Decidability and undecidability of timed devices with stopwatches

Mizuhito Ogawa
With Li Guoqiang, Shoji Yuen

18.9.2015
Plan of this talk

• Reachability of automata with continuous parameters.
  ✓ Decidable classes are often variants of timed automata ($x' = 1$), including recursive timed devices.
  ✓ Undecidable by introducing stopwatches ($x' = 0$ or $1$).
    – Bounded numbers of clocks recover decidability, e.g., TA with 2 stopwatches, NeTA-F with single global clock.

• Techniques
  ✓ Undecidability: Wrapping, divergence of regions.
  ✓ Decidability:
    – WQO over regions (WSTS), semi-bisimulation
Automaton with continuous parameters

- Each transition may have guards \((x > c, y \leq c)\), reset \((x \leftarrow [c,c'], x \leftarrow y)\) under the relation \(x' = f(x), c,c' \in \mathbb{N}\).

- Differential \(x'\) (slope)

  ✓ Timed automata: \(x' = 1\) (stopwatch: \(x' = 0\) or 1)
  ✓ Rectangular hybrid automata: \(x' = \text{constant}\)
    - When \(x'\) changes, \(x\) is reset to 0 (strong reset)
      \(\Rightarrow\) reduced to timed automata (rectangular region)
  ✓ (Semi-)Linear hybrid automata: \(x' = Ax\)
    - “\(o\)-minimal” and “strong reset” give discretization.

Initially, \(x\) is set to 0
Timed automata (Alur, et.al. 94)

- Press quickly twice, the light will be brightened.
  ✓ Add time constraints: e.g., quickly = “less-than 1”

- It accepts, e.g., (press,2.1) (press,2.53) (press,8.7)
  \[ x=0 \quad x=2.1; x\leftarrow 0, \quad x=0.43 \quad x=6.17 \]

- Reachability to a state \( q \leftrightarrow \exists \text{timed run to } q. \)
Example: Timed automaton (2-clocks)

- It accepts timed words, in which
  - $c$ occurs after a delay of at least 2 from last $b$, and
  - $d$ occurs within 3 from last $a$.

- **Remark**: 1-clock is *not enough* for these timed words. Actually, expressiveness enlarges depending to the number of clocks.
Non-examples: Timed automata

- Delay between the first and the second event $a$ is the same as the delay between the second and the third.
  - e.g., a timed word $(a, t)(a, t + t')(a, t + 2t')$

- Each occurrence of $a$ has the corresponding occurrence of $a$ of the delay of 1.
  - e.g., unboundedly many occurrences of $a$ in a unit.

Infinite clocks needed
Decidable properties of timed automata

• Decidable
  ✓ Reachability / emptiness
    – Discretization (region construction)
  ✓ Inclusion / universality (single clock)
    – Not closed by determinization / complement.

• Undecidable
  ✓ Inclusion / universality (multiple clocks)
Complement fails

- Some occurrence of $a$ does not have the occurrence of $a$ of the delay 1.

- Complement: Each occurrence of $a$ has the corresponding occurrence of $a$ of the delay 1.

Infinite clocks needed
Ideas to show decidability / undecidability
Bisimulation and discretization

- Bisimulation between continuous & discrete systems
  - Two clock valuations \( \nu \sim \nu' \) iff \( \nu + t \) and \( \nu' + t \)
    satisfy the same clock constraints for each \( t \geq 0 \).
  - For \( k \)-clocks, the congruence \( \sim \) over \( (\mathbb{R}_{\geq 0})^k \) gives discretization.

- Discretization
  - If discretization converges, reachability is decidable.
Region construction for TA

- Upper/lower triangles and boundaries of unit tiles up to $C$ are regions, where $C$ is the largest integer appearing in constraints or resets.

$\nu \sim \nu'$ iff they hold the same set of constraints of the form, for $c \leq C$, $x_i < c$, $x_i = c$, $x_i - x_j < c$, $x_i - x_j = c$.
On-demand zone construction

• The reachability is PSPACE-complete (with 3 clocks).

Set: $Q/\sim = \{Q_O \cap Q_F, Q_O \setminus Q_F, Q_F \setminus Q_O, Q \setminus (Q_O \cup Q_F)\}$
while: $\exists P, P' \in Q/\sim$ and $\sigma \in \Sigma$ such that $\emptyset \neq P \cap \text{Pre}_\sigma(P') \neq P$
    set: $P_1 = P \cap \text{Pre}_\sigma(P'), P_2 = P \setminus \text{Pre}_\sigma(P')$
    refine: $Q/\sim = (Q/\sim \setminus \{P\}) \cup \{P_1, P_2\}$
end while:

\[
\begin{cases}
Q_0 = \text{initial configurations } (P_{\text{init}} \times 0^k)
Q_F = \text{final configurations } (P_f \times R^k)
\end{cases}
\]
Undecidability with extensions on constraints

- **Def.** A *diagonal* (clock) constraint is of the forms “x–y ◇ c” for ◇ ∈ {>, ≥, =, ≤, <}.

- The number of region becomes infinite. Reachability becomes undecidable with
  - “x = 2y”
  - “x + y ◇ c” (with ≥ 4 clocks).
  - **Stopwatch** (x’ = 0)
  - Update “x ← x-1”.
  - Update “x ← x+1” + *diagonal constraints*
    - “x ← x+1” only keeps decidability.
TA with stopwatches

- *Wrapping*: Simulating two counter machine by $2^i \cdot 3^j$ with 2 clocks + 1 stopwatch.
Example divergence of regions (*Updates*)

- Update $x \leftarrow x-1$
- Diagonal constraints, e.g. $x < y$, with Update $x \leftarrow x+1$
Decidability when discretization diverges

- When discretization has infinite regions
  - WQO over regions (WSTS)
  - Semi-bisimulation

- Semi-bisimulation (for reachability)

```
\[
\text{continuous} \quad t \xrightarrow{\text{continuous}} t' \quad t_0 - \ldots \xrightarrow{\exists} t'_m \xrightarrow{\exists} t'_{m+1}
\]
```

```
\[
\text{discrete} \quad s \xrightarrow{\exists} s' \quad s_0 - \ldots \xrightarrow{\exists} s_m \xrightarrow{\exists} s_{m+1}
\]
```

- Example: Inclusion/universality of single-clock TA.
  - Its discretization satisfies bisimulation.
Well-structured transition systems (WSTS)

- **Def.** A WSTS \((S, \Delta)\) consists of
  - WQO \((S, \leq)\) (a possibly *infinite* states)
  - \(\Delta \subseteq S \times S\) monotonic transitions
    i.e., \(s_1 \rightarrow s_2 \wedge s_1 \leq t_1\) imply \(\exists t_2. t_1 \rightarrow t_2 \wedge s_2 \leq t_2\)

- **Theorem.** Coverability of a WSTS is decidable.
  [Finkel87, Abdulla, et al. 00, Finkel-Schnoebelen 01]

- Determinization of single-clock TA is *semi*-bisimilar to a *downward-compatible* WSTS.
  i.e., \(t_1 \rightarrow t_2 \wedge s_1 \leq t_1\) imply \(\exists s_2. s_1 \rightarrow s_2 \wedge s_2 \leq t_2\)
  \(\Rightarrow\) Universality.
Timed recursive devices
Timed Recursive Devices: *Invoke (queue)*

- Task automata (for schedulability)
  - Reachability is undecidable
    - Reasonable assumptions for schedulability reduces the problems to finite products of TAs.
      - Deadline is bounded.
      - Minimum (positive) execution time is fixed.
Timed Recursive Devices: *Interrupt (stack)*

- Pushdown systems with a finite set of TAs, which are control states and stack alphabet.

- Interrupted TAs are on the stack
  - Timed Recursive State Machine (TRSM) Benerecetti, et al. 10
  - Recursive Timed Automata (RTA) Trivedi, Wojtczak 10
  - Nested Timed Automata (NeTA) Li, Cai, O, Yuen 15
Global and local clocks

• For \{TA_1, \ldots, TA_m\}, we assume that each TA_i has \( k \)-local clocks.
  ✓ Timed recursive devices can have global clocks.
  ✓ For (possibly global) clocks \( x, z \), we can set \( z \leftarrow x, x \leftarrow z \).

• \textbf{Remark}: Global clocks work as channels to exchange local clock values of TA in the stack.
Storing local clock values

• All clocks are global (i.e., a working TA keeps them)
  ✓ Call-by-reference RTA

• All clocks are local
  ✓ In the stack *frozen* : Call-by-value RTA
  ✓ In the stack *proceeding* : NeTA
  ✓ Either *proceeding* or *frozen* : Local TRSM

• Clocks are either global or local
  ✓ Either *call-by-reference* or *-value* : Glitch-free RTA
  ✓ Either *proceeding* or *frozen* : NeTA-F

*Can simulate stopwatches*
Decidability and undecidability of NeTA-F

- **NeTA-F**: Extension of NeTA such that
  - PDA with **global clocks**, and
  - States = Stack alphabet = \{TA_1, TA_2, \ldots, TA_n\}
  - When pushed, TA can select **frozen** or **proceeding** (accordingly all its local clocks are **frozen** or **proceeding**)

- **Theorem** The reachability of NeTA-F is
  - **Undecidable**, with **multiple global clocks**.
  - **Decidable**, with a **single global clock**.
    - 1 clock + 1 stopwatch are not enough for wrapping.
      (Communication between 2 TA has only single one-directed channel.)
Conclusion

• Reachability of automata with continuous parameters.
  ✓ Main decidable classes are variants of *timed automata* \((x' = 1)\), including *recursive timed devices*.
  ✓ Undecidable by introducing stopwatches \((x' = 0\) or 1).  
    – Bounded numbers of clocks recover decidability, e.g., TA with 2 stopwatches, NeTA-F with single global clock.

• Techniques
  ✓ Undecidability: Wrapping, divergence of regions.
  ✓ Decidability:
    – WQO over regions (WSTS), *semi*-bisimulation
Thank you!