Восходящий анализ. RNGLR

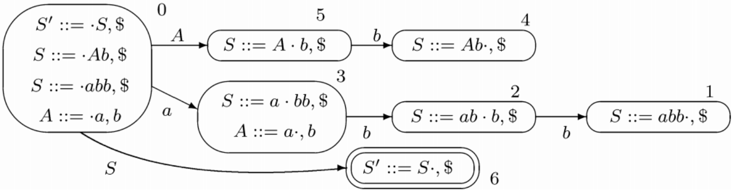
Автор: Екатерина Вербицкая

Санкт-Петербургский государственный университет
Математико-механический факультет

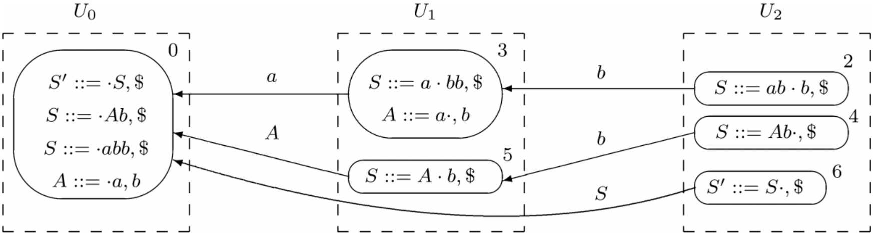
1 декабря 2015г.

GSS

$$S \ ::= \ Ab \mid abb \qquad A \ ::= \ a \qquad (\Gamma_0)$$

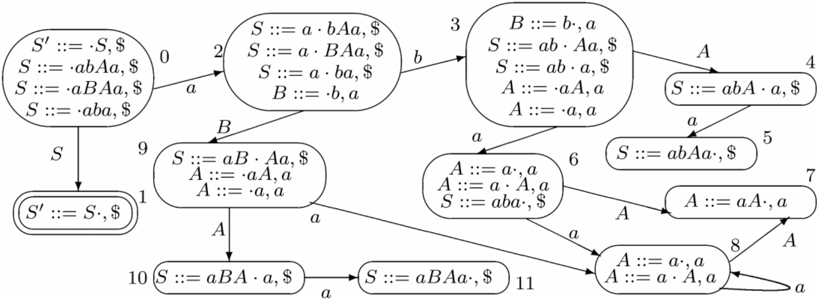


the input ab results in the GSS

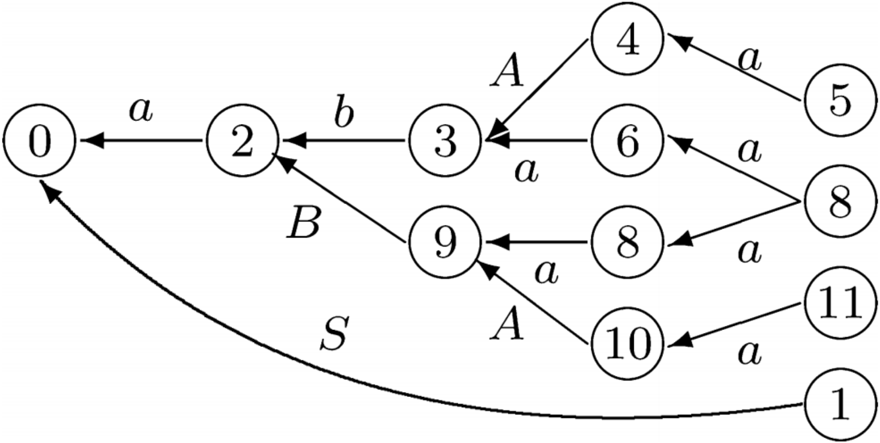


GSS

$$S \ ::= \ abAa \mid aBAa \mid aba \qquad A \ ::= \ a \mid aA \qquad B \ ::= \ b \qquad (\Gamma_1)$$



Грамматика Γ_1



Грамматика Γ_2

$$1.\ S ::= aSA \qquad 2.\ S ::= \epsilon \qquad 3.\ A ::= \epsilon \qquad (\Gamma_2)$$

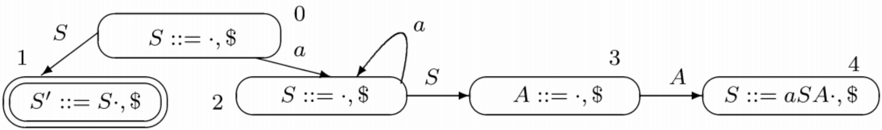
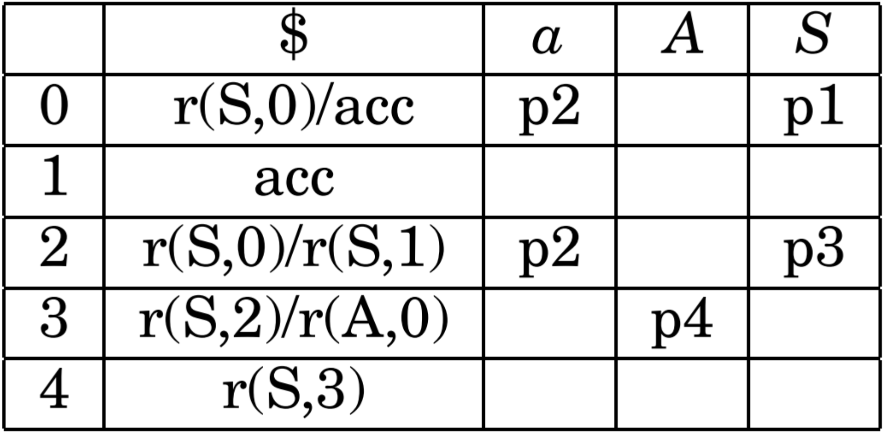
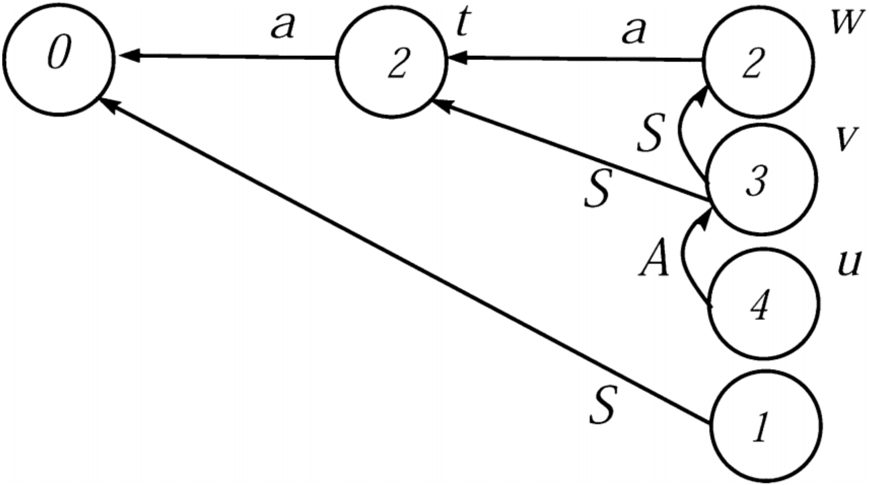


Таблица для Γ_2

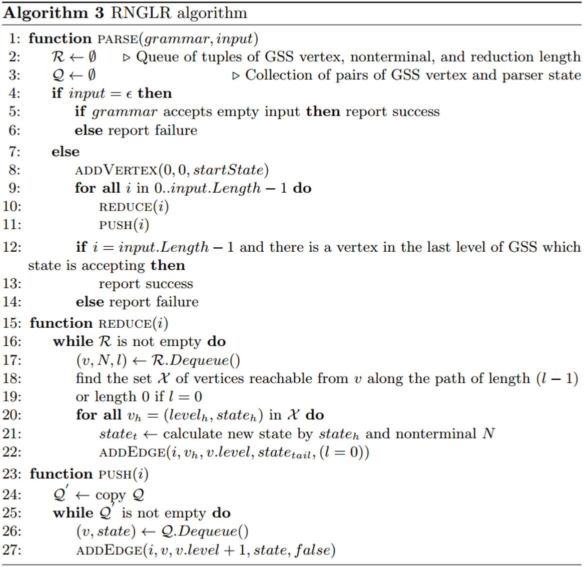


	$	a	A	S
0	r(S,0)/acc	p2		p1
1	acc			
2	r(S,0)/r(S,1)	p2		p3
3	r(S,2)/r(A,0)		p4	
4	r(S,3)			

GSS для *aa*



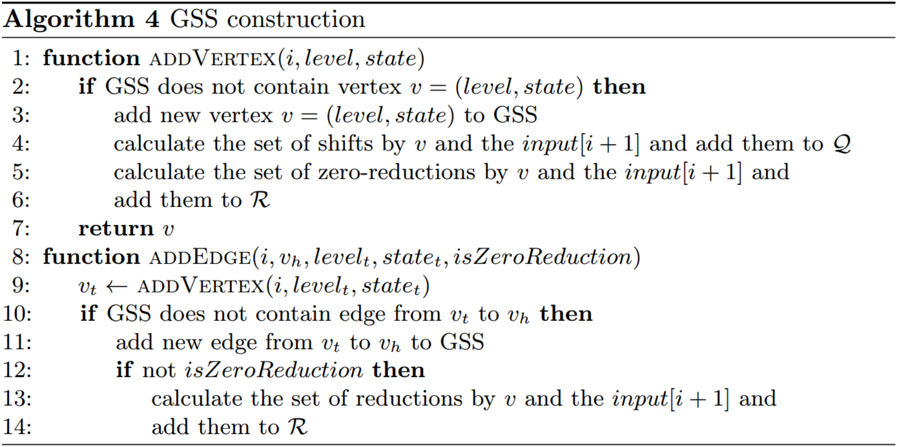
Алгоритм RNGLR



Algorithm 3 RNGLR algorithm

```
function PARSE($grammar, input$)
    $\mathcal{R} \leftarrow \emptyset$        ▷ Queue of tuples of GSS vertex, nonterminal, and reduction length
    $\mathcal{Q} \leftarrow \emptyset$                        ▷ Collection of pairs of GSS vertex and parser state
    if $input = \epsilon$ then
        if $grammar$ accepts empty input then report success
        else report failure
    else
        ADDVERTEX($0, 0, startState$)
        for all $i$ in $0..input.Length - 1$ do
            REDUCE($i$)
            PUSH($i$)
        if $i = input.Length - 1$ and there is a vertex in the last level of GSS which state is accepting then
            report success
        else report failure
function REDUCE($i$)
    while $\mathcal{R}$ is not empty do
        $(v, N, l) \leftarrow \mathcal{R}.Dequeue()$
        find the set $\mathcal{X}$ of vertices reachable from $v$ along the path of length $(l - 1)$
        or length 0 if $l = 0$
        for all $v_h = (level_h, state_h)$ in $\mathcal{X}$ do
            $state_t \leftarrow$ calculate new state by $state_h$ and nonterminal $N$
            ADDEDGE($i, v_h, v.level, state_{tail}, (l = 0)$)
function PUSH($i$)
    $\mathcal{Q}^{'} \leftarrow$ copy $\mathcal{Q}$
    while $\mathcal{Q}^{'}$ is not empty do
        $(v, state) \leftarrow \mathcal{Q}.Dequeue()$
        ADDEDGE($i, v, v.level + 1, state, false$)
```

Построение GSS



Algorithm 4 GSS construction

```
function AddVertex($i, level, state$)
    if GSS does not contain vertex $v = (level, state)$ then
        add new vertex $v = (level, state)$ to GSS
        calculate the set of shifts by $v$ and the $input[i+1]$ and add them to $\mathcal{Q}$
        calculate the set of zero-reductions by $v$ and the $input[i+1]$ and
        add them to $\mathcal{R}$
    return $v$
function AddEdge($i, v_h, level_t, state_t, isZeroReduction$)
    $v_t \leftarrow$ AddVertex($i, level_t, state_t$)
    if GSS does not contain edge from $v_t$ to $v_h$ then
        add new edge from $v_t$ to $v_h$ to GSS
        if not $isZeroReduction$ then
            calculate the set of reductions by $v$ and the $input[i+1]$ and
            add them to $\mathcal{R}$
```