Computing Linear Arithmetic Representation of Reachability Relation of One-counter Automata

Authors: **Xie Li**, Taolue Chen, Zhilin Wu and Mingji Xia

SETTA 2020: Guangzhou, China, November 24-28, 2020

ISCAS 中国科学院软件研究所 Institute of Software Chinese Academy of Sciences

Birkbeck UNIVERSITY OF LONDON

中国科学院大学 University of Chinese Academy of Sciences

Overview

- Introduction to One-counter Automata(OCA) and its Reachability Problem.

Overview

- Introduction to One-counter Automata(OCA) and its Reachability Problem.
- Computing the Reachability Relation of OCA.

Overview

- Introduction to One-counter Automata(OCA) and its Reachability Problem.
- Computing the Reachability Relation of OCA.
- Introduction to Tool ORAREACH.
- Experimental Results of our Tool OCAREACH.

What is One-counter Automata(OCA)

- DFA with a **counter** where counter value is **non-negative**.

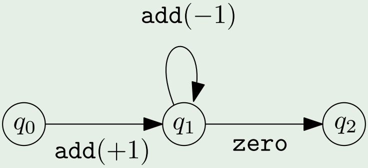
What is One-counter Automata(OCA)

- DFA with a **counter** where counter value is **non-negative**.
- Transitions: $q \xrightarrow{\texttt{Op}} q'$ where $\texttt{Op} \in \{\texttt{add}(+1), \texttt{add}(-1), \texttt{zero}\}$

What is One-counter Automata(OCA)

- DFA with a **counter** where counter value is **non-negative**.
- Transitions: $q \xrightarrow{\texttt{Op}} q'$ where $\texttt{Op} \in \{\texttt{add}(+1), \texttt{add}(-1), \texttt{zero}\}$

Example (OCA)



Semantic of OCA

Semantic of OCA: A transition system where the configuration is of the form (q, c) and counter changes corresponds to OCA.

$$(q_1, c_1) \rightarrow_{\mathcal{A}} (q_2, c_2)$$

if $q_1 \xrightarrow{\texttt{add}(+1)} q_2$ in the OCA and $c_1 + 1 = c2$, or
if $q_1 \xrightarrow{\texttt{zero}} q_2$ and $c_1 = c_2 = 0$.

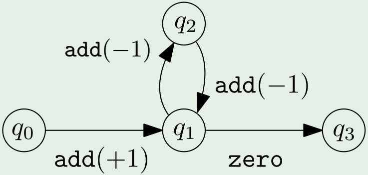
Reachability

Reachability Problem of OCA: whether $(q_s, c_s) \rightarrow_{\mathcal{A}}^{*} (q_t, c_t)$

Reachability

Reachability Problem of OCA: whether $(q_s, c_s) \rightarrow^*_{\mathcal{A}} (q_t, c_t)$

Example

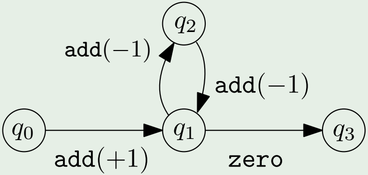


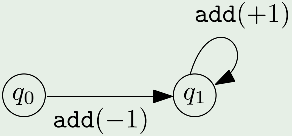
- $(q_3, 0)$ is reachable from $(q_0, 1)$.
- $(q_3, 0)$ is not reachable from $(q_0, 0)$

Reachability

Reachability Problem of OCA: whether $(q_s, c_s) \rightarrow^*_{\mathcal{A}} (q_t, c_t)$

Example





- $(q_3, 0)$ is reachable from $(q_0, 1)$.
- $(q_3, 0)$ is not reachable from $(q_0, 0)$

Due to the non-negative requirement, $(q_1, 1)$ is not reachable from $(q_0, 0)$

Weighted Directed Graph, Path Flow and Support

- An OCA can be regarded as a weighted directed graph $G_{\mathcal{A}} = (V, E, \eta)$.

Weighted Directed Graph, Path Flow and Support

- An OCA can be regarded as a weighted directed graph $G_{\mathcal{A}} = (V, E, \eta)$.
- **Path**: a sequence of vertices $v_0 \cdot v_1 \cdots v_k$ where $(v_i, v_{i+1}) \in E$.

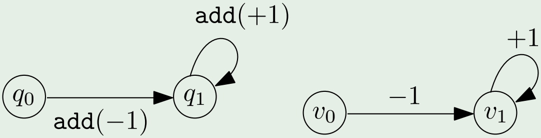
Weighted Directed Graph, Path Flow and Support

- An OCA can be regarded as a weighted directed graph $G_{\mathcal{A}} = (V, E, \eta)$.
- **Path**: a sequence of vertices $v_0 \cdot v_1 \cdots v_k$ where $(v_i, v_{i+1}) \in E$.
  - Weight of path
  - Drop of path

Weighted Directed Graph, Path Flow and Support

- An OCA can be regarded as a weighted directed graph $G_{\mathcal{A}} = (V, E, \eta)$.
- **Path**: a sequence of vertices $v_0 \cdot v_1 \cdots v_k$ where $(v_i, v_{i+1}) \in E$.
  - Weight of path
  - Drop of path
- **Flow**: a function $f : E \to \mathbb{N}$.

Example



- path: $v_0 \cdot v_1 \cdot v_1$
- weight: $+1$, drop: -1

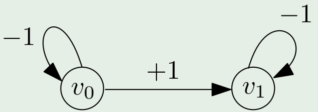
Path Flow and Support

- s-t **path flow**: the flow corresponds to a path.
- Support: edge-induced subgraph of path.

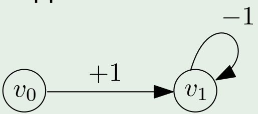
Path Flow and Support

- *s*-*t* **path flow**: the flow corresponds to a path.
- Support: edge-induced subgraph of path.

Example



- Support:



- Path: $v_0 \cdot v_1 \cdot v_1$
- Pathflow: $f(v_0, v_0) = 0$
  $f(v_0, v_1) = 1$
  $f(v_1, v_1) = 2$
- Weight:
  $weight(f) = \Sigma_{e \in E} f(e) \cdot \eta(e)$

The Difficulty of the Reachability Problem

NON-NEGATIVE

NON-NEGATIVE

If we do not require the non-negative of counter.

- Flow is a path flow $\Leftrightarrow$ Requirements on flow.

The Difficulty of the Reachability Problem

NON-NEGATIVE

If we do not require the non-negative of counter.

- Flow is a path flow $\Leftrightarrow$ Requirements on flow.
  - $v_s \cdots v_t$ where $s \neq t$
  - $v_s \cdots v_s$

NON-NEGATIVE

If we do not require the non-negative of counter.

- Flow is a path flow $\Leftrightarrow$ Requirements on flow.
  - $v_s \cdots v_t$ where $s \neq t$
  - $v_s \cdots v_s$
- Weight equals to the value change.

The Difficulty of the Reachability Problem

NON-NEGATIVE

If we do not require the non-negative of counter.

- Flow is a path flow $\Leftrightarrow$ Requirements on flow.
  - $v_s \cdots v_t$ where $s \neq t$
  - $v_s \cdots v_s$
- Weight equals to the value change.

Non-negative implies the constraint: everywhere along the path, the counter need to be non-negative.

Certificate of the Reachability

Use **path flow** as certificate of OCA reachability problem.

Certificate of the Reachability

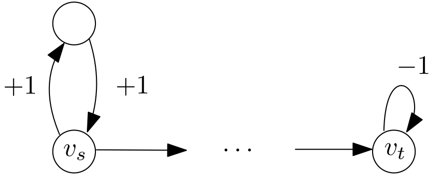
Use **path flow** as certificate of OCA reachability problem.

- Type-1 Certificate:
  - Flow is a path flow.
  - No positive cycle.
  - Weight correctly added.
  - Path flow can be divided into edge decompositions (which implies non-negative).

Certificate of the Reachability

Use **path flow** as certificate of OCA reachability problem.

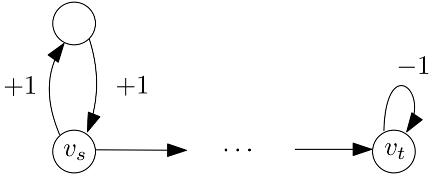
- Type-1 Certificate:
  - Flow is a path flow.
  - No positive cycle.
  - Weight correctly added.
  - Path flow can be divided into edge decompositions (which implies non-negative).
- Type-3 Certificate:



Certificate of the Reachability

Use **path flow** as certificate of OCA reachability problem.

- Type-1 Certificate:
  - Flow is a path flow.
  - No positive cycle.
  - Weight correctly added.
  - Path flow can be divided into edge decompositions (which implies non-negative).
- Type-3 Certificate:



- Type-2 Certificate: reverse of type-1 certificate

Edge Decomposition

Definition

Edge Decomposition Edge decomposition of a path flow f is

Edge Decomposition

Definition

Edge Decomposition Edge decomposition of a path flow f is

- $f = \Sigma_{i \in |E|} f_i + f_{e_i}$ where f_i is also a path flow, $e_i \in E$.

Edge Decomposition

Definition

Edge Decomposition Edge decomposition of a path flow f is

- $f = \Sigma_{i \in |E|} f_i + f_{e_i}$ where f_i is also a path flow, $e_i \in E$.
- e_i does not appear in support of f_j where $j > i$.

Edge Decomposition

Definition

Edge Decomposition Edge decomposition of a path flow f is

- $f = \Sigma_{i \in |E|} f_i + f_{e_i}$ where f_i is also a path flow, $e_i \in E$.
- e_i does not appear in support of f_j where $j > i$.
- $weight(f_i) + weight(e_i) \geq 0$ for all i.

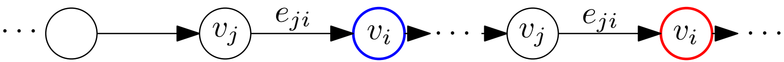
Edge Decomposition

Definition

Edge Decomposition Edge decomposition of a path flow f is

- $f = \Sigma_{i \in |E|} f_i + f_{e_i}$ where f_i is also a path flow, $e_i \in E$.
- e_i does not appear in support of f_j where $j > i$.
- $weight(f_i) + weight(e_i) \geq 0$ for all i.

This definition implies the non-negative requirement of Type-1 certificate.



Decidability of Reachability of OCA

Theorem (Haase)

The reachability can be solved iff we can find a certificate that is of the form

$$(\mathit{Type\text{-}1})^{n_1}(\mathit{Type\text{-}3})^{n_3}(\mathit{Type\text{-}2})^{n_2}$$

where $n_i \in \{0, 1\}$

Reachability Relation of OCA

Reachability Relation of OCA:

$$\phi_{\mathcal{A},q_s,q_t}(x_s, x_t)$$

$\phi_{\mathcal{A},q_s,q_t}(c_s, c_t)$ is true iff $(q_s, c_s) \rightarrow^*_{\mathcal{A}} (q_t, c_t)$.

How to Reduce the Reachability Relation to PA Formula

$$\phi_{\mathcal{A},q_s,q_t}(x_s, x_t) = \exists (y_e)_{e\in E}.\phi^{T1RC} \vee \phi^{T2RC} \vee \phi^{T3RC} \vee \cdots$$

Type-3 certificate:

$$\phi^{T3RC}_{q_s,q_t}(x_s, x_t, (y_{e,1})_{e\in E})$$

- Positive Cycle.
- Weight of Path flow equals to the change.
- Negative Cycle.

How to Reduce the Reachability Relation to PA Formula

Type-1 certificate:

$$\phi^{T1RC}_{q_s,q_t}(x_s, x_t, (y_{e,1})_{e\in E})$$

- Weight equals to the counter change.

How to Reduce the Reachability Relation to PA Formula

Type-1 certificate:

$$\phi^{T1RC}_{q_s,q_t}(x_s, x_t, (y_{e,1})_{e\in E})$$

- Weight equals to the counter change.
- Existence of edge decomposition of the path flow.

How to Reduce the Reachability Relation to PA Formula

Type-1 certificate:

$$\phi^{T1RC}_{q_s,q_t}(x_s, x_t, (y_{e,1})_{e\in E})$$

- Weight equals to the counter change.
- Existence of edge decomposition of the path flow.
  - The order of the edge last appears. *idx*.

How to Reduce the Reachability Relation to PA Formula

Type-1 certificate:

$$\phi^{T1RC}_{q_s,q_t}(x_s, x_t, (y_{e,1})_{e\in E})$$

- Weight equals to the counter change.
- Existence of edge decomposition of the path flow.
  - The order of the edge last appears. *idx*.
  - Correct concatenation of the splitted path flows.

OCAReach: Experimental Evaluation

Implemented in Java.
INPUT: file describing the OCA.
OUTPUT: a PA formula ϕ representing reachability relation.

- Experiment on handcrafted cases.

state num.	2	2	2	2	3	3	4	4	4
transtion num.	1	2	2	5	2	3	3	3	6
zero-test num.	0	1	1	0	0	1	1	1	1
time (s)	0.066	0.062	0.078	0.076	0.066	0.072	0.061	0.079	0.093
size (kB)	0.302	0.404	0.697	0.302	0.133	0.929	0.348	0.325	2.592

state num.	5	6	6	6	7	8	10	10	
transtion num.	6	6	7	8	9	7	11	11	
zero-test num.	1	2	2	2	2	2	2	3	
time (s)	0.087	0.078	0.106	0.091	0.106	0.090	0.116	0.117	
size (kB)	2.057	2.469	7.457	3.078	6.427	4.807	8.443	7.515	

- On random cases.

Conclusion and Future Work

We built the gap between the theory and implementation by the tool OCAREACH

Future work:

- Optimize our tool to improve the efficiency.
- More cases and benchmark for experiment.