Accelerated Model-Checking for Real-Time Systems

Frédéric Herbreteau
LaBRI, Université de Bordeaux - CNRS, FRANCE

Séminaire DTIM Onera, Toulouse
31st may 2010

Plan

Plan

Plan

- Model-Checking Real-Time Systems
  - Linear Hybrid Automata
  - Symbolic Semantics: Linear Hybrid Relations
- Accelerated Model-Checking
  - Symbolic Model-Checking
  - Meta-Transitions
- Periodic Acceleration
  - Transitive Closure for Periodic LHR
  - Dealing with Ultimate Periodicity
- Conclusions and Future Work

Goal

Automatically check **safety** requirements for **real-time systems**.

Example

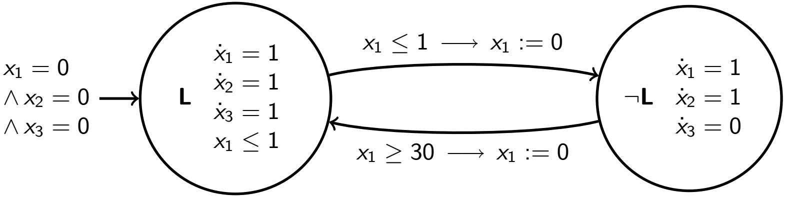
The "Leaking Gas Burner" [CHR91]:

"Whenever the gas burner is used for at least 60s. and provided that it leaks for at most 1s. every 30s., then the accumulated leaking time does not exceed 1/20th of total elapsed time"

Model-checking based on the computation of **exact loop invariants**

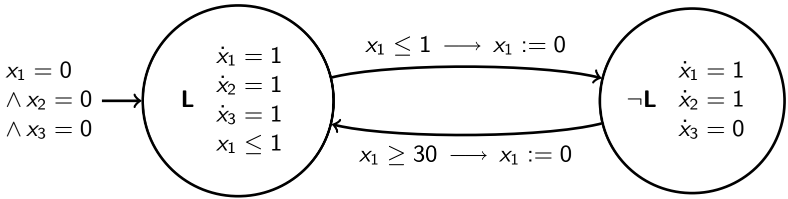
Modeling the "Leaking Gas Burner"

"Whenever the gas burner is used for at least 60s. and provided that it leaks for at most 1s. every 30s., then the accumulated leaking time does not exceed 1/20th of total elapsed time"

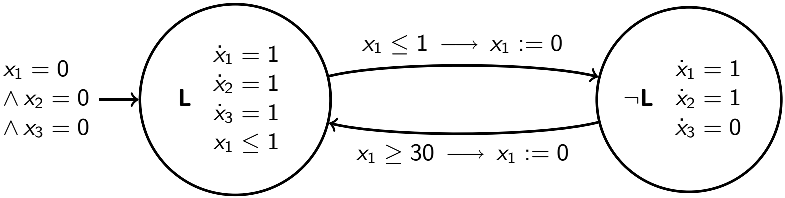


Check the invariance of $(x_2 \geq 60 \Rightarrow 20x_3 \leq x_2)$

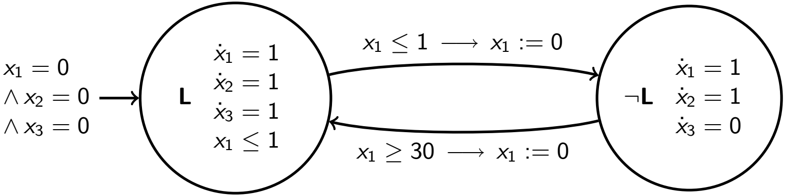
Semantics overview



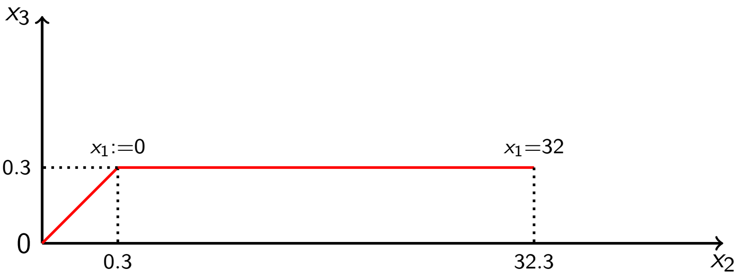
Semantics overview

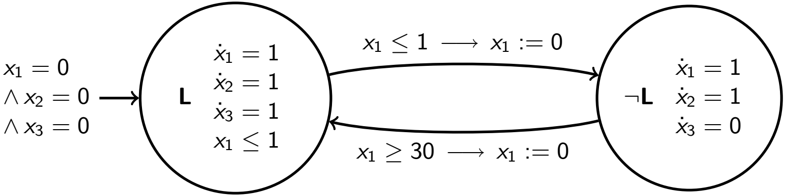


Semantics overview

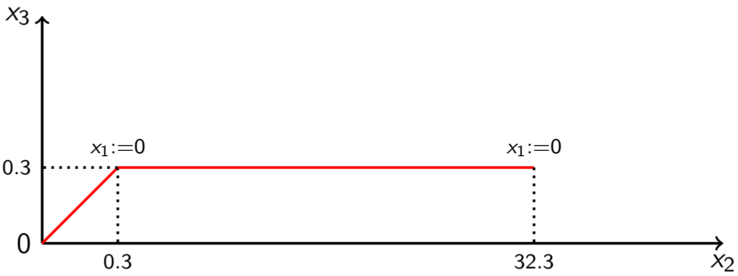


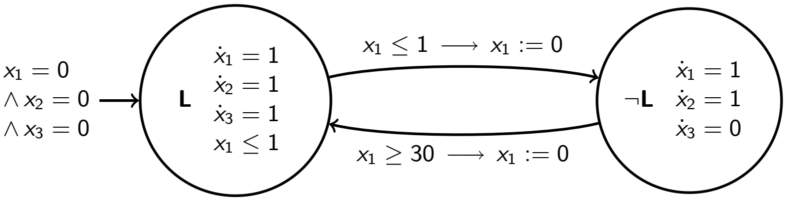
Semantics overview



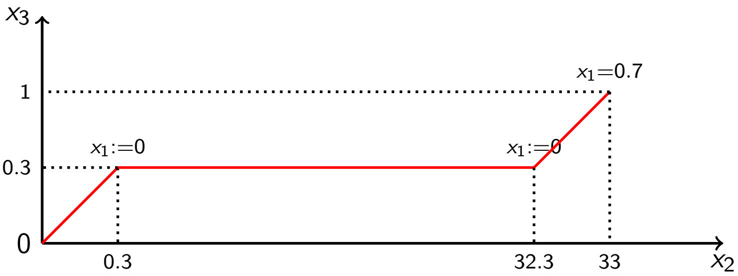


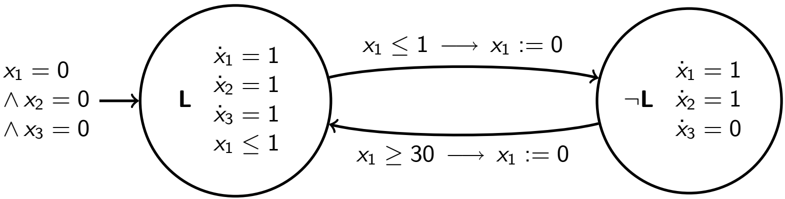
Semantics overview



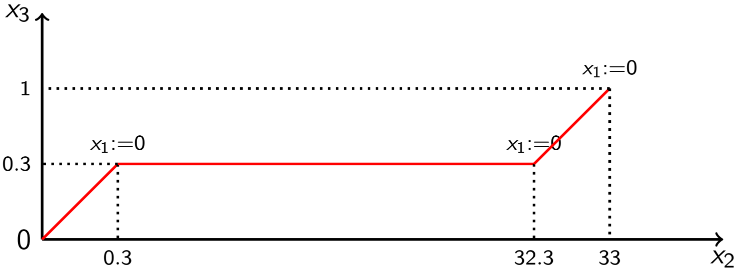


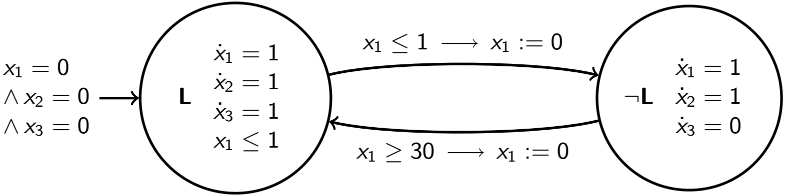
Semantics overview



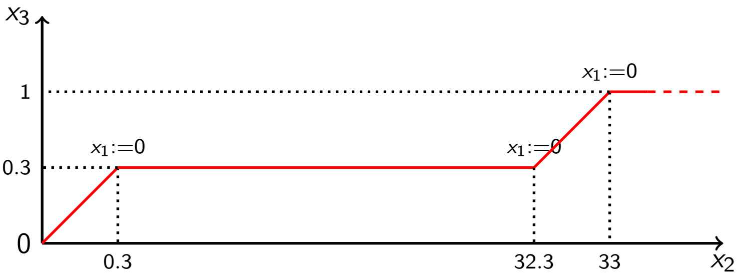


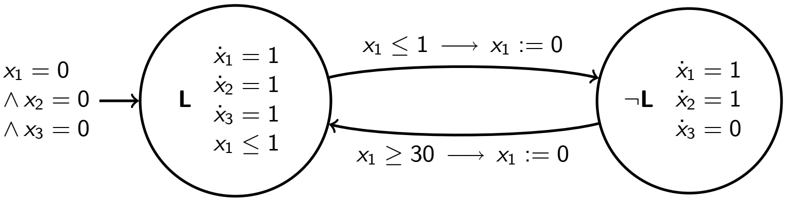
Semantics overview



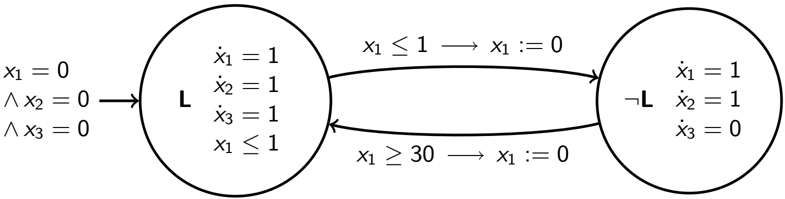


Semantics overview

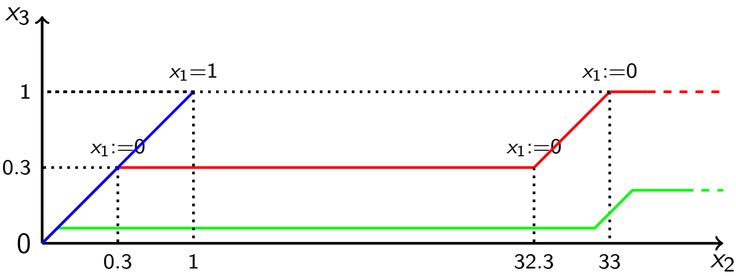


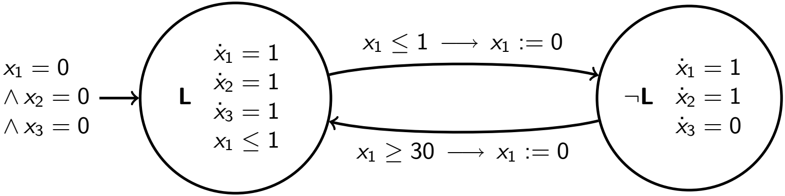


Semantics overview

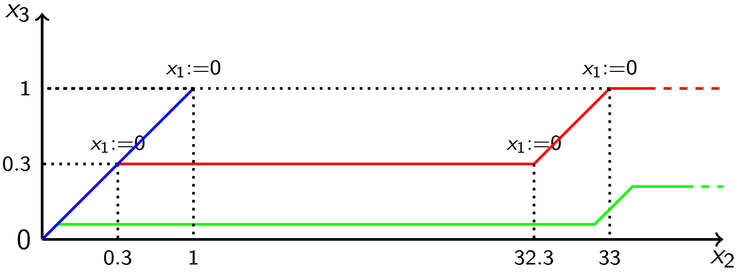


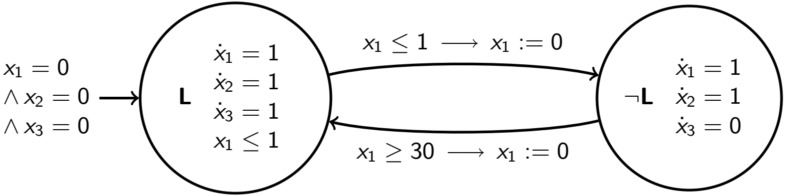
Semantics overview



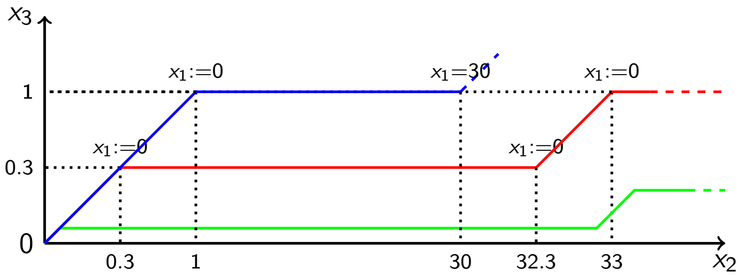


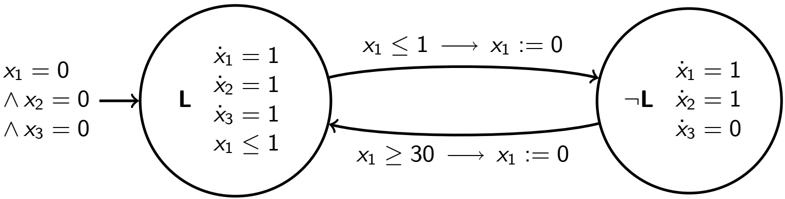
Semantics overview



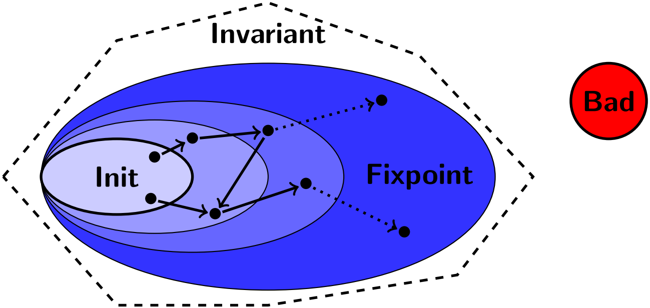


Semantics overview





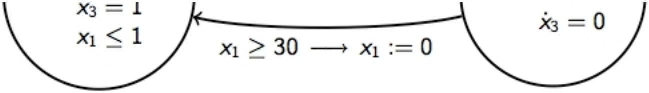
Checking safety properties



- Algorithm:
  - Compute the **fixpoint** of $\rightarrow$ from *Init*
  - Check if *Fixpoint* $\subseteq$ *Invariant* (or *Fixpoint* $\cap$ *Bad* $= \emptyset$)
- Problem: **infinite** and **uncountable** state-space

Plan

Symbolic semantics: transition



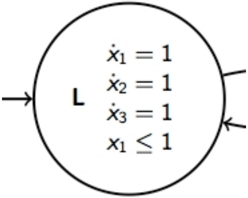
$$(x_1 \geq 30) \wedge (x_1' = 0 \ \wedge \ x_2' = x_2 \ \wedge \ x_3' = x_3) \wedge (x_1' \leq 1)$$

- For a transition: $P_e.x \leq q_e \rightarrow x' = A_e.x + b_e \wedge P_{l'}.x' \leq q_{l'}$

$$\underbrace{\begin{pmatrix} P_e & 0 \\ -A_e & I_n \\ A_e & -I_n \\ 0 & P_{l'} \end{pmatrix}}_{P} \cdot \begin{pmatrix} x \\ x' \end{pmatrix} \leq \underbrace{\begin{pmatrix} q_e \\ b_e \\ -b_e \\ q_{l'} \end{pmatrix}}_{q}$$

- Sound and complete: $P.(v \ v') \leq q$ iff $(l, v) \rightarrow (l', v')$

Symbolic semantics: time elapsing



$\exists t \in \mathbb{R}_{\geq 0}$ s.t.

$x_1' = x_1 + t \wedge x_2' = x_2 + t \wedge x_3' = x_3 + t \wedge x_1' \leq 1$

- For a state: $\exists t \in \mathbb{R}_{\geq 0},\ P_l'.(x' - x) \leq q_l'.t \ \wedge\ P_l.x \leq q_l$

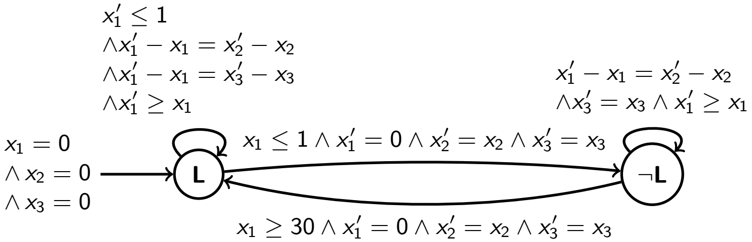
$$\begin{pmatrix} -P_l' & P_l' & -q_l' \\ 0 & P_l & 0 \\ 0 & 0 & -1 \end{pmatrix} . \begin{pmatrix} x \\ x' \\ t \end{pmatrix} \leq \begin{pmatrix} 0 \\ q_l \\ 0 \end{pmatrix}$$

  Projection onto $(x\ x')$ yields a polyhedron $P.(x\ x') \leq q$

- Sound and complete: $P.(v\ v') \leq q$ iff $(l, v) \rightarrow (l, v')$

LHR Systems

- System of **real-valued counters** with **discrete** evolutions defined by LHR $\theta(x, x')$

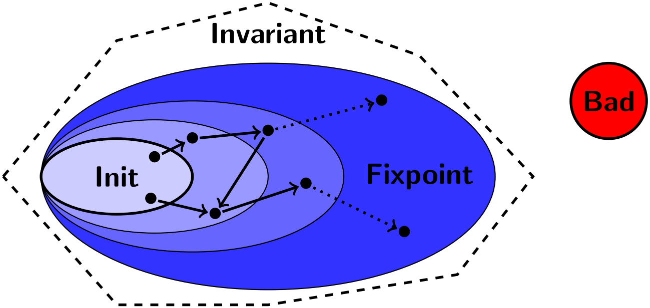


- Goal: check **safety** properties of **LHR systems**

Plan

Plan

Effective symbolic state-space computation

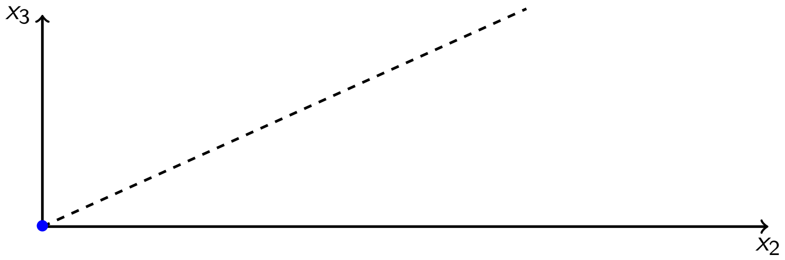


- **Symbolic state:** polyhedron $P_s.x \leq q_s$
- **Symbolic transition:** polyhedron $P_t.(x\ x') \leq q_t$
- **Intersection** + **projection** onto x' + **renaming** yields new state $P'_s.x \leq q'_s$.

State-space computation for the LGB

Successive symbolic states in L:

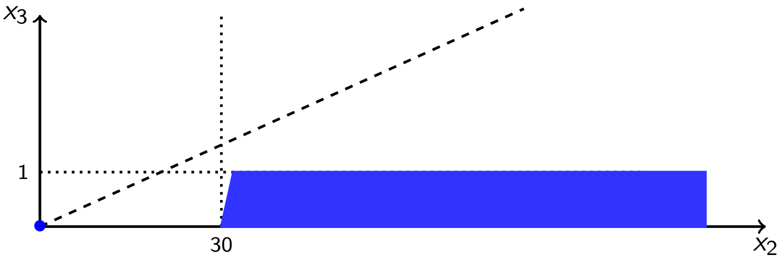
1. $x_1 = 0 \wedge x_2 = 0 \wedge x_3 = 0$



State-space computation for the LGB

Successive symbolic states in L:

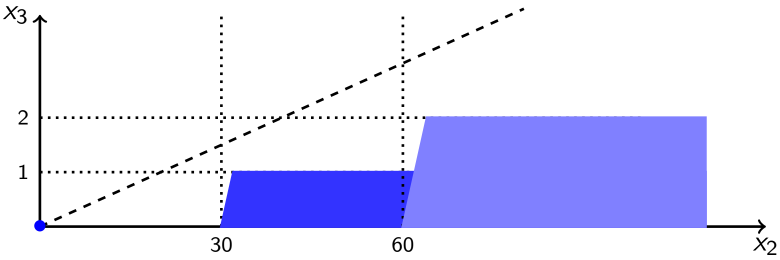
1. $x_1 = 0 \wedge x_2 = 0 \wedge x_3 = 0$
2. $x_1 = 0 \wedge x_3 \leq 1 \wedge x_2 - x_3 \geq 30 \wedge x_3 \geq 0$



State-space computation for the LGB

Successive symbolic states in L:

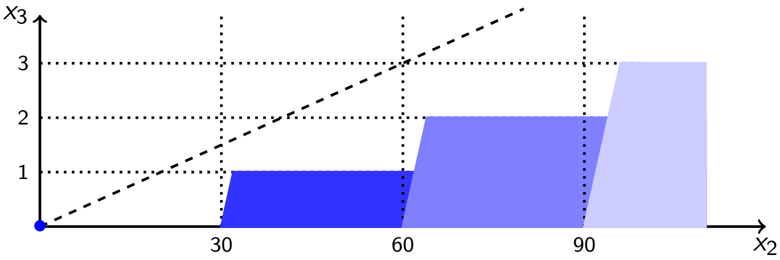
1. $x_1 = 0 \wedge x_2 = 0 \wedge x_3 = 0$
2. $x_1 = 0 \wedge x_3 \leq 1 \wedge x_2 - x_3 \geq 30 \wedge x_3 \geq 0$
3. $x_1 = 0 \wedge x_3 \leq 2 \wedge x_2 - x_3 \geq 60 \wedge x_3 \geq 0$



State-space computation for the LGB

Successive symbolic states in L:

1. $x_1 = 0 \wedge x_2 = 0 \wedge x_3 = 0$
2. $x_1 = 0 \wedge x_3 \leq 1 \wedge x_2 - x_3 \geq 30 \wedge x_3 \geq 0$
3. $x_1 = 0 \wedge x_3 \leq 2 \wedge x_2 - x_3 \geq 60 \wedge x_3 \geq 0$
4. $x_1 = 0 \wedge x_3 \leq 3 \wedge x_2 - x_3 \geq 90 \wedge x_3 \geq 0$

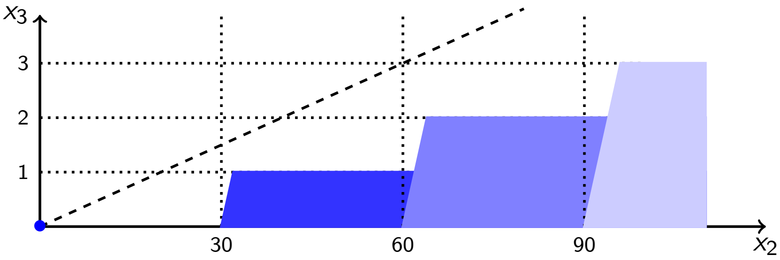


State-space computation for the LGB

Successive symbolic states in L:

1. $x_1 = 0 \wedge x_2 = 0 \wedge x_3 = 0$
2. $x_1 = 0 \wedge x_3 \leq 1 \wedge x_2 - x_3 \geq 30 \wedge x_3 \geq 0$
3. $x_1 = 0 \wedge x_3 \leq 2 \wedge x_2 - x_3 \geq 60 \wedge x_3 \geq 0$
4. $x_1 = 0 \wedge x_3 \leq 3 \wedge x_2 - x_3 \geq 90 \wedge x_3 \geq 0$

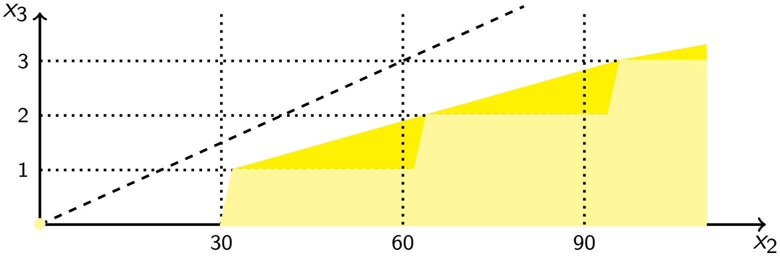
... **The algorithm does not reach a fixpoint!**



Termination ensured by widening [CC77,CH78]

Capture an (over-approximated) **invariant** of the loop

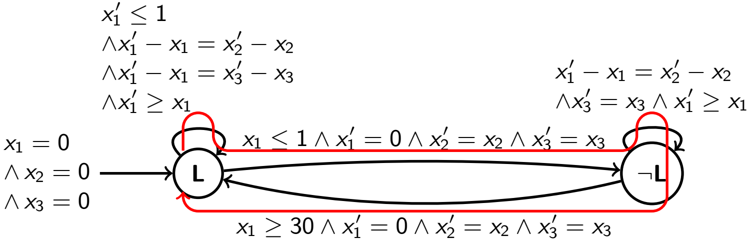
"The intuition is clear: when a constraint is translated or rotated, it can do do infinitely many times, so it is removed" [CH78,HPR94,AHH94]



Conclusion: $(x_2 \geq 60 \Rightarrow 20x_3 \leq x_2)$ **holds !**

Plan

Exact loop invariant for the LGB



- Symbolic state in L after k iterations of the cycle:

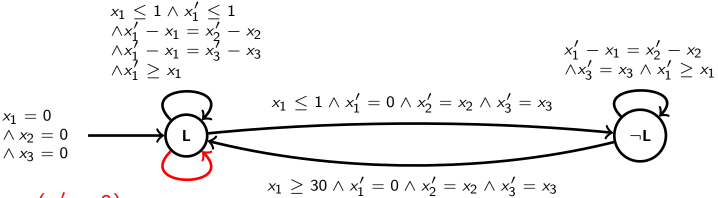
$$(x_1 = 0) \wedge (0 \leq x_3) \wedge (x_3 \leq k + 1) \wedge (x_2 - x_3 \geq 30(k + 1))$$

- State-space computation does not terminate because it has to compute the set:

$$\{(x_1 = 0) \wedge (0 \leq x_3) \wedge (x_3 \leq k+1) \wedge (x_2 - x_3 \geq 30(k+1)) \mid k \in \mathbb{N}\}$$

Acceleration

- **Idea:** Add a **meta-transition** that captures the loop invariant

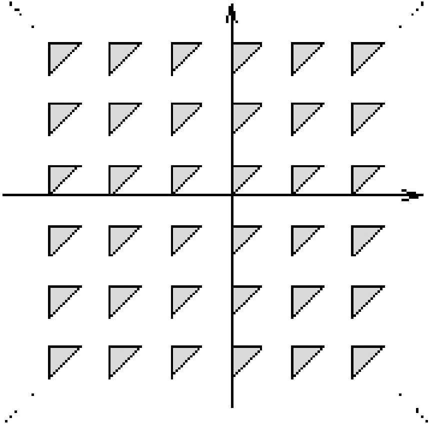


$$\exists k \in \mathbb{N}. \begin{array}{l} (x_1' = 0) \\ \wedge (x_1 + x_3' - x_3 \leq (k+1)) \\ \wedge (x_3 \leq x_3') \\ \wedge ((x_2' - x_2) - (x_3' - x_3) \geq 30(k+1)) \end{array}$$

- **Problems:**
  - the loop invariant is **not a polyhedron** $\rightarrow$ $\mathrm{FO}(\mathbb{R}, \mathrm{Z}, +, \leq)$
  - how do we **compute** the loop invariant? $\rightarrow$ **periodic** on k

Quick overview of FO($\mathbb{R}, Z, +, \leq$)

"Every set of discrete values defined in FO($\mathbb{R}, Z, +, \leq$) is **ultimately periodic**" [Wei99]

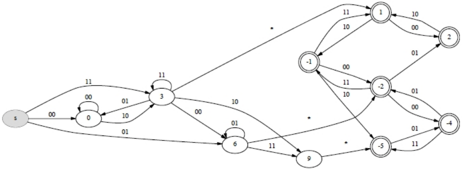


Real Vector Automata [BBR97,BRW98,BJW01]

- Real numbers encoded as **infinite words** over $\{0, 1, \bullet\}$:

$$3.5 \text{ encoded as } 0^{+}11 \bullet 1(0)^{\omega} \text{ or } 0^{+}11 \bullet 0(1)^{\omega}$$

- Every set definable in $\mathrm{FO}(\mathbb{R}, \mathrm{Z}, +, \leq)$ is represented by a **Weak Deterministic Büchi Automaton**



$$3x - 6y = 4$$

- Well-known **efficient** algorithms for $\cap$, $\cup$, $\exists$ and **canonical** representation

Plan

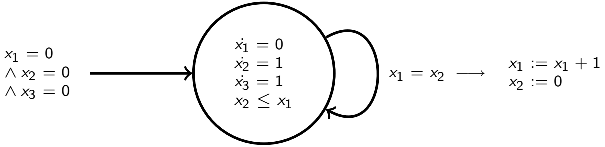
Plan

Iterability of LHR

Definition

A LHR $\theta(x, x')$ is **iterable** if its reflexive and transitive closure is definable in $\mathrm{FO}(\mathbb{R}, \mathrm{Z}, +, \leq)$

- Some LHR are **not iterable**: $x_3 = \sum_{j=1}^{k} j = \frac{k(k-1)}{2}$ at iteration k

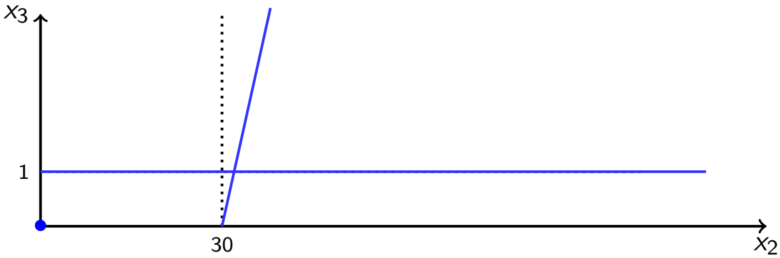


$$(x_1 - x_2 = x_3' - x_3) \wedge (x_2 \leq x_1) \wedge (x_1' = x_1 + 1) \wedge (x_2' = 0)$$

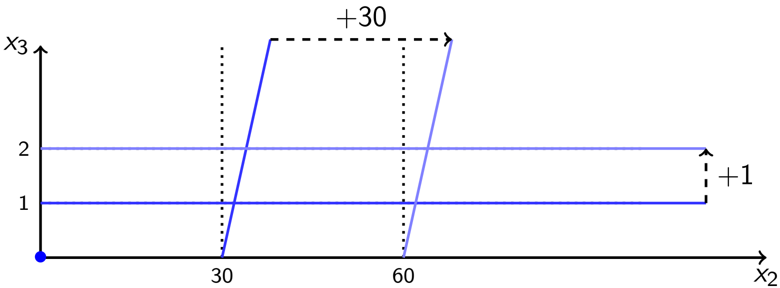
- **Deciding** if a given LHR is **iterable** is an open problem

Trajectory of extremal rays

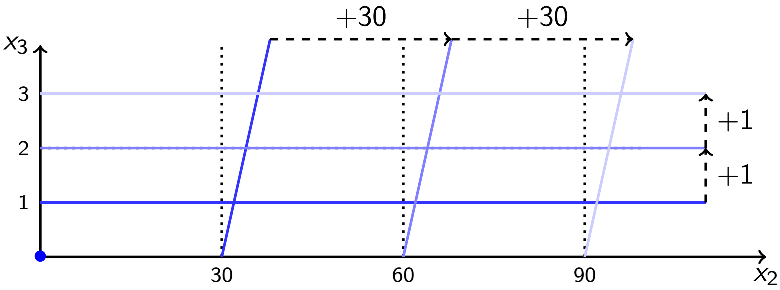
Trajectory of extremal rays



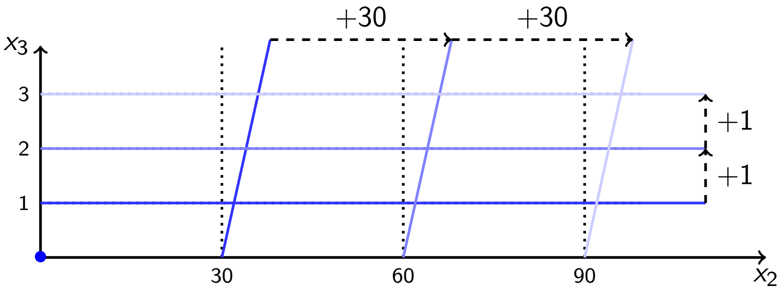
Trajectory of extremal rays



Trajectory of extremal rays



Trajectory of extremal rays



- **Periodic translations** of the faces of $P.(x\ x') \leq q$

$$
\begin{aligned}
&(x_1' = 0) \\
&\wedge(x_1 + x_3' - x_3 \leq (k+1)) \\
&\wedge(x_3 \leq x_3') \\
&\wedge((x_2' - x_2) - (x_3' - x_3) \geq 30(k+1))
\end{aligned}
$$

Periodic LHR

Definition

A LHR $\theta(x, x')$ is **periodic** if it has the following form:

$$\begin{pmatrix} P_0 & 0 \\ -P_1 & P_1 \\ 0 & P_2 \end{pmatrix} \cdot \begin{pmatrix} x \\ x' \end{pmatrix} \leq \begin{pmatrix} q_0 \\ q_1 \\ q_2 \end{pmatrix}$$

Intuition: $(-P_1 \ P_1).(x \ x') \leq q_1$ is a **translation**

$$\begin{pmatrix} -P_1 & P_1 & 0 \\ 0 & -P_1 & P_1 \end{pmatrix} \cdot \begin{pmatrix} x \\ x' \\ x'' \end{pmatrix} \leq \begin{pmatrix} q_1 \\ q_1 \end{pmatrix} \quad \rightsquigarrow \quad (-P_1 \ P_1).(x \ x'') \leq \mathbf{2} q_1$$

Periodic acceleration of LHR

Theorem

The reflexive and transitive closure θ^* of any **periodic LHR** θ is definable in $\mathrm{FO}(\mathbb{R}, \mathrm{Z}, +, \leq)$

Proof.

- For any $k \geq 2$ and V in $\mathrm{FO}(\mathbb{R}, \mathrm{Z}, +, \leq)$

$$\theta^k(V) = \theta(\theta_1^{k-2}(\theta(V) \cap C) \cap C)$$

  - θ_1^k defined by $P_1.(x\ x') \leq \mathbf{k}q_1$
  - $C = \{v \mid P_0.v \leq q_0 \text{ and } P_2.v \leq q_2\}$

- θ^* obtained by **integer quantification** over k

$\square$

Back to "Leaking Gas Burner"

The cycle in the "Leaking Gas Burner" is **not a periodic LHR**:

$$(x_1' = 0) \wedge (x_3' - x_3 \leq 1 - \mathbf{x}_1) \wedge (x_3' - x_3 \leq x_2' - x_2 - 30) \wedge (x_3 \leq x_3')$$

$$\begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ \mathbf{1} & 0 & -1 & 0 & 0 & 1 \\ 0 & 1 & -1 & 0 & -1 & 1 \\ 0 & 0 & 1 & 0 & 0 & -1 \end{pmatrix} . \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_1' \\ x_2' \\ x_3' \end{pmatrix} \leq \begin{pmatrix} 0 \\ 0 \\ 1 \\ -30 \\ 0 \end{pmatrix}$$

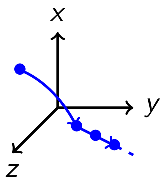
The **initial value of** x_1 constrains the time spent in the leaking location

Plan

Focusing on the iterative subspace

- Since $x_1' = 0$ each iteration, except the first one, takes place in the **plane** (x_2, x_3)

$$(x_1' = 0) \wedge (x_3' - x_3 \leq 1 - \mathbf{x}_1) \wedge (x_3' - x_3 \leq x_2' - x_2 - 30) \wedge (x_3 \leq x_3')$$



- By restriction to the plane (x_2, x_3), one gets a **periodic LHR**:

$$\begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 \\ \mathbf{0} & 0 & -1 & 0 & 0 & 1 \\ 0 & 1 & -1 & 0 & -1 & 1 \\ 0 & 0 & 1 & 0 & 0 & -1 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_1' \\ x_2' \\ x_3' \end{pmatrix} \leq \begin{pmatrix} 0 \\ 0 \\ 1 \\ -30 \\ 0 \end{pmatrix}$$

Subspace reduction

Theorem

Any LHR $\theta(x, x')$ with dimension n such that $dim(\theta(\mathbb{R}^n)) = p < n$ is **reducible** to a LHR θ' with dimension p

Proof.

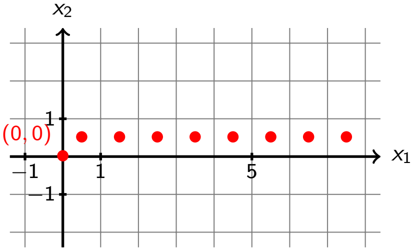
- Compute a **change of variables** $x = U.y + u_0$
  - u_0 is any point in $\theta(\mathbb{R}^n)$
  - U is a basis for $\theta(\mathbb{R}^n) - u_0$
- Obtain $\theta'(y, y')$ by adding $x = U.y + u_0$ and $x' = U.y' + u_0$ to $\theta(x, x')$, and by projecting out x and x'
- $\forall k > 0$ and $\forall V$ in $\mathrm{FO}(\mathbb{R}, \mathrm{Z}, +, \leq)$, we have $\theta^k(V) = U.(\theta')^{k-1}(V') + u_0$ where $V' \subseteq \mathbb{R}^m$ is the solution of $\theta(V) = U.V' + u_0$

$\square$

A similar result holds when $dim(\theta^{-1}(\mathbb{R}^n)) < n$

Ultimately periodic LHR

- This LHR is **not periodic** and subspace reduction does not apply $\theta: \ (x_1' + x_2' = x_1 + x_2 + 1) \ \wedge \ (x_1' - x_2' = x_1 + x_2)$
- However, it is **ultimately periodic**

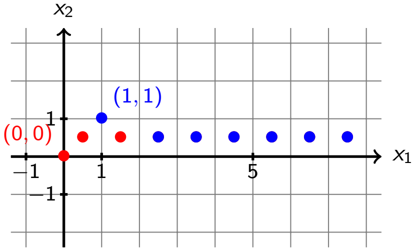


- Indeed: $\theta^2: \ (x_1' = x_1 + x_2 + 3/2) \wedge (x_2' = 1/2)$

Ultimately periodic LHR

- This LHR is **not periodic** and subspace reduction does not apply $\theta : \ (x_1' + x_2' = x_1 + x_2 + 1) \ \wedge \ (x_1' - x_2' = x_1 + x_2)$

- However, it is **ultimately periodic**

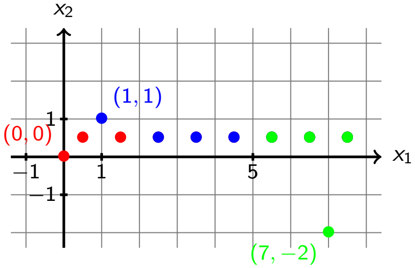


- Indeed: $\theta^2 : \ (x_1' = x_1 + x_2 + 3/2) \wedge (x_2' = 1/2)$

Ultimately periodic LHR

- This LHR is **not periodic** and subspace reduction does not apply θ : $(x_1' + x_2' = x_1 + x_2 + 1) \ \wedge \ (x_1' - x_2' = x_1 + x_2)$

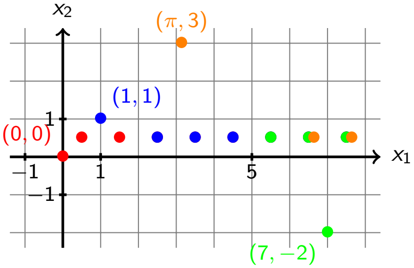
- However, it is **ultimately periodic**



- Indeed: θ^2 : $(x_1' = x_1 + x_2 + 3/2) \wedge (x_2' = 1/2)$

Ultimately periodic LHR

- This LHR is **not periodic** and subspace reduction does not apply $\theta : \ (x_1' + x_2' = x_1 + x_2 + 1) \ \wedge \ (x_1' - x_2' = x_1 + x_2)$
- However, it is **ultimately periodic**



- Indeed: $\theta^2 : \ (x_1' = x_1 + x_2 + 3/2) \wedge (x_2' = 1/2)$

Dimensions preserved by iteration

- Iterations of θ **only preserves the value of** $x_1 + x_2$, not the individual values of x_1 and x_2

$$\theta : \; (x_1' + x_2' = x_1 + x_2 + 1) \wedge (x_1 + x_2 \leq x_1' - x_2')$$

- Using the change of variables $y = x_1 + x_2$, we obtain a **periodic LHR**:

$$\begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix} \cdot \begin{pmatrix} y \\ y' \end{pmatrix} \leq \begin{pmatrix} 1 \\ -1 \end{pmatrix} \qquad\qquad y' = y + 1$$

- This change of variables is detected and obtained from the **rank** of P

$$\begin{pmatrix} -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & 1 & -1 & 1 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_1' \\ x_2' \end{pmatrix} \leq \begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix}$$

Rank reduction

Theorem

Any LHR θ with dimension n defined by $(P\ P').(x\ x') \leq q$ and s.t. $rank(P) = p < n$ is **reducible** to a LHR θ' with dimension p

Proof.

- Compute a **change of variables** $y = U.x$ from the linearly independent rows in P
- Obtain $\theta'(y, y')$ by adding $y = U.x$ and $y' = U.x'$ to $\theta(x, x')$, and then by projecting out x and x'
- $\forall k > 0$ and $\forall V$ in FO($\mathbb{R}$, Z, $+$, $\leq$), we have $\theta^k(V) = \theta''((\theta')^{k-1}(U.S))$ where $\theta'' \subseteq \mathbb{R}^p \times \mathbb{R}^n$ is defined by $P'.x' + P''.y \leq q$ where $P = P''.U$.

$\square$

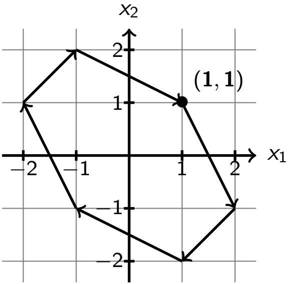
A similar result holds when $rank(P_{x'}) < n$

Rotations and permutations

- The following LHR is **not periodic** and **irreducible**:

$$\begin{cases} x_1' = x_1 + x_2 \\ x_2' = -x_1 \end{cases} \qquad \begin{pmatrix} -1 & -1 & 1 & 0 \\ 1 & 1 & -1 & 0 \\ 1 & 0 & 0 & 1 \\ -1 & 0 & 0 & -1 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_1' \\ x_2' \end{pmatrix} \leq \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \end{pmatrix}$$

- However, its trajectory is periodic and it has **period 6**



Finding the permutation of variables

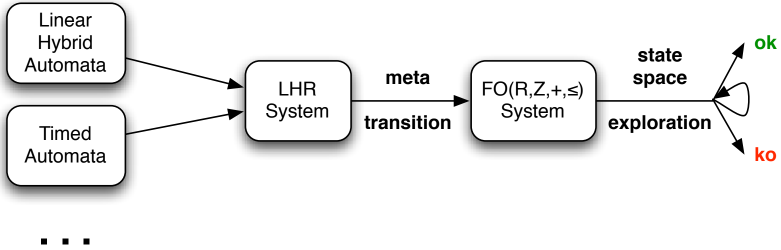
$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \rightarrow \begin{pmatrix} x_1 + x_2 \\ -x_1 \end{pmatrix} \rightarrow \begin{pmatrix} x_2 \\ -x_1 - x_2 \end{pmatrix} \rightarrow \begin{pmatrix} -x_1 \\ -x_2 \end{pmatrix} \rightarrow \begin{pmatrix} -x_1 - x_2 \\ x_1 \end{pmatrix} \rightarrow \begin{pmatrix} -x_2 \\ x_1 + x_2 \end{pmatrix} \rightarrow \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

- The constraints in $\theta^k(x, x')$ are **linear combinations** of the constraints in $\theta(x, x')$
- $\theta^k(x, x')$ is periodic, i.e. $P_k = -P'_k$, if there exists a **linear combination** $A \in \mathbb{Z}^{n \times n}$ **of the constraints in** $\theta(x, x')$ s.t. both:
  - $A.P = -P'$ (where $\theta(x, x') \equiv (P\ P').(x\ x') \leq q$)
  - and $A^k = I$ for some integer $k > 0$

$$\underbrace{\begin{pmatrix} 0 & -1 \\ 1 & 1 \end{pmatrix}}_{A} \cdot \underbrace{\begin{pmatrix} -1 & -1 \\ 1 & 0 \end{pmatrix}}_{P} = \underbrace{\begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix}}_{-P'} \quad \text{and} \quad A^6 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$$

Plan

Summary of the method



- Meta-transitions capture **infinite periodic loop iterations**
- **No guarantee of termination** of the state-space computation

Completeness results

- Reachability is **undecidable** for Linear Hybrid Automata
- **Complete** method for **Timed Automata**:
  - Every loop is **reducible** to a **periodic LHR** [CJ98,BIL06,BH06]
  - Timed automata are **flat** [CJ99]
- State-space computation does not terminate in presence of **nested loops**

Hybrid Acceleration Toolkit (HAT)

- **Prototype tool** built on top of **LASH** and Polylib
- Currently implements:
  - a LHR manipulation engine
  - **Periodic acceleration**
  - **Subspace, Rank and Static reductions**
- Takes **both the model and the meta-transitions** as an input, and computes its state-space
- **Automatically** handles **reduction and acceleration** of meta-transitions

Analysing the "Leaking Gas Burner" with HAT

```
***********************************
HAT - Hybrid Acceleration Toolkit v0.1
***********************************

State space computation.....
     mem usage: 351488, max mem usage: 1757134, time: 2.000000
          loc[0] size: 39
          loc[1] size: 0
     mem usage: 378832, max mem usage: 1951048, time: 2.000000
          loc[0] size: 442
          loc[1] size: 39
     mem usage: 408940, max mem usage: 12214286, time: 6.000000
          loc[0] size: 559
          loc[1] size: 469
     mem usage: 426264, max mem usage: 12214286, time: 11.000000
          loc[0] size: 559
          loc[1] size: 834
     mem usage: 412380, max mem usage: 13711777, time: 14.000000
          loc[0] size: 559
          loc[1] size: 684
     Fixpoint reached.....17.000000
Property checking.....
     Checking: invariant z>=60 -> 20y<=z
          loc[0]: ok
          loc[1]: ok
Finished.....
     mem usage: 0, max mem usage: 14518513, time: 17.000000
```

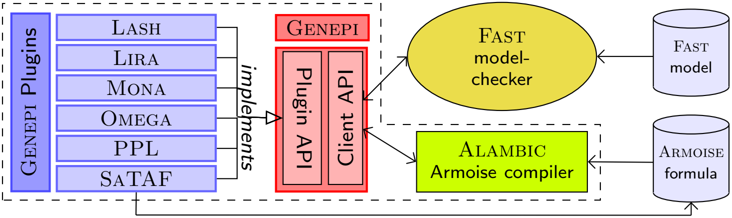
Perspectives

- Rotation reduction for **non square LHR**
- **Completeness** results for other **decidable subclasses of LHA** (Rectangular Initialized Hybrid Automata)
- Loop invariants for **nested loops**, i.e. acceleration of **parametrized** LHR
  - may lead to **non linear** systems
- Model-checking **temporal logics** instead of reachability properties only
  - loop acceleration integrates well in SCC or nested-DFS algorithms (LTL)
- **Heuristics** for the choice of **meta-transitions**

Related works

- Programs with **FIFO queues** [Boigelot&Godefroid'96,'97, Abdulla&Annichini&Bouajjani'99]
- Programs with **integer counters** [Boigelot&Wolper'94,'95,'98,'00, Finkel&Leroux'00,'02, Bardin&Finkel&Leroux'04]
- Programs with **real counters** [Boigelot&Bronne&Rassart'97, Comon&Jurski'98,'99, Boigelot&Herbreteau&Jodogne'03, Boigelot&Herbreteau'06, Bozga&Iosif&Lakhnech'06, Bozga&Girlea&Iosif'09]
- **Combination** of datatypes [Annichini&Asarin&Bouajjani'00, Annachini&Bouajjani&Sighireanu'01, Bardin&Finkel'04]

TaPaS tool [Leroux&Point'09]



- GENEPI : generic API to the Presburger arithmetic
- ARMOISE/ALAMBIC : programmation with Presburger arithmetic
- FAST : symbolic model-checker for systems with integer counters

http://altarica.labri.fr/forge/projects/altarica/wiki/TaPAS