Verification of Flat FIFO Systems

Alain Finkel (LSV, ENS Paris-Saclay, France, UMI ReLaX)
M. Praveen (Chennaï Mathematical Institute, Chennaï, India, UMI ReLaX)

30th CONCUR, Amsterdam, 27th August 2019
Motivation

Verification of infinite-state FIFO systems

- Model defined in 1970 for communication protocols.
- Difficult to verify since reachability is undecidable.
- Used for choreography, contract, interfaces, web services, ...
- Reachability is decidable for interesting subclasses.
- Interesting papers about synchronizability (more or less correct).
A FIFO system from (LY 2019) with 3 processes $P, Q, R$ and 4 channels: $pq, pr, qp, rq$

(a) Process $P$
A FIFO system from (LY 2019) with 3 processes \( P, Q, R \) and 4 channels: \( pq, pr, qp, rq \)

(a) Process \( P \)

(b) Process \( Q \)
A FIFO system from (LY 2019) with 3 processes $P$, $Q$, $R$ and 4 channels: $pq, pr, qp, rq$

(a) Process $P$

(b) Process $Q$

(c) Process $R$
process $P$

- $pq!a_1$
- $pr!c$
- $pq!a_2$
- $pr!c$
- $qp?b$
- $pq!y$
- $qp?x$

process $Q$

- $pq?a_1$
- $rq?d$
- $pq?a_2$
- $rq?d$
- $qp?b$
- $pq?y$
- $qp!x$

process $R$

- $pr?c$
- $rq!d$
process $P$

$pq!a_1 \quad pr!c$

$pq!a_2$

$pr!c$

$qp?b$

$\leftrightarrow pq!y$

$qp?x$

$c$

process $Q$

$pq?a_1 \quad rq?d$

$pq?a_2$

$rq?d$

$pq?a_1$

$qp!y$

$qp!b$

$qp?x$

process $R$

$pr?c$

$rq!d$

$pq!b$

$qp?a_2$

$qp?x$

$c$

$d$
What is mainly known about FIFO systems?

- Reachability and boundedness are **undecidable** for
  - one FIFO automata
  - two communicating machines (2-CFSM)
What is mainly known about FIFO systems?

- Reachability and boundedness are undecidable for
  - one FIFO automata
  - two communicating machines (2-CFSM)
- The reachability set is recognizable for
  - synchronous systems of CFSM
  - $k$-bounded systems ($k \geq 0$)
  - half-duplex systems of 2-CFSM (not for 3-CFSM).
  - lossy/insertion systems and variants with time, data and priority (but not perfect FIFO) but boundedness is still undecidable.
What is mainly known about FIFO systems?

- Reachability and boundedness are undecidable for
  - one FIFO automata
  - two communicating machines (2-CFSM)
- The reachability set is recognizable for
  - synchronous systems of CFSM
  - \( k \)-bounded systems \((k \geq 0)\)
  - half-duplex systems of 2-CFSM (not for 3-CFSM).
  - lossy/insertion systems and variants with time, data and priority (but not perfect FIFO) but boundedness is still undecidable.
- Reachability is decidable for
  - recognizable systems
  - 1-existential bounded systems
  - flat systems.
What precisely about Flat FIFO systems (FFS)?

**Known results**

- The reachability set can be effectively represented by $(A, \phi)$ where $A$ is a flat automaton, $\phi$ Presburger formula (BH’99).
- By analysing the proof, reachability is in 2-EXPTIME.
- Control-state reachability is NP-complete (EGM’12).
What precisely about Flat FIFO systems (FFS)?

**Known results**

- The reachability set can be effectively represented by $(A, \phi)$ where $A$ is a flat automaton, $\phi$ Presburger formula (BH’99).
- By analysing the proof, reachability is in 2-EXPTIME.
- Control-state reachability is NP-complete (EGM’12).

**Open complexity and decidability problems**

- Reachability: decidable but exact complexity unknown
- Repeated reachability?
- (letter)-Boundedness?
- Termination?
- LTL, CTL*, equivalences?
Most reachability problems are NP-complete

- Reachability
- Repeated reachability
- (letter)-channel boundedness
- Termination
Our contributions

- **Most reachability problems are NP-complete**
  - Reachability
  - Repeated reachability
  - (letter)-channel boundedness
  - Termination
- **Flat FIFO systems are flat counters systems**
  - FFS are bisimilar to FCS
  - The reachability set is semilinear (also in BH'99)
  - FFS are trace-flattable
  - LTL and $CTL^*$ are decidable.
Outline

1. Introduction and motivation
2. Words and FIFO loops
3. Complexity for Flat FIFO Systems
   - \( \mathbf{NP} \) Upper Bound
   - \( \mathbf{NP} \) Lower Bound
   - \( \mathbf{NP} \) -complete results
4. Construction of an Equivalent Counter System
   - The synchronized counter system
   - The synchronized counter system is trace-flattable
   - LTL and \( \mathbf{CTL}^* \) are decidable
5. Conclusion and perspectives
Two useful lemmas

**Lemma**

Let \( x, y \in \Sigma^+ \) and \( w \in \Sigma^* \).

The equation \( x^\omega = wy^\omega \) holds iff \( \exists z \neq \epsilon, z \) primitive and \( \exists x', x'' \) such that \( w \in x^*x' \) and \( x = x'x'' \) and \( x''x' \in z^* \) and \( y \in z^* \).

**Proof.**

By using Levi’s Lemma.
Two useful lemmas

**Lemma**

Let $x, y \in \Sigma^+$ and $w \in \Sigma^*$. The equation $x^\omega = wy^\omega$ holds iff $\exists z \neq \epsilon$, $z$ primitive and $\exists x', x''$ such that $w \in x^*x'$ and $x = x'x''$ and $x''x' \in z^*$ and $y \in z^*$.

**Proof.**

By using Levi’s Lemma.

**Lemma**

An elementary loop labeled by $\sigma$ is infinitely iterable from $(q, w)$ iff for every channel $c$, $x_c^\sigma = \epsilon$ or ($\sigma$ is fireable at least once from $(q, w)$ and $(x_c^\sigma)^\omega = w(c) \cdot (y_c^\sigma)^\omega$ and $|x_c^\sigma| \leq |y_c^\sigma|$) where $x_c^\sigma$ is the word consumed by $\sigma$ from channel $c$. 

9/31
Path Schemas

(a) Flat FIFO system

(b) Path schema denoted by $p_0(l_1)^* p_1(l_2)^* p_2$

Figure: Example flat FIFO system and path schema
Reachability to Control State Reachability

Theorem (Theorem 3, Theorem 7 in EGM’12)

Let $S = p_0(\ell_1)^* p_1 \cdots (\ell_k)^* p_k$ be a FIFO path schema. We can compute in polynomial time an existential Presburger formula $\phi(x_1, \ldots, x_k)$ such that: there is a run $r = p_0(\ell_1)^{n_1} p_1 \cdots (\ell_k)^{n_k} p_k$ of $S$ iff $\phi(n_1, \ldots, n_k)$ is true. Hence control-state reachability is decidable.

Corollary

Reachability is in $\text{NP}$. $(q, w(1), w(2), \ldots, w(p))$ is reachable iff $q_{\text{stop}}$ is reachable.
Proposition

The repeated control state reachability problem is in $\mathbb{NP}$.

Proof.

Let $q$ be in an elementary loop labeled with $\sigma$ in system $S$(else...).
$q$ is infinitely repeated iff $\forall c \ [x_c^\sigma = \epsilon]$ or $[\exists w \ (q, w) \xrightarrow{\sigma} \text{ and } (x_c^\sigma)^\omega = w(c) \cdot (y_c^\sigma)^\omega \text{ and } |x_c^\sigma| \leq |y_c^\sigma|]$ (from Lemma 2)

1. Verify that for every channel $c$, $|x_c^\sigma| \leq |y_c^\sigma|$

2. Verify $\exists (q, w)$ s.t. $(q, w) \xrightarrow{\sigma}$ and $\forall c \ s.t. \ x_c^\sigma \neq \epsilon$, $(x_c^\sigma)^\omega = w(c) \cdot (y_c^\sigma)^\omega$.

3. For verifying $(x_c^\sigma)^\omega = w(c) \cdot (y_c^\sigma)^\omega$ (Lemma 1), one guesses $x'_c, x''_c, z_c \in M^*$ such that $x_c^\sigma = x'_c x''_c$ and $x''_c x'_c, y_c^\sigma \in z_c^*$.

4. Remark that $|x'_c|, |x''_c| \leq |x_c^\sigma|$ and $|z_c| \leq |y_c^\sigma|$

5. It remains to verify $\exists (q, w)$ s.t. $\forall c, w(c) \in (x_c^\sigma)^* x'_c$ and $(q, w) \xrightarrow{\sigma}$.

6. To do that, we add a channel $c'$ for every channel $c$ in system $S$. 

\checkmark
Recall, we have:

- $q$ is reached repeatedly in $S$ \iff\ $\exists w(c) \text{ s.t. } w(c) \in (x_c^\sigma)^* x_c'$ and $(q, w) \xrightarrow{\sigma} \ Quadrates$.
- $\exists w'(c') \text{ s.t. } w'(c') \in (x_c^\sigma)^* x_c'$ and $(q', w') \xrightarrow{\sigma'} \ Quadrates$.
- $q'$ is reachable in $S'$ and $(q', w') \xrightarrow{\sigma'} \ Quadrates$.
- $q_f$ is reachable in $S'$.

Hence repeated control state reachability reduces to control-state reachability.
Corollary

For flat FIFO systems, the non-termination and unboundedness problems are in $\mathsf{NP}$.

Proof.

- Termination reduces to repeated control-state reachability since a flat system is non-terminating iff there is an infinite run $r$ that visits at least one control state infinitely often.

- The effect of a loop $\ell$ labeled with $\sigma$ is $v_\ell \in \mathbb{Z}^F$ s.t. $\forall c \in F$
  \[ v_\ell(c) = |x_{c}^\sigma| - |y_{c}^\sigma|. \]

- Unboundedness reduces to repeated control-state reachability since a flat FIFO system is unbounded iff there is at least one infinitely iterable loop $\ell$ with $v_\ell \geq 0$ and $v_\ell(c) \geq 1$ for some $c$. 

Proposition

The problem of checking whether a letter $a$ is unbounded in channel $c$ is in \( \mathbf{NP} \).

Proof.

In the proceedings.
Theorem

For flat FIFO systems, reachability, repeated control-state reachability, non-termination, unboundedness, channel-unboundedness and letter-channel-unboundedness are NP-hard.

Proof.

We reduce 3-SAT to reachability. Given a 3-CNF formula $\text{clause}_1 \land \cdots \land \text{clause}_m$ over variables $x_1, \ldots, x_n$, we construct a flat FIFO system with $2n + m$ channels: $\{x_i, \hat{x}_i \mid i \in [1, n]\} \cup \{c_i \mid i \in [1, m]\}$.

- channel $x_i$ is used to keep a guess of the truth assignment to $x_i$.
- channel $\hat{x}_i$ is a “control channel” that ensures that only one guess is made.
- channel $c_i$ is used to verify that $\text{clause}_i$ is satisfied.

The given 3-CNF formula is satisfiable iff the last control state of the cleanup gadget for variable $x_n$ can be reached with all channels being empty.
The gadget for the example clause $c_1 = x_1 \lor \neg x_2 \lor x_3$

(a) Gadget for variable $x_i$

(b) Gadget for clause
$c_1 = x_1 \lor \neg x_2 \lor x_3$

(c) Gadget for cleaning up variable $x_i$
Theorem (Most properties are NP-complete)

For flat FIFO systems, the 7 reachability properties are NP-complete:

1. reachability
2. repeated reachability
3. repeated control-state reachability
4. termination
5. boundedness
6. channel-boundedness
7. letter-channel-boundedness.

Cyclicity can be decided in linear time.
After reachability properties, model checking

- model-checking with atomic formula $\#_c^a \geq k$
- not a consequence of the previous results (BH’99, EGM’12)
- translate a flat FIFO system into a flat counter system
- to use the existing counter systems tools
Counting abstraction system $S_{\text{count}}$:
- **count perfectly** the number of (letter × transition) sent and received
- **loose** the order of letters.
- $(a, t_1)^{++}$ is the incrementation of counter $(a, t_1)$
- $(a, t_3)^{--}$ is the decrementation of counter $(a, t_3)$.

(a) Flat FIFO system

(b) Counting abstraction system $S_{\text{count}}$
Order system $S^c_{\text{order}}$:

- is almost a finite automaton (it don't modify counters but makes zero-tests) that respects the FIFO policy of sent (hence received) letters.
- $(b, t_2)$ is the label of transition from $q_2$ to $q_1$ that don't modify counters.
- its language is the sequences of sent letters: $[(a, t_1).(b, t_2)]^*.(a, t_3)^*$
- don't count so loose the number of letters.
- $(a, t_1) + (b, t_2) = 0$ means that it leaves a loop $\ell$ only if all letters sent by $\ell$ have been consumed.
Synchronized counter system

- $S_{\text{count}}$ is synchronized with $S_{\text{order}}^{c}$ by rendez-vous on transition labels.
- A decrementation $(a, t_1)\downarrow$ in $S_{\text{count}}$ is synchronized with the label $(a, t_1)$ in $S_{\text{order}}^{c}$; this insures that receptions follow the FIFO ordering.
- Incrementations in $S_{\text{count}}$ are not synchronized since sending is free.

\[(a, t_1) + (b, t_2) = 0\]
Proposition

The synchronized counter system $S_{\text{sync}}$ is (weakly) bisimilar to the flat FIFO system.

Proof.

Prove the weak bisimulation by routine induction on the length of the run of $S_{\text{sync}}$ reaching the configuration $(\overline{q}, \nu)$.  
Modify the synchronized system $S_{\text{sync}}$ to obtain a bisimulation.
Proposition

The synchronized counter system $S_{\text{sync}}$ is trace-flattable (hence, for example, the tool FAST will terminate).

Remark

$S_{\text{count}}$ is not flat in general.
Proof.

Suppose a run is visiting states $q_3, q_4$ of $S_{\text{count}}$ and states $q_3, q_4$ of $S_{\text{order}}$. (grey part no longer reachable).

(a) (possibly reachable) non flat $S_{\text{count}}$

(b) (possibly reachable) $S_{\text{order}}^c$
Part of synchronized counter system still reachable

Proof.

The part of synchronized counter system still reachable is flat.

\begin{center}
\begin{tikzpicture}
  \node (q4) at (0,0) {$(q_4, q_3)$};
  \node (a_t3) at (-2,-2) {$(a, t_3)^{++}$};
  \node (q3) at (-1,-4) {$(q_3, q_3)$};
  \node (a_t3_neg) at (2,-2) {$(a, t_3)^{--}$};
  \node (q3_q4) at (1,-4) {$(q_3, q_4)$};
  \draw[->] (q4) -- (a_t3); \draw[->] (a_t3) -- (q3); \draw[->] (q3) -- (a_t3_neg); \draw[->] (a_t3_neg) -- (q3_q4); \draw[->] (q3) -- (q3_q4); \node at (0,-5) {$\tau$};
\end{tikzpicture}
\end{center}
Theorem

*LTL and CTL* are decidable for flat FIFO systems.*

Proof.

Trace-flattening preserves LTL and bisimulation preserves \( CTL^* \).
Open problems

Still open

- Collect case studies.
- Build and experiment a tool that flatten FIFO systems.
- Solve many open complexity problems: LTL, $CTL^*$, equivalences for FFS.

Info

- The paper, with complete proofs, is on HAL.
- https://hal.archives-ouvertes.fr/hal-02267453
Post-doc positions are available at LSV.

- To make theory and/or a tool for counter/FIFO systems.
- Collaborations with many researchers in LSV (ENS Paris-Saclay), LaBRI (Univ. Bordeaux), Canada, India (Chennai, Bombay), Germany,...
Thank you