Rytter for CFPQ

Semyon Grigorev
Saint Petersburg State University
St. Petersburg, Russia
semen.grigorev@jetbrains.com

Ekaterina Shemetova
Saint Petersburg State University
St. Petersburg, Russia
nozhkin.ii@gmail.com

ABSTRACT

Abstract

CCS CONCEPTS

• **Information systems** → **Graph-based database models**; **Query languages for non-relational engines**; • **Software and its engineering** → *Functional languages*; • **Theory of computation** → *Grammars and context-free languages*;

KEYWORDS

Graph Databases, Language-Constrained Path Problem, Context-Free Path Querying, Parser Combinators, Generalized LL, GLL, Neo4J, Scala

ACM Reference Format:

Semyon Grigorev and Ekaterina Shemetova. 2018. Rytter for CFPQ. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

1 INTRODUCTION

Two steps reduction of CFPQs to Boolean matrix multiplication. First step is reduction of arbitrary CFPQ to Dyck query. Second step is adoptation Rytter's results from [**?**] for graph.

2 FROM ARBITRARY CFPQ TO DYCK QUERY

This reduction is inspired by the construction described in [1].

Consider a context-free grammar $\mathcal{G} = (\Sigma, N, P, S)$ in BNF where Σ is a terminal alphabet, N is a nonterminal alphabet, P is a set of productions, $S \in N$ is a start nonterminal. Also we denote a directed labeled graph by $G = (V, E, L)$ where $E \subseteq V \times L \times V$ and $L \subseteq \Sigma$.

We should construct new input graph G' and new grammar $\mathcal{G}'$ such that $\mathcal{G}'$ specifies a Dyck language and there is a simple mapping from CFPQ($\mathcal{G}', G'$) to CFPQ($\mathcal{G}, G$). Step-by-step example with description is provided below.

Let the input grammar is

$$S \rightarrow a\ S\ b \mid a\ C\ b$$
$$C \rightarrow c \mid C\ c$$

The input graph is presented in fig. **??**

(1) Let $\Sigma_{()} = \{t_(, t_) | t \in \Sigma\}$.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM...$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

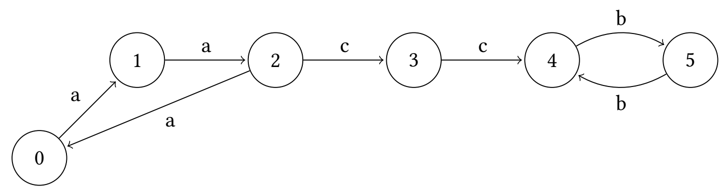


Figure 1: The input graph

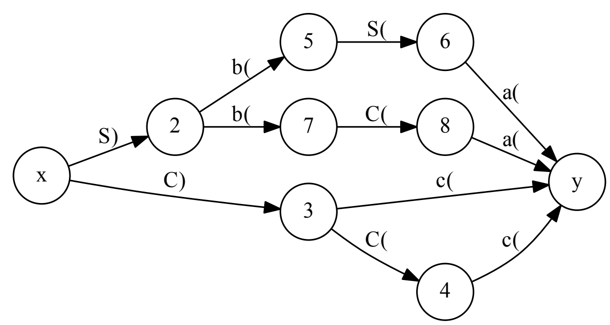


Figure 2: The input graph

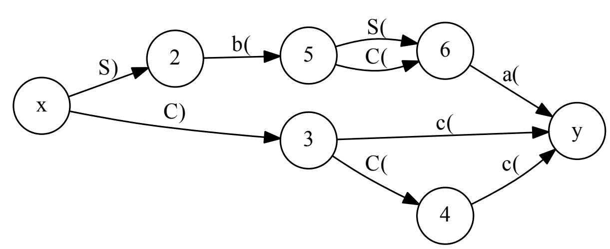


Figure 3: The input graph

(2) Let $N_{()} = \{N_(, N_) | N \in N\}$.
(3) Let $M_{\mathcal{G}} = (V_{\mathcal{G}}, E_{\mathcal{G}}, L_{\mathcal{G}})$ is a directed labeled graph, where $L_{\mathcal{G}} \subseteq (\Sigma_{()} \cup N_{()})$. This graph is created the same manner as described in [1] but we do not require the grammar be in CNF. Let $x \in V_{\mathcal{G}}$ and $y \in V_{\mathcal{G}}$ is "start" and "final" vertices respectively. This graph may be treated as a finite automaton, so it can be minimized and we can compute an ε-closure if the input grammar contains ε productions. The graph $M_{\mathcal{G}}$ for our example is:
The minimized graph:
(4) For each $v \in V$ create $M^v_{\mathcal{G}}$: unique instance of $M_{\mathcal{G}}$.

(5) New graph $G^{'}$ is a graph G where each label t is replaced with $t^{i}_{)}$ and some additional edges are created:
- Add an edge $(v', S_{(}, v)$ for each $v \in V$.
- And the respective $M^{v}_{\mathcal{G}}$ for each $v \in V$:
  - reattach all edges outgoing from x^{v} ("start" vertex of $M^{v}_{\mathcal{G}}$) to v;
  - reattach all edges incoming to y^{v} ("final" vertex of $M^{v}_{\mathcal{G}}$) to v.

  New input graph is ready:

(6) New grammar $\mathcal{G}' = (\Sigma^{'}, N', P', S')$ where $\Sigma^{'} = \Sigma_{()} \cup N_{()}$, $N' = \{S'\}$, $P' = \{S' \to b_{(}\ S'\ b_{)}; S' \to b_{(}\ b_{)} \mid b_{(}, b_{)} \in \Sigma^{'}\} \cup \{S' \to S'\ S'\}$ is a set of productions, $S' \in N'$ is a start nonterminal.

Now, if CFPQ$(\mathcal{G}', G')$ contains a pair (u_0', v') such that $e = (u_0', S_{(}, u_1') \in E'$ is an extension edge (step 5, first subitem), then $(u_1', v') \in$ CFPQ$(\mathcal{G}, G)$. In our example, we can find the following path: $7 \xrightarrow{S_{(}} 1 \xrightarrow{S_{)}} 22 \xrightarrow{b_{(}} 25 \xrightarrow{C_{(}} 26 \xrightarrow{a_{(}} 1 \xrightarrow{a_{)}} 2 \xrightarrow{C_{)}} 33 \xrightarrow{C_{(}} 34 \xrightarrow{c_{(}} 2 \xrightarrow{c_{)}} 3 \xrightarrow{C_{)}} 43 \xrightarrow{c_{(}} 3 \xrightarrow{c_{)}} 4 \xrightarrow{b_{)}} 5$. Edge $7 \xrightarrow{S_{(}} 1$ is the extension, so (1,5) should be in CFPQ$(\mathcal{G}, G)$ and it is true.

3 GRAPH INPUT

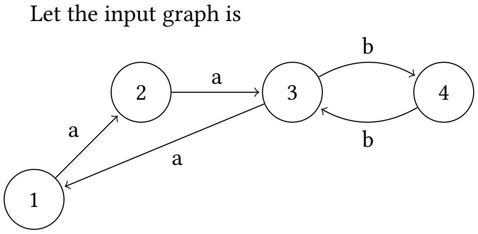
Let the input grammar is

$$S \to a\ S\ b$$
$$S \to a\ b$$

The input grammar in CNF is

$$\begin{aligned} S &\to A\ S_1 \\ S_1 &\to S\ B \\ S &\to A\ B \\ A &\to a \\ B &\to b \end{aligned}$$

Let the input graph is



The *IMPLIED* relation:

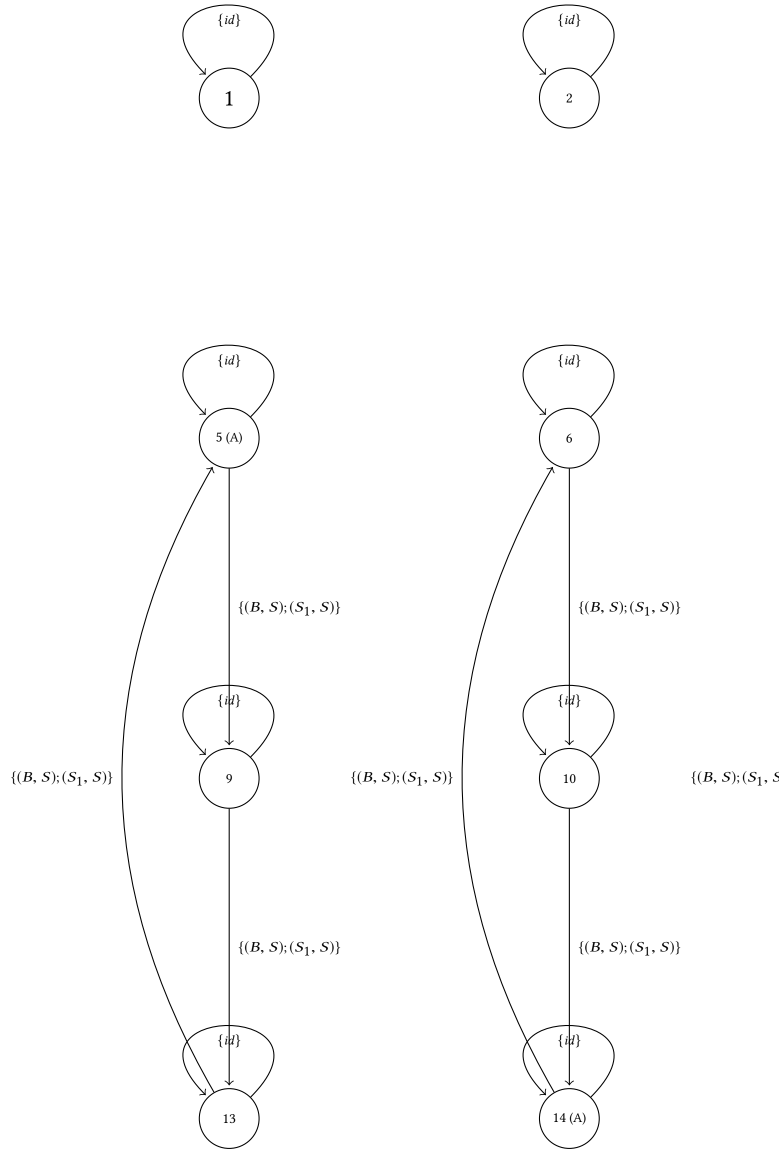
$(B,2,3) \Rightarrow (S,1,3)$	$(B,2,4) \Rightarrow (S,1,4)$	$(B,2,2) \Rightarrow (S,1,2)$	$(B,2,1) \Rightarrow (S,1,1)$
$(B,3,4) \Rightarrow (S,2,4)$	$(B,3,3) \Rightarrow (S,2,3)$	$(B,3,2) \Rightarrow (S,2,2)$	$(B,3,1) \Rightarrow (S,2,1)$
$(B,1,2) \Rightarrow (S,3,2)$	$(B,1,3) \Rightarrow (S,3,3)$	$(B,1,4) \Rightarrow (S,3,4)$	$(B,1,1) \Rightarrow (S,3,1)$
$(S_1,2,3) \Rightarrow (S,1,3)$	$(S_1,2,4) \Rightarrow (S,1,4)$	$(S_1,2,2) \Rightarrow (S,1,2)$	$(S_1,2,1) \Rightarrow (S,1,1)$
$(S_1,3,4) \Rightarrow (S,2,4)$	$(S_1,3,3) \Rightarrow (S,2,3)$	$(S_1,3,2) \Rightarrow (S,2,2)$	$(S_1,3,1) \Rightarrow (S,2,1)$
$(S_1,1,2) \Rightarrow (S,3,2)$	$(S_1,1,3) \Rightarrow (S,3,3)$	$(S_1,1,4) \Rightarrow (S,3,4)$	$(S_1,1,1) \Rightarrow (S,3,1)$
$(A,2,3) \Rightarrow (S,2,4)$	$(A,1,3) \Rightarrow (S,1,4)$	$(A,3,3) \Rightarrow (S,3,4)$	$(A,4,3) \Rightarrow (S,4,4)$
$(A,3,4) \Rightarrow (S,3,3)$	$(A,4,4) \Rightarrow (S,4,3)$	$(A,2,4) \Rightarrow (S,2,3)$	$(A,1,4) \Rightarrow (S,1,3)$
$(S,2,3) \Rightarrow (S_1,2,4)$	$(S,1,3) \Rightarrow (S_1,1,4)$	$(S,3,3) \Rightarrow (S_1,3,4)$	$(S,4,3) \Rightarrow (S_1,4,4)$
$(S,3,4) \Rightarrow (S_1,3,3)$	$(S,4,4) \Rightarrow (S_1,4,3)$	$(S,2,4) \Rightarrow (S_1,2,3)$	$(S,1,4) \Rightarrow (S_1,1,3)$

Grid:

We should introduce the *id* implication such that for every $A \in$ IMPLIED

- $id \times A = A \times id$

In order to compute transitive closure in logarithmic time we add self-loop with weight $\{id\}$ to each vertex.



Note that our graph is a Cartesian product of tho graph H and V with respective matrices.

$$H = \begin{pmatrix} \{id\} & \varnothing & \varnothing & \varnothing \\ \varnothing & \{id\} & \varnothing & \varnothing \\ \varnothing & \varnothing & \{id\} & \{(A,S);(S,S_1)\} \\ \varnothing & \varnothing & \{(A,S);(S,S_1)\} & \{id\} \end{pmatrix}$$

$$V = \begin{pmatrix} \{id\} & \varnothing & \varnothing & \varnothing \\ \varnothing & \{id\} & \{(B,S);(S_1,S)\} & \varnothing \\ \varnothing & \varnothing & \{id\} & \{(B,S);(S_1,S)\} \\ \varnothing & \{(B,S);(S_1,S)\} & \varnothing & \{id\} \end{pmatrix}$$

Matrix of $G = V \otimes I + I \otimes H$ where I is identity matrix of size $n \times n$ and $\otimes$ is a Kronecker product.

One step is APSP (or transitive closure) of G. It can be computed as $(V \otimes I + I \otimes H)^{(n^2)}$. It can be "over approximated" as $M = (V^{(n^2)} \otimes I +$

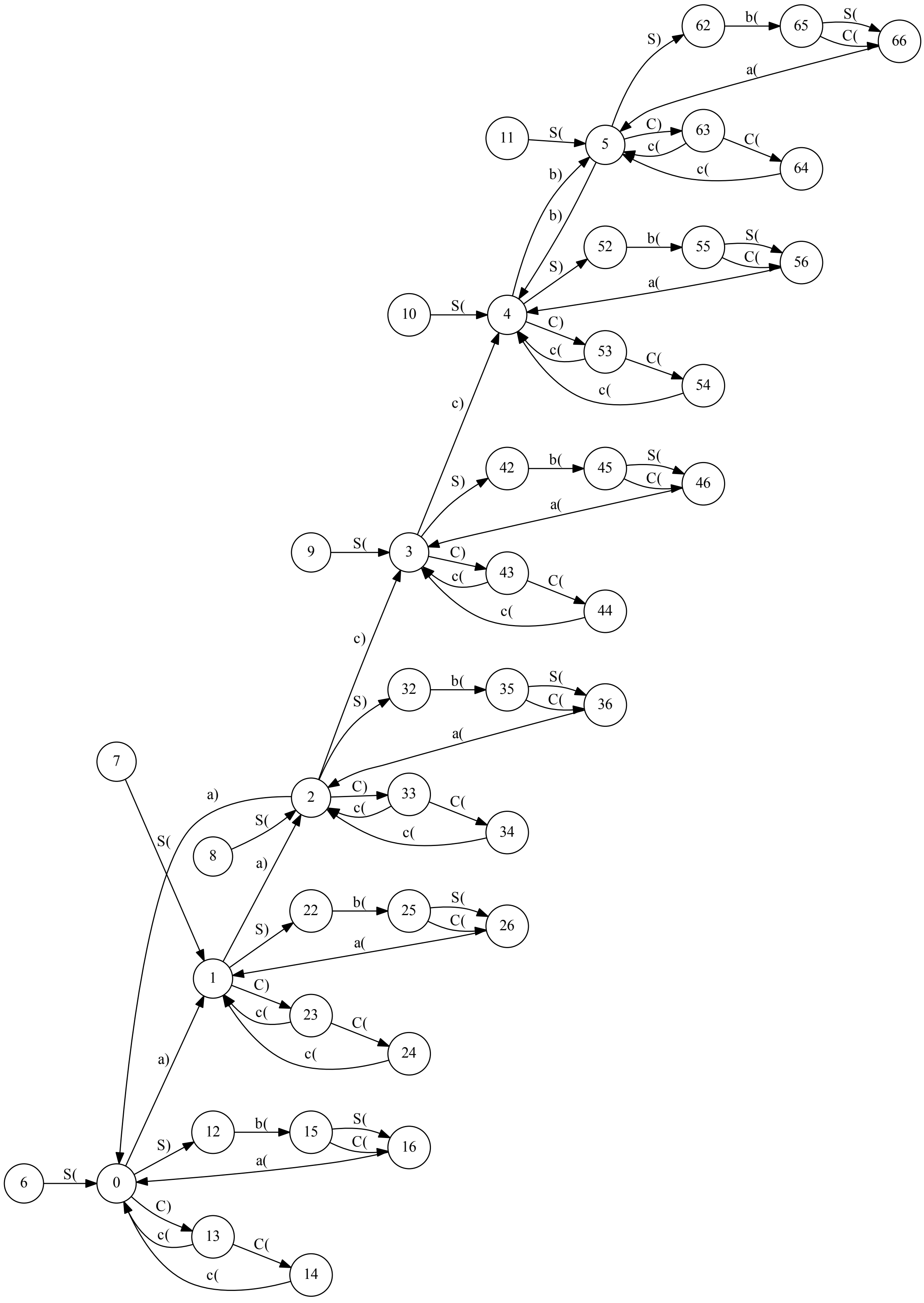
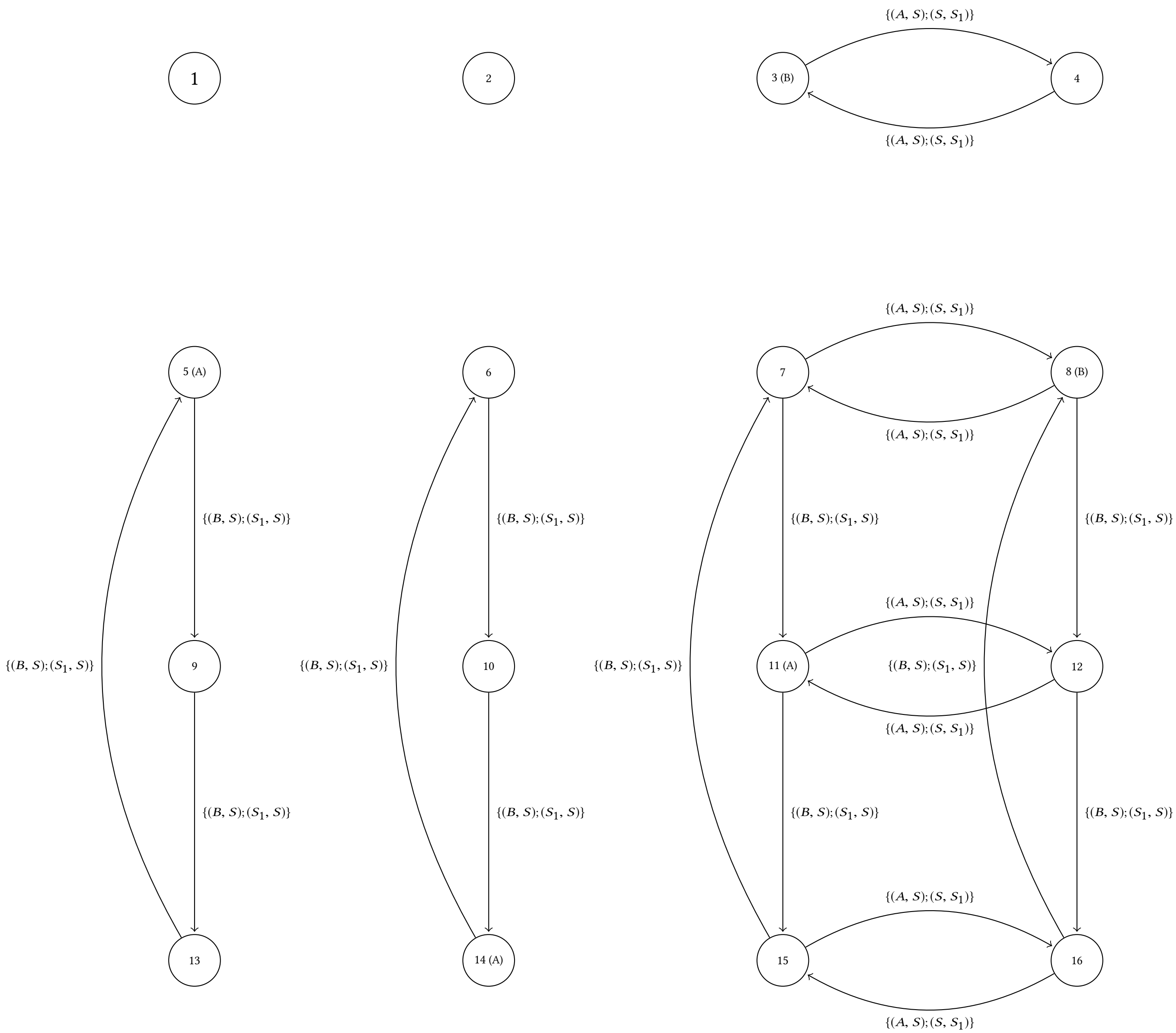


Figure 4: The same generation query (Query 2) in Meerkat



$V^{(n^2)} \otimes H^{(n^2)} + I \otimes H^{(n^2)})$. Now we should check validity of nonterminals. It can be don by multiplication of vector x and M. $x * (V^{(n^2)} \otimes I + V^{(n^2)} \otimes H^{(n^2)} + I \otimes H^{(n^2)}) = x * V^{(n^2)} \otimes I + x * V^{(n^2)} \otimes H^{(n^2)} + x * I \otimes H^{(n^2)}$. It is known that $(B \otimes C) * \mathrm{vec}(X) = \mathrm{Y} \equiv C * X * B^T = Y$. Hence $\mathrm{vec}(X) * (B \otimes C) = \mathrm{Y} \equiv C^T * X^T * B = Y$. As a result, we can compute distance matrix as $I^T * X * V^{(n^2)} + (H^{(n^2)})^T * X * V^{(n^2)} + (H^{(n^2)})^T * X * I$.

$H^2 =$

$$\begin{pmatrix} \{id\} & \varnothing & \varnothing & \varnothing \\ \varnothing & \{id\} & \varnothing & \varnothing \\ \varnothing & \varnothing & \{id; (A, S_1)\} & \{(A, S); (S, S_1)\} \\ \varnothing & \varnothing & \{(A, S); (S, S_1)\} & \{id; (A, S_1)\} \end{pmatrix}$$

$H^4 = H^2$

$(H^2)^T =$

$$\begin{pmatrix} \{id\} & \varnothing & \varnothing & \varnothing \\ \varnothing & \{id\} & \varnothing & \varnothing \\ \varnothing & \varnothing & \{id; (A, S_1)\} & \{(A, S); (S, S_1)\} \\ \varnothing & \varnothing & \{(A, S); (S, S_1)\} & \{id; (A, S_1)\} \end{pmatrix}$$

$V^2 =$

$$\begin{pmatrix} \{id\} & \varnothing & \varnothing & \varnothing \\ \varnothing & \{id\} & \{(B, S); (S_1, S)\} & \varnothing \\ \varnothing & \varnothing & \{id\} & \{(B, S); (S_1, S)\} \\ \varnothing & \{(B, S); (S_1, S)\} & \varnothing & \{id\} \end{pmatrix}$$

$V^4 = V^2$

$X =$

$$\begin{pmatrix} \varnothing & \varnothing & \{(\bot, B)\} & \varnothing \\ \{(\bot, A)\} & \varnothing & \varnothing & \{(\bot, B)\} \\ \varnothing & \varnothing & \{(\bot, A)\} & \varnothing \\ \varnothing & \{(\bot, A)\} & \varnothing & \varnothing \end{pmatrix}$$

$X^T =$

$$\begin{pmatrix} \varnothing & \{(\bot, A)\} & \varnothing & \varnothing \\ \varnothing & \varnothing & \varnothing & \{(\bot, A)\} \\ \{(\bot, B)\} & \varnothing & \{(\bot, A)\} & \varnothing \\ \varnothing & \{(\bot, B)\} & \varnothing & \varnothing \end{pmatrix}$$

$X^T * V^2 =$

$$\begin{pmatrix} \varnothing & \{(\bot, A)\} & \varnothing & \varnothing \\ \varnothing & \varnothing & \varnothing & \{(\bot, A)\} \\ \{(\bot, B)\} & \varnothing & \{(\bot, A)\} & \varnothing \\ \varnothing & \{(\bot, B)\} & \{(\bot, S)\} & \varnothing \end{pmatrix}$$

$(H^2)^T * X^T =$

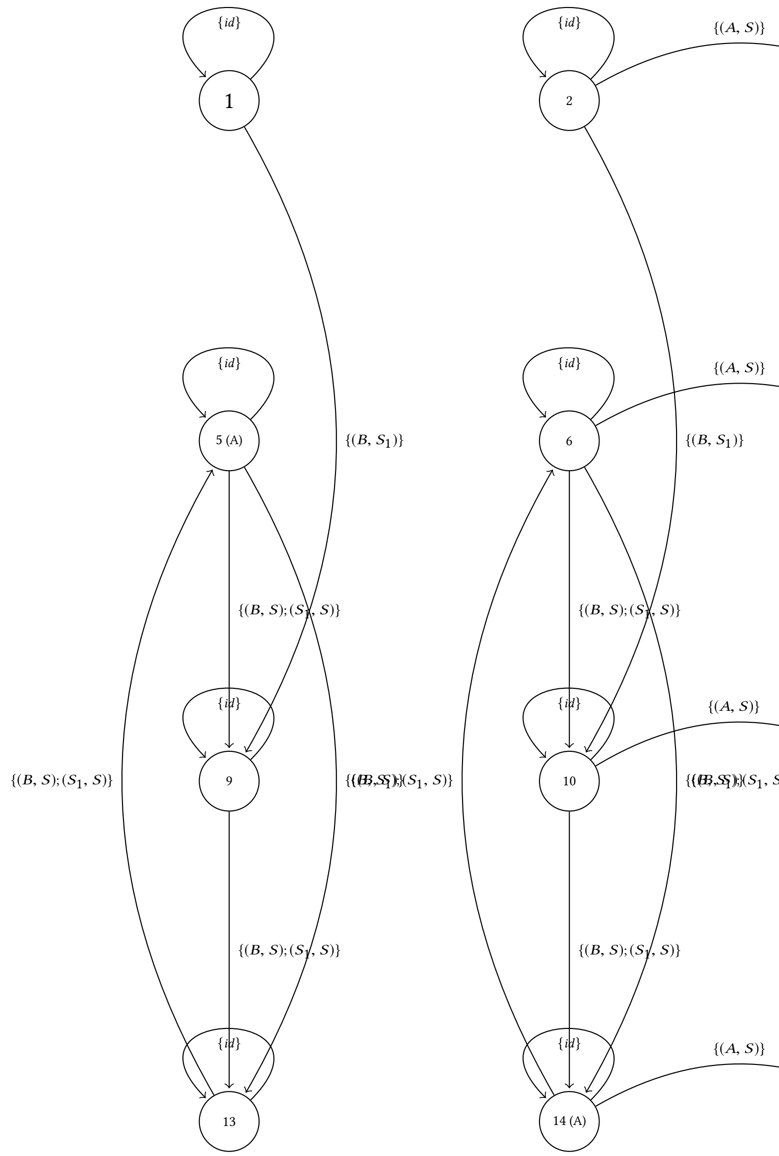
$$\begin{pmatrix} \varnothing & \{(\bot, A)\} & \varnothing & \varnothing \\ \varnothing & \varnothing & \varnothing & \{(\bot, A)\} \\ \{(\bot, B)\} & \varnothing & \{(\bot, A); (\bot, S_1)\} & \varnothing \\ \varnothing & \{(\bot, B)\} & \{(\bot, S)\} & \varnothing \end{pmatrix}$$

$(H^2)^T * X^T * V^2 =$

$$\begin{pmatrix} \varnothing & \{(\bot, A)\} & \varnothing & \varnothing \\ \varnothing & \varnothing & \varnothing & \{(\bot, A)\} \\ \{(\bot, B)\} & \varnothing & \{(\bot, A); (\bot, S_1)\} & \{(\bot, S)\} \\ \varnothing & \{(\bot, B)\} & \{(\bot, S)\} & \varnothing \end{pmatrix}$$

$(X^T * V^2 + (H^2)^T * X^T * V^2 + (H^2)^T * X^T)^T =$

$$\begin{pmatrix} \varnothing & \varnothing & \{(\bot, B)\} & \varnothing \\ \{(\bot, A)\} & \varnothing & \varnothing & \{(\bot, B)\} \\ \varnothing & \varnothing & \{(\bot, A); (\bot, S_1)\} & \{(\bot, S)\} \\ \varnothing & \{(\bot, A)\} & \{(\bot, S)\} & \varnothing \end{pmatrix}$$



$H =$

$$\begin{pmatrix} \{id\} & \varnothing & \varnothing & \varnothing \\ \varnothing & \{id\} & \{(A, S)\} & \varnothing \\ \varnothing & \varnothing & \{id\} & \{(A, S); (S, S_1)\} \\ \varnothing & \varnothing & \{(A, S); (S, S_1)\} & \{id\} \end{pmatrix}$$

$V =$

$$\begin{pmatrix} \{id\} & \varnothing & \{(B, S_1)\} & \varnothing \\ \varnothing & \{id\} & \{(B, S); (S_1, S)\} & \{(B, S_1)\} \\ \varnothing & \varnothing & \{id\} & \{(B, S); (S_1, S)\} \\ \varnothing & \{(B, S); (S_1, S)\} & \varnothing & \{id\} \end{pmatrix}$$

REFERENCES

[1] Krishnendu Chatterjee, Bhavya Choudhary, and Andreas Pavlogiannis. 2017. Optimal Dyck Reachability for Data-dependence and Alias Analysis. *Proc. ACM Program. Lang.* 2, POPL, Article 30 (Dec. 2017), 30 pages. https://doi.org/10.1145/3158118