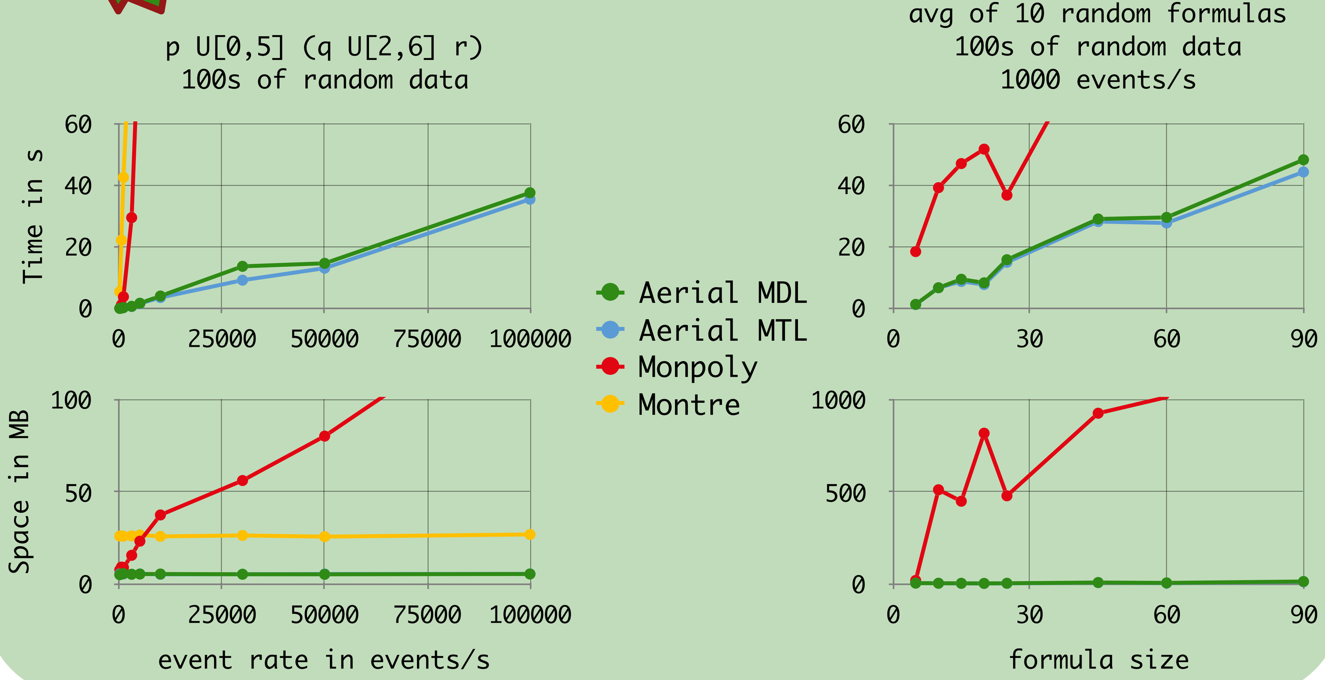
Within the next 2 time-units both "enter" and "exit" must happen and "enter" must happen before "exit".

log / stream

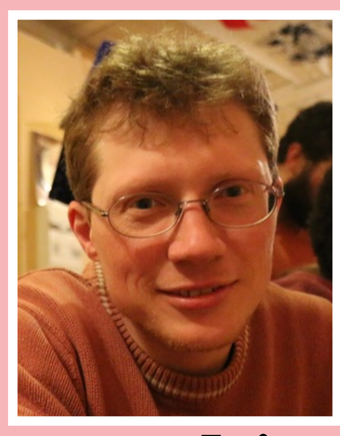
- @0
- @1 enter
- @2 enter exit
- @1:0 true
- @0:0 true
- @3 enter
- @3 enter
- @4 enter
- @3:1 = @3:0
- @6 exit
- @2:0 false
- @3:0 false
- @4:0 true

verdicts

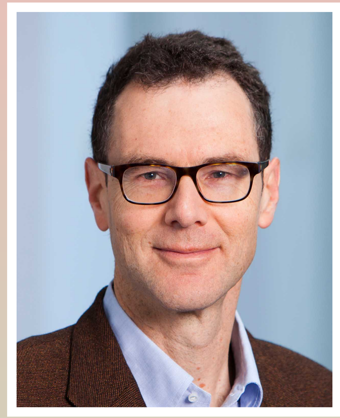
Evaluation



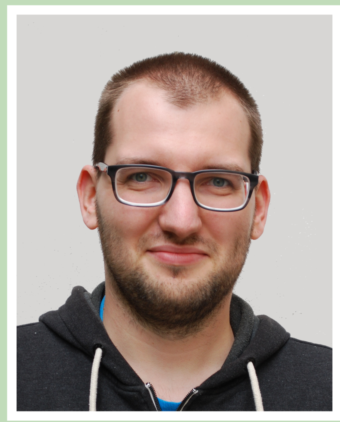
Aerial: Almost Event-Rate Independent Algorithms for Monitoring Metric Regular Properties


Eugen Zălinescu
Technische Universität München


Felix Klaedtke
NEC


David Basin


Srđan Krstić


Dmitriy Traytel
ETH zürich

75 NRP Big Data National Research Programme

AERIAL

https://bitbucket.org/traytel/aerial