Topics in Timed Automata

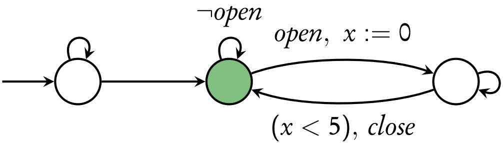
B. Srivathsan

RWTH-Aachen

Software modeling and Verification group

Model-Checking Real-Time Systems

Correctness: Safety + **Liveness** + **Fairness**



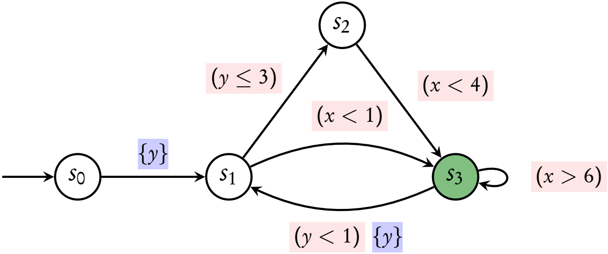
"Infinitely often, the gate is open for at least 5 s."

Realistic counter-examples: infinite **non-Zeno** runs

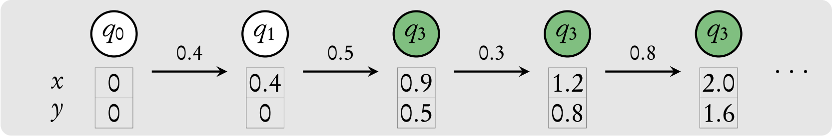
Lecture 8:

Non-Zenoness

Timed Büchi automata



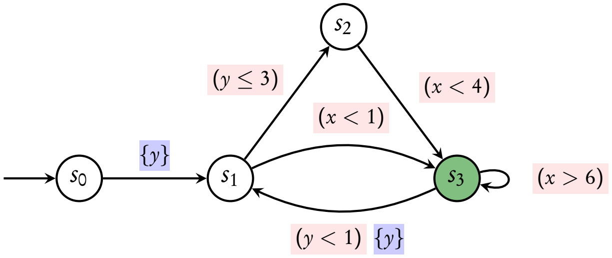
Run: infinite sequence of transitions



- **accepting** if infinitely often green state
- **non-Zeno** if time diverges ($\sum_{i \geq 0} \delta_i \to \infty$)

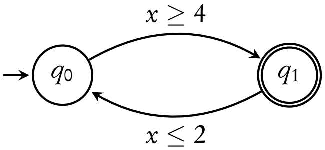
Büchi non-emptiness problem

Given a TBA, does it **have** a **non-Zeno** accepting run

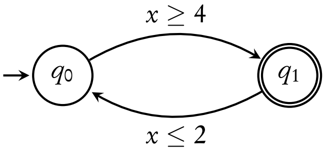


Theorem [AD94]

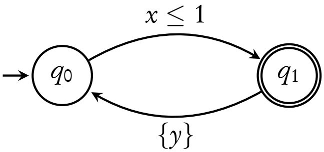
This problem is **PSPACE-complete**



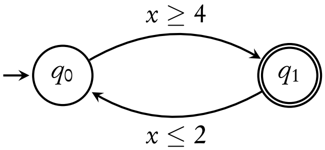
No infinite run



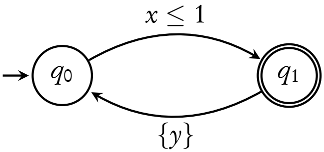
No infinite run



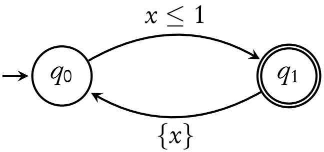
No non-Zeno run



No infinite run



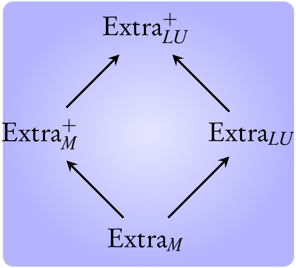
No non-Zeno run



Non-Zeno run $\surd$

How do we detect **infinite non-Zeno** runs given an automaton?

Abstract zone graphs again



$$ZG^{\mathfrak{a}}(\mathcal{A}): \quad (q_0, Z_0) \rightarrow (q_1, Z_1) \rightarrow (q_2, Z_2) \longrightarrow \cdots$$

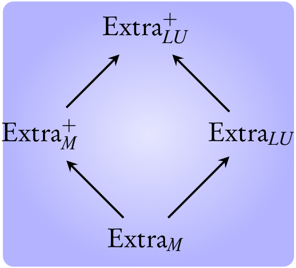
$$\qquad\qquad \Cup \qquad\qquad \Cup \qquad\qquad \Cup$$

$$\mathcal{A}: \quad (q_0, v_0) \rightarrow (q_1, v_1) \rightarrow (q_2, v_2) \longrightarrow \cdots$$

Sound and complete [Tri09, Li09]

All the above abstractions preserve **repeated state reachability**

Abstract zone graphs again



$$ZG^{\mathfrak{a}}(\mathcal{A}): \quad (q_0, Z_0) \rightarrow (q_1, Z_1) \rightarrow (q_2, Z_2) \longrightarrow \cdots$$

$$\Cup \qquad \Cup \qquad \Cup$$

$$\mathcal{A}: \quad (q_0, v_0) \rightarrow (q_1, v_1) \rightarrow (q_2, v_2) \longrightarrow \cdots$$

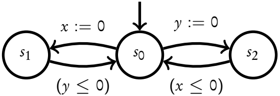
Sound and complete [Tri09, Li09]

All the above abstractions preserve **repeated state reachability**

What about **non-Zenoness**?

Time progress criterion [AD94]

$\bigwedge_{x \in X} \text{unbounded}(x) \vee \text{fluctuating}(x)$



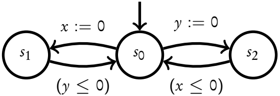
Region graph:

$(s_1, 0 = x < y)$ $\quad$ $(s_2, 0 = y < x)$

$(s_0, 0 = x = y) \longrightarrow (s_1, 0 = x = y) \longrightarrow (s_0, 0 = x = y) \longrightarrow (s_2, 0 = y = x) \dashrightarrow$

Time progress criterion [AD94]

$\bigwedge_{x \in X} \text{unbounded}(x) \vee \text{fluctuating}(x)$



Region graph:

$$(s_1, 0 = x < y) \qquad (s_2, 0 = y < x)$$

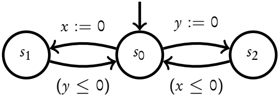
$$(s_0, 0 = x = y) \longrightarrow (s_1, 0 = x = y) \longrightarrow (s_0, 0 = x = y) \longrightarrow (s_2, 0 = y = x) \dashrightarrow$$

Zone graph with Extra^+_M:

$$(s_0, 0 \leq x = y) \longrightarrow (s_1, 0 \leq x \leq y) \longrightarrow (s_0, 0 \leq x = y) \longrightarrow (s_2, 0 \leq y \leq x) \dashrightarrow$$

Time progress criterion [AD94]

$\bigwedge_{x \in X} \text{unbounded}(x) \vee \text{fluctuating}(x)$



Region graph:

$(s_1, 0 = x < y)$ $\qquad$ $(s_2, 0 = y < x)$

$(s_0, 0 = x = y) \longrightarrow (s_1, 0 = x = y) \longrightarrow (s_0, 0 = x = y) \longrightarrow (s_2, 0 = y = x) \dashrightarrow$

Zone graph with Extra_M^+:

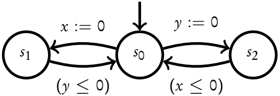
$(s_0, 0 \leq x = y) \longrightarrow (s_1, 0 \leq x \leq y) \longrightarrow (s_0, 0 \leq x = y) \longrightarrow (s_2, 0 \leq y \leq x) \dashrightarrow$

Zone graph with Extra_{LU}^+:

$(s_0, \top) \longrightarrow (s_1, \top) \longrightarrow (s_0, \top) \longrightarrow (s_2, \top) \dashrightarrow$

Time progress criterion [AD94]

$\bigwedge_{x\in X}$ unbounded$(x) \vee$ fluctuating(x)



Region graph:

$(s_1, 0 = x < y)$ $\qquad$ $(s_2, 0 = y < x)$

$(s_0, 0 = x = y) \longrightarrow (s_1, 0 = x = y) \longrightarrow (s_0, 0 = x = y) \longrightarrow (s_2, 0 = y = x) \dashrightarrow$

Zone graph with Extra_M^+:

$(s_0, 0 \leq x = y) \longrightarrow (s_1, 0 \leq x \leq y) \longrightarrow (s_0, 0 \leq x = y) \longrightarrow (s_2, 0 \leq y \leq x) \dashrightarrow$

Zone graph with Extra_{LU}^+:

$(s_0, \top) \longrightarrow (s_1, \top) \longrightarrow (s_0, \top) \longrightarrow (s_2, \top) \dashrightarrow$

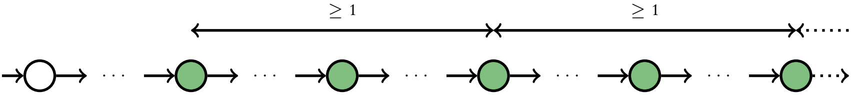
The **time progress** criterion is **not sound** on **zones**

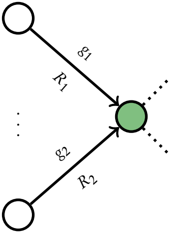
Coming next...

Strongly non-Zeno construction [TYB05]

From TBA to Strongly non-Zeno TBA

Key Idea : reduce non-Zenoness to Büchi acceptation

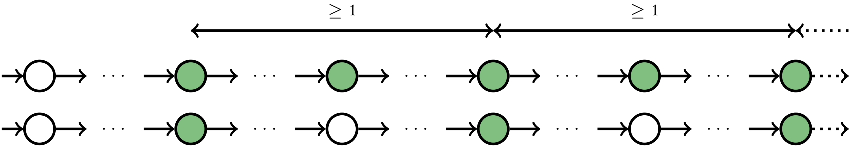


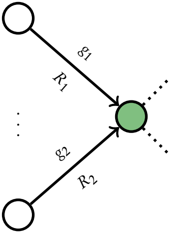


A

From TBA to Strongly non-Zeno TBA

Key Idea : reduce non-Zenoness to Büchi acceptation

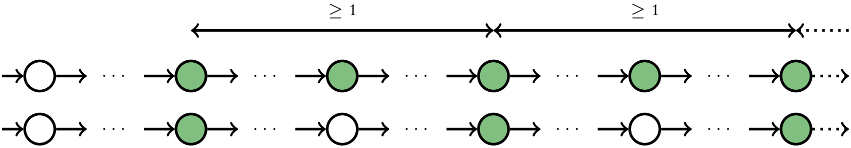


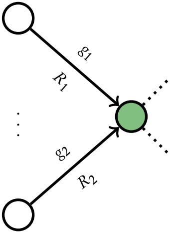


A

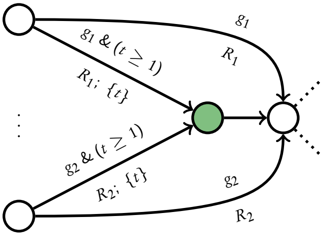
From TBA to Strongly non-Zeno TBA

Key Idea : reduce non-Zenoness to Büchi acceptation





A



A′

Adding a clock for non-Zenoness [TYB05]

$\mathbf{A}'$: strongly non-Zeno TBA
$|X| + 1$ clocks and at most $2 \cdot |Q|$ states

Theorem [TYB05]

$\mathbf{A}$ has a non-Zeno accepting run iff $ZG^{\mathfrak{a}}(\mathbf{A}')$ has an **accepting** run

Adding a clock for non-Zenoness [TYB05]

$\mathbf{A}'$: strongly non-Zeno TBA

$|X| + 1$ clocks and at most $2 \cdot |Q|$ states

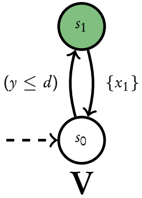
Theorem [TYB05]

$\mathbf{A}$ has a non-Zeno accepting run iff $ZG^{\mathfrak{a}}(\mathbf{A}')$ has an **accepting** run

Question: Is this good enough?

Adding one clock leads to an **exponential blowup** in the zone graph! [HSW12]

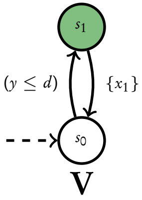
Guard $t \geq 1$ Allows to Count...

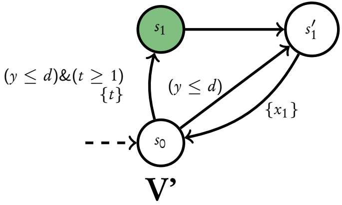


Run of **V**: 2 different zones in s_0

$$\cdots (s_0, y \leq x_1 \leq x_2) \xrightarrow{y \leq d} (s_1, y \leq x_1 \leq x_2 \,\&\, y \leq d) \xrightarrow{\{x_1\}}$$
$$(s_0, 0 = x_1 \leq y \leq x_2) \xrightarrow{y \leq d} (s_1, x_1 \leq y \leq x_2 \,\&\, y \leq d) \xrightarrow{\{x_1\}}$$
$$(s_0, 0 = x_1 \leq y \leq x_2) \cdots$$

Guard $t \geq 1$ Allows to Count...





Run of **V'**: $d + 2$ different zones in s_0

$$\cdots (s_0, y \leq x_1 \leq x_2 \leq t) \xrightarrow{(y \leq d)\&(t \geq 1),\, t:=0} \rightarrow \xrightarrow{\{x_1\}}$$

$$(s_0, 0 = x_1 \leq t \leq y \leq x_2 \& \mathbf{y} - \mathbf{t} \geq 0) \xrightarrow{(y \leq d)\&(t \geq 1),\, t:=0} \rightarrow \xrightarrow{\{x_1\}}$$

$$(s_0, 0 = x_1 \leq t \leq y \leq x_2 \& \mathbf{y} - \mathbf{t} \geq \mathbf{1}) \xrightarrow{(y \leq d)\&(t \geq 1),\, t:=0} \rightarrow \xrightarrow{\{x_1\}}$$

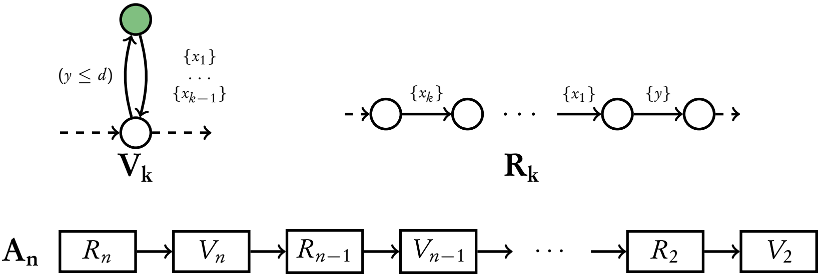
$$(s_0, 0 = x_1 \leq t \leq y \leq x_2 \& \mathbf{y} - \mathbf{t} \geq \mathbf{2}) \xrightarrow{(y \leq d)\&(t \geq 1),\, t:=0} \rightarrow \xrightarrow{\{x_1\}}$$

$$\cdots$$

$$(s_0, 0 = x_1 \leq t \leq y \leq x_2 \& \mathbf{y} - \mathbf{t} \geq \mathbf{d})$$

Remark: $y - t \geq c$ implies $x_2 - x_1 \geq c$

...and Leads to a Combinatorial Explosion

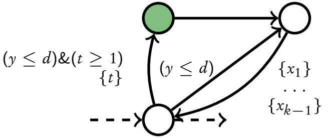


Lemma

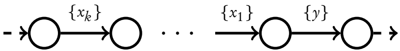
$ZG^{\mathfrak{a}}(A_n)$ has linear size in n

Key Idea: at V_k only two possible zones that **collapse** to the same zone after R_{k-1}.

...and Leads to a Combinatorial Explosion

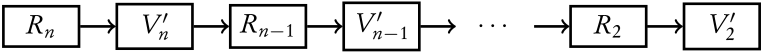


$\mathbf{V'_k}$



$\mathbf{R_k}$

$\mathbf{A'_n}$



Lemma

$ZG^{\mathfrak{a}}(A'_n)$ has size exponential in n

Key Idea: at V'_k, $\bigwedge_{i \in [k;n]} x_i - x_{i-1} \geq c_i$ with $c_i \in [0; d]$ chosen **non-deterministically**

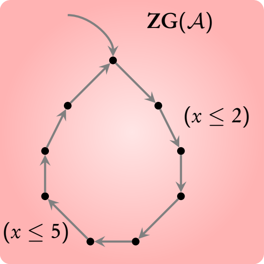
What we have:

- $ZG^{\mathfrak{a}}(A_n)$ has size $\mathcal{O}(n)$
- $ZG^{\mathfrak{a}}(A'_n)$ has size $\mathcal{O}(2^n)$

Coming next:

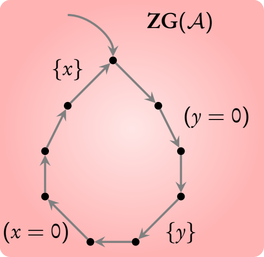
A $|ZG^{\mathfrak{a}}(A_n)|.\mathcal{O}(|X|^2)$ algorithm [HSW12]

When does a path in $ZG(\mathcal{A})$ yield only Zeno runs?



Blocking clocks

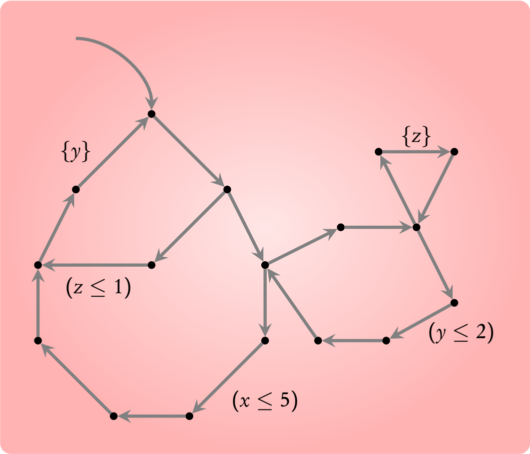
x never reset but checked for upper bound



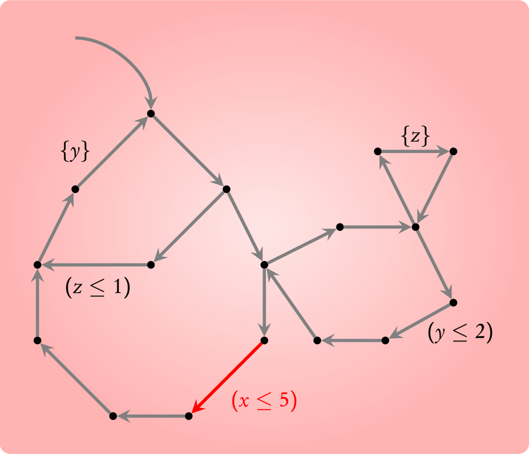
Zero-checks

x and y should be 0 all along the path

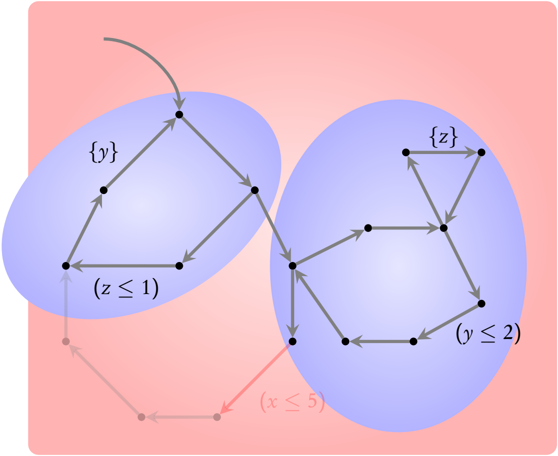
Blocking clocks



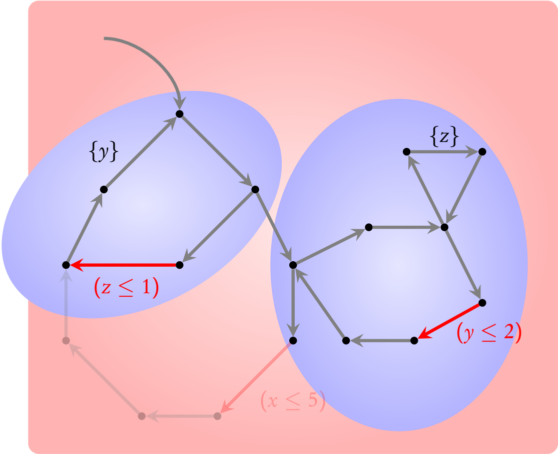
Blocking clocks



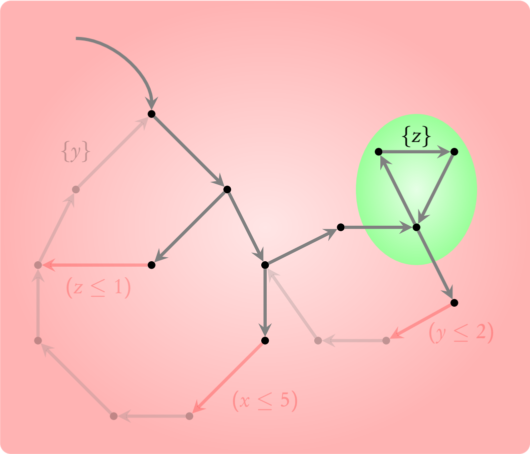
Blocking clocks



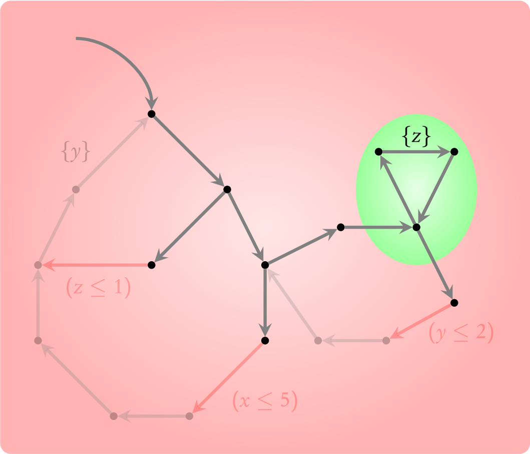
Blocking clocks



Blocking clocks



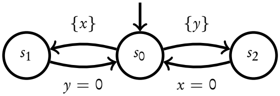
Blocking clocks



Theorem

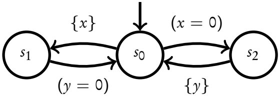
Blocking clocks can be detected in $|\mathrm{ZG}^{\mathfrak{a}}(\mathcal{A})| \cdot (|X| + 1)$ time

The case of zero checks



$$s_0 \xrightarrow{\{x\}} s_1 \xrightarrow{(y=0)} s_0 \xrightarrow{\{y\}} s_2 \xrightarrow{(x=0)} s_0$$

All states are in the scope of a zero check!



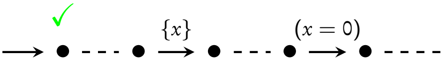
$$s_0 \xrightarrow{\{x\}} s_1 \xrightarrow{(y=0)} s_0 \xrightarrow{(x=0)} s_2 \xrightarrow{\{y\}} s_0$$

State s_2 is **clear**: all zero-checks are **preceded** by resets!

Zero-checks

Can time elapse here?

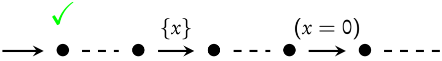
Zero-checks



Time can elapse at a node if

every zero-check is **preceded** by a reset

Zero-checks



Time can elapse at a node if

every zero-check is **preceded** by a reset

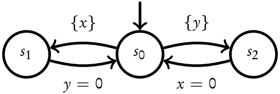
Guessing Zone Graph ($GZG^{\mathfrak{a}}(\mathcal{A})$) :

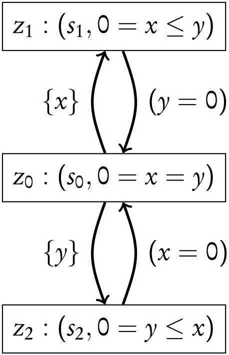
$$(q, Z, Y) \xrightarrow{\{x\}} (q', Z', Y \cup \{x\})$$

$$(q, Z, Y) \xrightarrow{(x=0)} \text{enabled only if } x \in Y$$

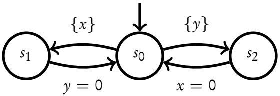
$$(q, Z, Y) \xrightarrow{\tau} (q, Z, \emptyset)$$

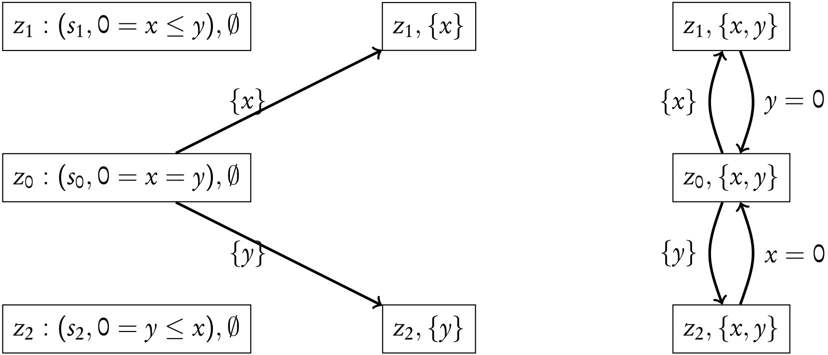
Zero checks (1st example)



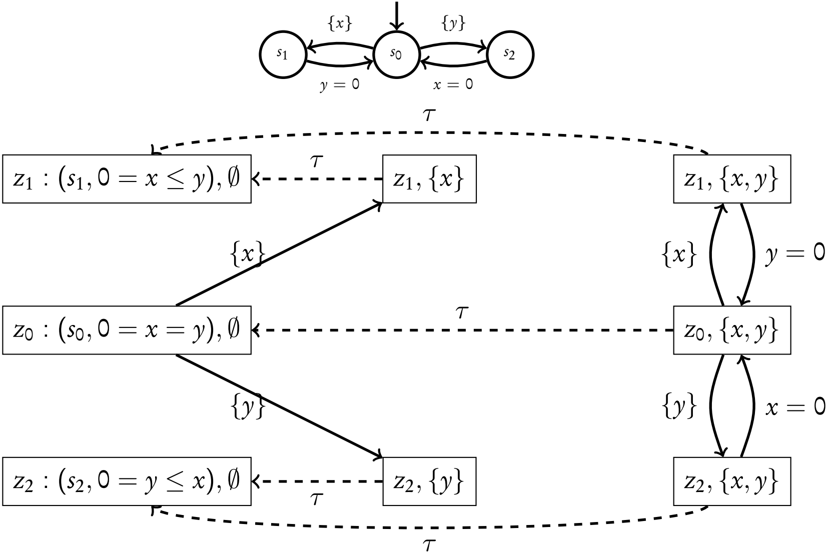


Zero checks (1st example)

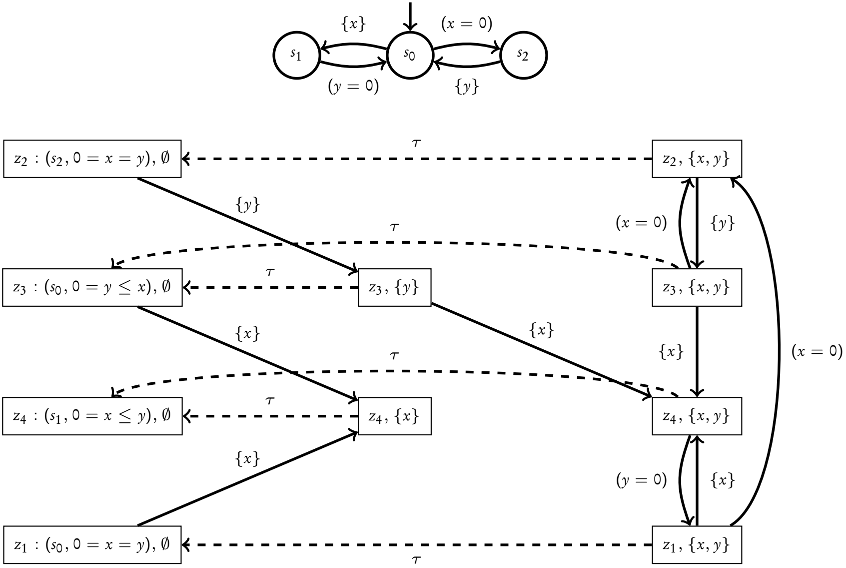




Zero checks (1st example)



Zero checks (2nd example)



Algorithm

Theorem [HSW12]

A has a non-Zeno run iff there is an **SCC** in $\mathrm{GZG}^{\mathfrak{a}}(A)$ that contains:

- an **accepting** node
- **no blocking** clocks
- a **clear** node $(q, Z, \emptyset)$

Complexity: $|\,\mathrm{GZG}^{\mathfrak{a}}(A)| \cdot (|X| + 1)$

$2^{|X|}$ **more nodes** in $\mathrm{GZG}^{\mathfrak{a}}(A)$ than in $\mathrm{ZG}^{\mathfrak{a}}(A)$ **due to** Y **sets?**

$2^{|X|}$ **more nodes** in $\mathrm{GZG}^{\mathfrak{a}}(A)$ than in $\mathrm{ZG}^{\mathfrak{a}}(A)$ **due to** Y **sets?**

Theorem

- For each reachable node (q, Z), Z entails a **total order** on X.
- Extra_M, Extra_M^+ **preserve the order**.
- Y **respects** this order; only $|X| + 1$ sets needed.

$2^{|X|}$ **more nodes** in $\mathrm{GZG}^{\mathfrak{a}}(A)$ than in $\mathrm{ZG}^{\mathfrak{a}}(A)$ **due to** Y **sets?**

Theorem

- For each reachable node (q, Z), Z entails a **total order** on X.
- Extra_M, Extra_M^+ **preserve the order**.
- Y **respects** this order; only $|X| + 1$ sets needed.

Extra_{LU}, Extra_{LU}^+ **do not preserve order**

Theorem [HS11]

Non-Zenoness from LU-abstract zone graphs is **NP-complete**

Theorem [HS11]

A **slight weakening** of Extra_{LU}, Extra_{LU}^+ **preserves** order

Benchmarks

A	$ZG^{\mathfrak{a}}(A)$	$ZG^{\mathfrak{a}}(A')$		$GZG^{\mathfrak{a}}(A)$		
	size	size	otf	size	otf	opt
Train-Gate2 (mutex)	134	194	194	400	400	134
Train-Gate2 (bound. resp.)	988	227482	352	3840	1137	292
Train-Gate2 (liveness)	100	217	35	298	53	33
Fischer3 (mutex)	1837	3859	3859	7292	7292	1837
Fischer4 (mutex)	46129	96913	96913	229058	229058	46129
Fischer3 (liveness)	1315	4962	52	5222	64	40
Fischer4 (liveness)	33577	147167	223	166778	331	207
FDDI3 (liveness)	508	1305	44	3654	79	42
FDDI5 (liveness)	6006	15030	90	67819	169	88
FDDI3 (bound. resp.)	6252	41746	59	52242	114	60
CSMA/CD4 (collision)	4253	7588	7588	20146	20146	4253
CSMA/CD5 (collision)	45527	80776	80776	260026	260026	45527
CSMA/CD4 (liveness)	3038	9576	1480	14388	3075	832
CSMA/CD5 (liveness)	32751	120166	8437	186744	21038	4841

- Combinatorial explosion may **occur** in practice
- **Optimized** use of $GZG^{\mathfrak{a}}(A)$ gives best results

Conclusion

- Strongly non-Zeno construction can cause **exponential blowup**

- A **guessing zone graph** construction for non-Zenoness

Bibliography I

R. Alur and D.L. Dill.
A theory of timed automata.
Theoretical Computer Science, 126(2):183–235, 1994.

G. Behrmann, P. Bouyer, E. Fleury, and K. G. Larsen.
Static guard analysis in timed automata verification.
In *TACAS'03*, volume 2619 of *LNCS*, pages 254–270. Springer, 2003.

G. Behrmann, P. Bouyer, K. Larsen, and R. Pelánek.
Lower and upper bounds in zone based abstractions of timed automata.
Tools and Algorithms for the Construction and Analysis of Systems, pages 312–326, 2004.

G. Behrmann, P. Bouyer, K. G. Larsen, and R. Pelanek.
Lower and upper bounds in zone-based abstractions of timed automata.
Int. Journal on Software Tools for Technology Transfer, 8(3):204–215, 2006.

B. Bérard, B. Bouyer, and A. Petit.
Analysing the pgm protocol with UPPAAL.
Int. Journal of Production Research, 42(14):2773–2791, 2004.

G. Behrmann, A. David, K. G Larsen, J. Haakansson, P. Pettersson, W. Yi, and M. Hendriks.
Uppaal 4.0.
In *QEST'06*, pages 125–126, 2006.

M. Bozga, C. Daws, O. Maler, A. Olivero, S. Tripakis, and S. Yovine.
Kronos: a mode-checking tool for real-time systems.
In *CAV'98*, volume 1427 of *LNCS*, pages 546–550. Springer, 1998.

Bibliography II

P. Bouyer.
Untameable timed automata!
STACS 2003, pages 620–631, 2003.

P. Bouyer.
Forward analysis of updatable timed automata.
Form. Methods in Syst. Des., 24(3):281–320, 2004.

C. Courcoubetis and M. Yannakakis.
Minimum and maximum delay problems in real-time systems.
Form. Methods Syst. Des., 1(4):385–415, 1992.

D. Dill.
Timing assumptions and verification of finite-state concurrent systems.
In *AVMFSS*, volume 407 of *LNCS*, pages 197–212. Springer, 1989.

C. Daws and S. Tripakis.
Model checking of real-time reachability properties using abstractions.
In *TACAS'98*, volume 1384 of *LNCS*, pages 313–329. Springer, 1998.

Extended version: Using non-convex approximations for efficient analysis of timed automata.
`http://www.labri.fr/~sri/Papers/cav2011extended.pdf`.

R. Gómez and H. Bowman.
Efficient detection of zeno runs in timed automata.
In *Proc. 5th Int. Conf. on Formal Modeling and Analysis of Timed Systems, FORMATS 2007*, volume 4763 of *LNCS*, pages 195–210, 2007.

Bibliography III

M. Hendriks, G. Behrmann, K. G. Larsen, P. Niebert, and F. Vaandrager.
Adding symmetry reduction to Uppaal.
In *Int. Workshop on Formal Modeling and Analysis of Timed Systems*, volume 2791 of *LNCS*, pages 46–59. Springer, 2004.

F. Herbreteau and B. Srivathsan.
Coarse abstractions make zeno behaviours difficult to detect.
In *CONCUR*, volume 6901 of *LNCS*, pages 92–107, 2011.

K. Havelund, A. Skou, K. Larsen, and K. Lund.
Formal modeling and analysis of an audio/video protocol: An industrial case study using UPPAAL.
In *RTSS'97*, pages 2–13, 1997.

F. Herbreteau, B. Srivathsan, and I. Walukiewicz.
Efficient emptiness check for timed büchi automata.
Formal Methods in System Design, 40(2):122–146, 2012.

J. J. Jessen, J. I. Rasmussen, K. G. Larsen, and A. David.
Guided controller synthesis for climate controller using UPPAAL TiGA.
In *FORMATS'07*, volume 4763, pages 227–240. Springer, 2007.

Guangyuan Li.
Checking timed büchi automata emptiness using lu-abstractions.
In Joël Ouaknine, editor, *Formal modeling and analysis of timed systems. 7th Int. Conf. (FORMATS)*, volume 5813 of *Lecture Notes in Computer Science*, pages 228–242. Springer, 2009.

François Laroussinie and Ph. Schnoebelen.
The state explosion problem from trace to bisimulation equivalence.
In *Proceedings of the Third International Conference on Foundations of Software Science and Computation Structures*, FOSSACS '00, pages 192–207. Springer-Verlag, 2000.

Bibliography IV

S. Tripakis.
Verifying progress in timed systems.
In *Proc. 5th Int. AMAST Workshop, ARTS'99*, volume 1601 of *LNCS*, pages 299–314. Springer, 1999.

S. Tripakis.
Checking timed büchi emptiness on simulation graphs.
ACM Transactions on Computational Logic, 10(3):??–??, 2009.

S. Tripakis, S. Yovine, and A. Bouajjani.
Checking timed büchi automata emptiness efficiently.
Formal Methods in System Design, 26(3):267–292, 2005.

J. Zhao, X. Li, and G. Zheng.
A quadratic-time dbm-based successor algorithm for checking timed automata.
Inf. Process. Lett., 96(3):101–105, 2005.