Approximate Partial Order Reduction

Chuchu Fan, Zhenqi Huang & Sayan Mitra
University of Illinois at Urbana-Champaign
FM, FLOC, OXFORD
JULY 2018
data-driven reachability

Given a system model $A$, initial set $\Theta \subseteq Q$, and unsafe set $U \subseteq Q$, does there exist an execution of $A$ from $\Theta$ that hits $U$?

Check $\text{Reach}(\Theta) \cap U = \emptyset$?

For deterministic systems simulation data + sensitivity analysis have proven to be effective [Fan CAV16, TACAS 15, Huang CAV14]

- From a single execution $\xi_{q_0}$ form initial state $q_0$ and static sensitivity analysis compute $\text{Reach}(B_\delta(q_0))$
- Take union, check safety, and refine with smaller $\delta$ as needed

Nondeterministic challenge: $\text{Reach}(\Theta, \Sigma) \cap U = \emptyset$?

- Many concurrent executions $\xi_{q_0,\tau}$ from $q_0$ following different action sequences $\tau \in \Sigma$
- Combinatorial explosion $\Sigma$ + state space explosion $\Theta$
Partial order reduction (POR):

- If actions commute, then explore only one of the equivalent executions
- Can give exponential savings in computing $\text{Reach}(q_0, \Sigma)$

In our models, actions do not commute exactly, but approximately

We need to generalize a single execution $\xi(q_0, \tau)$ to compute $\text{Reach}(B_\delta(q_0), B_\varepsilon(\tau))$ where $B_\varepsilon(\tau)$ traces $\varepsilon$-equivalent to $\tau$

Hurdles

- $q_3$ and $q'_3$ may not satisfy the same actions
- Need to estimate impact of equivalent traces on similar (but not identical) states
A labeled transition system $\mathcal{A}$ is a tuple $\langle X \cup L, \Theta, A, \rightarrow \rangle$ where

- $X$: real-valued variables, $L$: finite-valued variables
- $Q = Val(X \cup L)$: the set of states,
- $\Theta$: initial states,
- $A$: finite set of actions,
- $\rightarrow: Q \times A \times Q$ is a transition relation, $guard(a) = \{ q \in Q \mid \exists q' \in Q, q \xrightarrow{a} q' \}$

A finite action sequences $\tau = a_0 a_1 \ldots, a_{n-1}$ is called a trace

A state $q_0 \in Q$ and $\tau$ uniquely specifies a potential execution $\xi_{q_0,\tau} = q_0, a_0, q_1, a_1, \ldots, a_{n-1}, q_n$

A valid execution satisfy $q_0 \in \Theta$ and for each $i, q_i \in guard(a_i)$
sensitivity to initial states: discrepancy

Discrepancy for action $a \in A$ is a continuous function $\beta_a : \mathbb{R}^+ \to \mathbb{R}^+$ such that for any $q_0, q_0'$,
- $|a(q_0) - a(q_0')| \leq \beta_a(|q_0 - q_0'|)$
- as $\delta \to 0$, $\beta_a(\delta) \to 0$

$\beta_a$ can be computed using Lipschitz constant, matrix measures, etc.
[Duggirala EMSOFT 13, Fan ACM TECS 16]

Proposition: Given a trace $\tau = a_0a_1 \ldots a_{n-1}$, for any $q_0, q_0'$,

$$|\xi_{q_0, \tau} - \xi_{q_0', \tau}| \leq \beta_{a_{n-1}} \ldots \beta_{a_0}(|q_0 - q_0'|)$$
example: platoon

A platoon of N cars on a single lane
Each car chooses 1 of 3 actions at each step: 
\(a\) (accelerate), \(b\) (brake), or \(c\) (cruise)

Car\(_1\) can choose any action at each time step

Others try to keep safe distance \(d\) with predecessor 
by choosing \(a\) if \(d > 50\), \(b\) if \(d < 30\), else \(c\)

Safety requirement \((U)\): Cars do not collide

Car\(_1\) has 3 choices: \(3^{10}\) executions of length 10
4 cars with different initial separation: 
\(3^4\) \(10\) executions

\[ x \left[ \begin{array}{ccc} 1 & \Delta t & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{array} \right] + \left[ \begin{array}{c} \text{acc}_1(\Delta t)^2/2 \\ \text{acc}_1 \Delta t \\ \text{acc}_2(\Delta t)^2/2 \\ \text{acc}_2 \Delta t \end{array} \right] = Ax + v(m), \text{ where} \\
m[1] = \text{accelerate}, m[2] = \text{brake}, \text{then acc}_1 > 0 \text{ acc}_2 < 0 \\
\text{For any} q, q', m, \beta_a(|q.x - q'.x|) = |A||q.x - q'.x|, \text{with} \\
\Delta t = 0.1, \beta_a(|q.x - q'.x|_2) = 1.06|q.x - q'.x|_2 \]
actions commute approximately

Two actions are $\varepsilon$-independent $a \sim^\varepsilon b$ if $ab(q_0).L = ba(q_0).L$ and $|ab(q_0).X - ba(q_0).X| < \varepsilon$

$\varepsilon$-independence is symmetric but not transitive
**ε-equivalent traces**

A trace $\tau = a_0 a_1 \ldots , a_{n-1}$ and an action $a$ are $\epsilon$-independent $\tau \sim^{\epsilon} a$ iff each $a_i \sim^{\epsilon} a$

**ε-equivalent traces:** $\tau \equiv^{\epsilon} \tau'$ if $\tau'$ can be constructed from $\tau$ by performing a sequence of swaps of consecutive $\epsilon$-independent actions

E.g., if $a_0 \sim^{\epsilon} a_1$ and $a_0 \sim^{\epsilon} a_2$, then $a_0 a_1 a_2 \equiv^{\epsilon} a_1 a_0 a_2 \equiv^{\epsilon} a_1 a_2 a_0$

$\equiv^{\epsilon}$ is a symmetric, reflexive and transitive relation of $\Sigma$

**Idea:**

- Use representative member $\tau$ of $\equiv^{\epsilon}$ to compute $\text{Reach}(q_0, B_{\epsilon}(\tau))$
- Use discrepancy to compute $\text{Reach}(B_{\delta}(q_0), B_{\epsilon}(\tau))$
boundung $\varepsilon$-equivalent executions

**Lemma**: For any initial state $q_0 \in Q$, action $a \in A$, trace $\tau \in \Sigma$, if $\tau \sim^\varepsilon a$, then $|\xi_{q_0,\tau a} - \xi_{q_0,a\tau}| \leq \gamma_{n-1}(\varepsilon)$,

where $\gamma_{n-1}(\varepsilon) = \sum_{i=0}^{n-1} \beta_i^\varepsilon$ and $\beta_{\text{max}} = \max_{b \in \tau} \beta_b$
(δ, ε)-trace equivalent discrepancy

Given \( q_0 \in Q \), trace \( \tau \in \Sigma \), and constants \( \delta, \epsilon \geq 0 \), \( r(\delta, \epsilon, q_0, \tau) > 0 \) is called a \((\delta, \epsilon)\)-trace equivalent discrepancy if for all \( q_0' \in B_\delta(q_0) \) and \( \tau \equiv^\epsilon \tau' \)

\[
\left| \xi_{q_0, \tau} - \xi_{q_0', \tau'} \right| \leq r
\]
computing \((\delta, \varepsilon)\)-discrepancy via earliest positions

**Lemma:** \((\delta, \varepsilon)\)-Ted for \(\xi_{q_0, \tau} a\) is given by

\[
r' = \begin{cases} 
\beta_a(r) & \text{Earliest}(\tau, a, \varepsilon) = \text{len}(\tau) \\
\beta_a(r) + \gamma_{\text{len}(\tau)-k-1}(\varepsilon) & \text{Earliest}(\tau, a, \varepsilon) < \text{len}(\tau)
\end{cases}
\]

where \(r\) is a \((\delta, \varepsilon)\)-Ted for \(\xi_{q_0, \tau}\).

**Earliest position** of \(a\) on \(\tau\) is \(\text{Earliest}(\tau, a, \varepsilon) = \min_{\phi \in e_{\tau} a, a \notin \eta} \text{len}(\phi)\)

E.g., if \(a_0 \sim^\varepsilon a_1\) and \(a_0 \sim^\varepsilon a_2\), then \(\text{Earliest}(a_0a_1, a_2, \varepsilon) = 1\)

\(\text{Earliest}(\cdot)\) can be computed using \(O(\text{len}(\tau)^2)\)
reachability

**Step 1:** Partition $\Theta$ to be $\delta_0$-balls
reachability

Step 1: Partition $\Theta$ to be $\delta_0$-balls

Step 2: For each $\delta_0$-ball $B_{\delta_0}(q_0)$, construct $\text{Reach}(B_{\delta_0}(q_0), t)$
  - For each $t \leq T$, find all representative traces $\{\tau_t\}$
  - Representative traces are mutually non-$\varepsilon$-equivalent
  - For each $\tau_t$, bloat $\xi_{q_0,\tau_t}.lstate$ with its $(\delta_0, \varepsilon)$-ted Union, check safety, refine...
Suppose 3 actions with $a \sim^\varepsilon b$ and $a \sim^\varepsilon c$. All of them are enabled at each steps. Reach set at $t$ is stored as tuples $R_t = \{(\tau_t, q_t, \delta_t)\}$, with $q_t = \xi_{q_0,\tau_t}$ and $\delta_t$ is a $(\delta_0, \varepsilon)$-ted for $\xi_{q_0,\tau_t}$.

If there are $k$ actions in total and they are mutually $\varepsilon$-independent, then $R_t$ contains at most $\binom{t+k-1}{k-1}$ tuples, compared with $k^t$ potential traces.

The algorithm can reduce the number of executions explored by $O(t!)$.
soundness and precision

Theorem (soundness): The reachability algorithms indeed computes an over-approximation of $\text{Reach}(\Theta, \Sigma)$.

Theorem (precision): The over-approximation can be made arbitrarily precise by reducing the size of $\delta_0, \varepsilon$.

(As $\delta_0$ and $\varepsilon$ go to 0, the algorithm actually converges to a simulation algorithm which simulates every valid execution from each initial state)
## Experiments: Platoon

|                  | Approximate POR | Brute Force (single initial) |
|------------------|-----------------|------------------------------|
| **2-car platoon**|                 |                              |
| # traces explored| 43758 (max)     | 9^{10}                       |
| Run time         | 5.1ms           | 2.9s                         |
| **4-car platoon**|                 |                              |
| # traces explored| 7986            | 81^{10}                      |
| Run time         | 62.3ms          | 6.2s                         |

![Graphs showing car positions over time for 2-car and 4-car platoons.](image)
conclusion

- Approximate partial order reduction generalizes the traditional independence by allowing approximately commuting actions.

- This notion works naturally with reachability algorithms that generalize individual executions to cover $\delta$-close initial states, and follow different, but $\epsilon$-independent, action sequences.

- Over-approximations made arbitrarily precise by reducing $\delta, \epsilon$.

Future directions
- Combine with symmetry reduction
- Applications to autonomous vehicle interactions
collaborators

Chuchu Fan

Zhenqi Huang