

Brzozowski's Derivatives for CFPQ

Semyon Grigorev

Saint Petersburg State University

7/9 Universitetskaya nab.

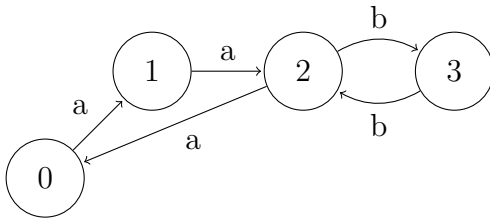
St. Petersburg, 199034, Russia

semen.grigorev@jetbrains.com, rsdpisuy@gmail.com

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 want to use Brzozowski's derivatives for $\text{CFPQ}(\mathcal{G}, G)$. One of problems is a cycles and recursion interaction. Possible solution is based on nonterminal naming: each nonterminal should have a form ${}_uN_v$ where $u \in V$ and $v \in V$. This means that nonterminal with name ${}_uN_v$ corresponds to derivative computed along some (in final grammar — along all) paths from u to v . If there are more than one path from u to v , then there are more than one production for ${}_uN_v$.

Let the input graph is



Let the input grammar is

$$S \rightarrow a S b \mid a b$$

Let we try to compute derivative of this grammar from vertex 2. This vertex in cycle and grammar has a recursion which should “aggressively”

interact with this cycle: cycle produce sequence of arbitrary number of a and any of these sequence may be a prefix of string produced by grammar. In the example below we show that derivative computation may be done in finite number of steps.

1. Edge $2 \xrightarrow{a} 0$.

$$\begin{aligned} {}_2S_0 &\rightarrow S b \mid b \\ S &\rightarrow a S b \mid a b \end{aligned}$$

2. Edge $0 \xrightarrow{a} 1$.

$$\begin{aligned} {}_2S_1 &\rightarrow {}_0S_1 b \\ {}_0S_1 &\rightarrow S b \mid b \\ S &\rightarrow a S b \mid a b \end{aligned}$$

3. Edge $1 \xrightarrow{a} 2$.

$$\begin{aligned} {}_2S_2 &\rightarrow {}_0S_2 b \\ {}_0S_2 &\rightarrow {}_1S_2 b \\ {}_1S_2 &\rightarrow S b \mid b \\ S &\rightarrow a S b \mid a b \end{aligned}$$

4. Edge $2 \xrightarrow{a} 0$.

$$\begin{aligned} {}_2S_0 &\rightarrow {}_0S_0 b \\ {}_0S_0 &\rightarrow {}_1S_0 b \\ {}_1S_0 &\rightarrow {}_2S_0 b \\ {}_2S_0 &\rightarrow S b \mid b \\ S &\rightarrow a S b \mid a b \end{aligned}$$

5. Edge $0 \xrightarrow{a} 1$.

$$\begin{aligned}
{}_2S_1 &\rightarrow {}_0S_1 \ b \\
{}_0S_1 &\rightarrow {}_1S_1 \ b \\
{}_1S_1 &\rightarrow {}_2S_1 \ b \\
{}_0S_1 &\rightarrow S \ b \mid b \\
S &\rightarrow a \ S \ b \mid a \ b
\end{aligned}$$

6. Edge $1 \xrightarrow{a} 2$.

$$\begin{aligned}
{}_2S_2 &\rightarrow {}_0S_2 \ b \\
{}_0S_2 &\rightarrow {}_1S_2 \ b \\
{}_1S_2 &\rightarrow {}_2S_2 \ b \\
{}_1S_2 &\rightarrow S \ b \mid b \\
S &\rightarrow a \ S \ b \mid a \ b
\end{aligned}$$

7. Edge $2 \xrightarrow{a} 0$.

$$\begin{aligned}
{}_2S_0 &\rightarrow {}_0S_0 \ b \\
{}_0S_0 &\rightarrow {}_1S_0 \ b \\
{}_1S_0 &\rightarrow {}_2S_0 \ b \\
{}_2S_0 &\rightarrow S \ b \mid b \\
S &\rightarrow a \ S \ b \mid a \ b
\end{aligned}$$

Given grammar is equal to grammar from step 4, so we can stop computations.

When grammar is in the presented form, then nonterminal with name ${}_uN_v$ is nullable iff there is a path from u to v derivable from N .

Open problems.

- How grammar should be presented for efficient derivatives computation and efficient comparison?
- How can we efficiently handle recomputations? For example in case when we add new production for any previously existing nonterminal. This question is related with data transferring in BSP model: we should propose some mechanism which can avoid unnecessary recomputations in BSP environment.

Seems that main idea from [1] can be reused “as is”. Vertices should exchange messages which contain derivative, check nullability and compute derivatives for outgoing edges if they were not computed in previous steps. It is possible to prove that process is finite: V and N are finite sets, so a total number of new nonterminals is finite. Proof of correctness looks challenging.

References

- [1] Maurizio Nol  and Carlo Sartiani. 2016. Regular Path Queries on Massive Graphs. In Proceedings of the 28th International Conference on Scientific and Statistical Database Management (SSDBM ’16), Peter Baumann, Ioana Manolescu-Goujot, Luca Trani, Yannis Ioannidis, Gergely G bor Barnaf ldi, L szl  Dobos, and Evelin B nyai (Eds.). ACM, New York, NY, USA, Article 13, 12 pages. DOI: <https://doi.org/10.1145/2949689.2949711>