39th International Conference on Current Trends
in Theory and Practice of Computer Science
Špindleruv Mlýn, Czech Republic

# Recursive descent parsing for grammars with contexts

Mikhail Barash

Ph.D. student,
Department of Mathematics and Statistics, University of Turku
Turku Centre for Computer Science (TUCS)
Finland

January 26–31, 2013

# What a recursive descent is?

- subclass of grammars allowing recursive descent: LL($k$)-grammars
  - $k$ look-ahead symbols

# What a recursive descent is?

- subclass of grammars allowing recursive descent: LL($k$)-grammars
  - $k$ look-ahead symbols

- LL(1) context-free grammar generating $\{ a^n b^n \mid n \geqslant 0 \}$:

$$S \to aSb \mid \varepsilon$$

```
S() {                                  |  a() {
    if (look-ahead is "a") {           |      if (current symbol is "a")
        a(); S(); b();                 |          advance position by 1
    }                                  |      else error
}                                      |  }
```

# What a recursive descent is?

- subclass of grammars allowing recursive descent: LL($k$)-grammars
  - $k$ look-ahead symbols

- LL(1) context-free grammar generating $\{ a^n b^n \mid n \geqslant 0 \}$:

$$S \to aSb \mid \varepsilon$$

```
S() {                                  |  a() {
    if (look-ahead is "a") {           |      if (current symbol is "a")
        a(); S(); b();                 |          advance position by 1
    }                                  |      else error
}                                      |  }
```

- First compilers for Pascal are recursive descent parsers.
  - Implemented "by hand" (N. Wirth, 1970).
  - Program code can be easily generated automatically.

# Extension of CFGs with Boolean operations

## Conjunctive grammars (Okhotin, 2001)

$A \to \alpha_1 \,\&\, \ldots \,\&\, \alpha_k, \qquad \alpha_i \in (\Sigma \cup N)^*$

# Extension of CFGs with Boolean operations

## Conjunctive grammars (Okhotin, 2001)

$A \to \alpha_1 \,\&\, \ldots \,\&\, \alpha_k, \qquad \alpha_i \in (\Sigma \cup N)^*$

- "*a string $w$ has property $A \iff$ $w$ has all the properties $\alpha_1, \ldots, \alpha_k$*"

# Extension of CFGs with Boolean operations

## Conjunctive grammars (Okhotin, 2001)

$A \to \alpha_1 \,\&\, \ldots \,\&\, \alpha_k, \qquad \alpha_i \in (\Sigma \cup N)^*$

- "*a string $w$ has property $A \iff$ $w$ has all the properties $\alpha_1, \ldots, \alpha_k$*"

## $\{ a^n b^n c^n \mid n \geqslant 0 \}$

$$
\begin{array}{lcl}
S & \to & AB \,\&\, DC \\
A & \to & aA \mid \varepsilon \\
C & \to & cC \mid \varepsilon
\end{array}
\qquad\qquad
\begin{array}{lcl}
B & \to & bBc \mid \varepsilon \\
D & \to & aDb \mid \varepsilon
\end{array}
$$

# Extension of CFGs with Boolean operations

## Conjunctive grammars (Okhotin, 2001)

$A \to \alpha_1 \,\&\, \ldots \,\&\, \alpha_k, \qquad \alpha_i \in (\Sigma \cup N)^*$

- "*a string $w$ has property $A$ $\Longleftrightarrow$ $w$ has all the properties $\alpha_1, \ldots, \alpha_k$*"

## $\{ a^n b^n c^n \mid n \geqslant 0 \}$

$$
\begin{array}{lcl}
S & \to & AB \,\&\, DC \\
A & \to & aA \mid \varepsilon \\
C & \to & cC \mid \varepsilon
\end{array}
\qquad\qquad
\begin{array}{lcl}
B & \to & bBc \mid \varepsilon \\
D & \to & aDb \mid \varepsilon
\end{array}
$$

- non-context-free languages generated
- complexity of basic parsing algorithms preserved
- nontrivial properties of subclasses

# Grammars with contexts (B, Okhotin, 2012)

$A \to \alpha_1 \,\&\, \ldots \,\&\, \alpha_k$

# Grammars with contexts (B, Okhotin, 2012)

$$A \to \alpha_1 \,\&\, \ldots \,\&\, \alpha_k \,\&\, \rhd\beta_1 \,\&\, \ldots \,\&\, \rhd\beta_m$$

# Grammars with contexts (B, Okhotin, 2012)

$$A \to \alpha_1 \,\&\, \ldots \,\&\, \alpha_k \,\&\, \rhd\beta_1 \,\&\, \ldots \,\&\, \rhd\beta_m \,\&\, \trianglerighteq\gamma_1 \,\&\, \ldots \,\&\, \trianglerighteq\gamma_n$$

# Grammars with contexts (B, Okhotin, 2012)

$$A \to \alpha_1 \,\&\, \ldots \,\&\, \alpha_k \,\&\, \rhd\beta_1 \,\&\, \ldots \,\&\, \rhd\beta_m \,\&\, \trianglerighteq\gamma_1 \,\&\, \ldots \,\&\, \trianglerighteq\gamma_n$$

Given a string $x = u\,w\,v$:

- each $\alpha_i$ defines $w$

# Grammars with contexts (B, Okhotin, 2012)

$$A \to \alpha_1 \,\&\, \ldots \,\&\, \alpha_k \,\&\, \rhd\beta_1 \,\&\, \ldots \,\&\, \rhd\beta_m \,\&\, \trianglerighteq\gamma_1 \,\&\, \ldots \,\&\, \trianglerighteq\gamma_n$$

Given a string $x = u\,w\,v$:

- each $\alpha_i$ defines $w$
- each $\beta_i$ defines $v$ (the right context of $w$)

# Grammars with contexts (B, Okhotin, 2012)

$$A \to \alpha_1 \,\&\, \ldots \,\&\, \alpha_k \,\&\, \rhd\beta_1 \,\&\, \ldots \,\&\, \rhd\beta_m \,\&\, \trianglerighteq\gamma_1 \,\&\, \ldots \,\&\, \trianglerighteq\gamma_n$$

Given a string $x = u\,w\,v$:

- each $\alpha_i$ defines $w$
- each $\beta_i$ defines $v$ (the right context of $w$)
- each $\gamma_i$ defines $wv$ (the extended right context of $w$)

# Grammars with contexts (B, Okhotin, 2012)

$$A \to \alpha_1 \,\&\, \ldots \,\&\, \alpha_k \,\&\, \rhd\beta_1 \,\&\, \ldots \,\&\, \rhd\beta_m \,\&\, \trianglerighteq\gamma_1 \,\&\, \ldots \,\&\, \trianglerighteq\gamma_n$$

Given a string $x = u\,w\,v$:

- each $\alpha_i$ defines $w$
- each $\beta_i$ defines $v$ (the right context of $w$)
- each $\gamma_i$ defines $wv$ (the extended right context of $w$)

**Semantics by *logical deduction* of elementary propositions**

$[X, \langle w \rangle v]$ : "a string $w$ written in right context $v$ has the property $X$"

# Grammars with contexts (B, Okhotin, 2012)

$$A \to \alpha_1 \,\&\, \ldots \,\&\, \alpha_k \,\&\, \rhd\beta_1 \,\&\, \ldots \,\&\, \rhd\beta_m \,\&\, \trianglerighteq\gamma_1 \,\&\, \ldots \,\&\, \trianglerighteq\gamma_n$$

Given a string $x = u\,w\,v$:

- each $\alpha_i$ defines $w$
- each $\beta_i$ defines $v$ (the right context of $w$)
- each $\gamma_i$ defines $wv$ (the extended right context of $w$)

**Semantics by *logical deduction* of elementary propositions**

$[X, \langle w \rangle v]$ : "a string $w$ written in right context $v$ has the property $X$"

**$\{a^n b^n \mathbf{c}\} \cup \{a^n b^{2n} \mathbf{d}\}$**

# Grammars with contexts (B, Okhotin, 2012)

$$A \to \alpha_1 \,\&\, \ldots \,\&\, \alpha_k \,\&\, \rhd\beta_1 \,\&\, \ldots \,\&\, \rhd\beta_m \,\&\, \trianglerighteq\gamma_1 \,\&\, \ldots \,\&\, \trianglerighteq\gamma_n$$

Given a string $x = u\,w\,v$:

- each $\alpha_i$ defines $w$
- each $\beta_i$ defines $v$ (the right context of $w$)
- each $\gamma_i$ defines $wv$ (the extended right context of $w$)

**Semantics by *logical deduction* of elementary propositions**

$[X, \langle w \rangle v]$ : "a string $w$ written in right context $v$ has the property $X$"

**$\{a^n b^n \mathbf{c}\} \cup \{a^n b^{2n} \mathbf{d}\}$**

$$\begin{array}{lcl} S & \to & A\mathbf{c} \mid B\mathbf{d} \\ A & \to & aAb \mid \varepsilon \\ B & \to & aBbb \mid \varepsilon \end{array} \qquad \begin{array}{lcl} L(A) & = & \{a^n b^n\} \\ L(B) & = & \{a^n b^{2n}\} \end{array}$$

# Grammars with contexts (B, Okhotin, 2012)

$$A \to \alpha_1 \,\&\, \ldots \,\&\, \alpha_k \,\&\, \rhd\beta_1 \,\&\, \ldots \,\&\, \rhd\beta_m \,\&\, \trianglerighteq\gamma_1 \,\&\, \ldots \,\&\, \trianglerighteq\gamma_n$$

Given a string $x = u\,w\,v$:

- each $\alpha_i$ defines $w$
- each $\beta_i$ defines $v$ (the right context of $w$)
- each $\gamma_i$ defines $wv$ (the extended right context of $w$)

**Semantics by *logical deduction* of elementary propositions**

$[X, \langle w \rangle v]$ : "a string $w$ written in right context $v$ has the property $X$"

**$\{a^n b^n \mathbf{c}\} \cup \{a^n b^{2n} \mathbf{d}\}$**

$$\begin{array}{lcl} S & \to & A\mathbf{c} \mid B\mathbf{d} \\ A & \to & aAb \mid \varepsilon \\ B & \to & aBbb \mid \varepsilon \end{array} \qquad \begin{array}{lcl} L(A) & = & \{a^n b^n\} \\ L(B) & = & \{a^n b^{2n}\} \end{array}$$

- the language is not standard LL($k$) for any $k$

# Grammars with contexts (B, Okhotin, 2012)

$$A \to \alpha_1 \,\&\, \ldots \,\&\, \alpha_k \,\&\, \rhd\beta_1 \,\&\, \ldots \,\&\, \rhd\beta_m \,\&\, \trianglerighteq\gamma_1 \,\&\, \ldots \,\&\, \trianglerighteq\gamma_n$$

Given a string $x = u\,w\,v$:

- each $\alpha_i$ defines $w$
- each $\beta_i$ defines $v$ (the right context of $w$)
- each $\gamma_i$ defines $wv$ (the extended right context of $w$)

**Semantics by *logical deduction* of elementary propositions**

$[X, \langle w \rangle v]$ : "a string $w$ written in right context $v$ has the property $X$"

**$\{a^n b^n \mathbf{c}\} \cup \{a^n b^{2n} \mathbf{d}\}$**

$$\begin{array}{lcl} S & \to & A\mathbf{c} \mid B\mathbf{d} \\ A & \to & aAb \mid \varepsilon \\ B & \to & aBbb \mid \varepsilon \end{array} \qquad \begin{array}{lcl} L(A) & = & \{a^n b^n\} \\ L(B) & = & \{a^n b^{2n}\} \end{array}$$

- the language is not standard LL($k$) for any $k$
- way out: use right contexts to peek the last symbol of the string

# Grammars with contexts (B, Okhotin, 2012)

$$A \to \alpha_1 \,\&\, \ldots \,\&\, \alpha_k \,\&\, \rhd\beta_1 \,\&\, \ldots \,\&\, \rhd\beta_m \,\&\, \trianglerighteq\gamma_1 \,\&\, \ldots \,\&\, \trianglerighteq\gamma_n$$

Given a string $x = u\,w\,v$:

- each $\alpha_i$ defines $w$
- each $\beta_i$ defines $v$ (the right context of $w$)
- each $\gamma_i$ defines $wv$ (the extended right context of $w$)

**Semantics by *logical deduction* of elementary propositions**

$[X, \langle w \rangle v]$ : "a string $w$ written in right context $v$ has the property $X$"

**$\{a^n b^n \mathbf{c}\} \cup \{a^n b^{2n} \mathbf{d}\}$**

$$\begin{array}{lcl} S & \to & A\mathbf{c} \mid B\mathbf{d} \\ A & \to & aAb \mid \varepsilon \\ B & \to & aBbb \mid \varepsilon \\ X & \to & aX \mid bX \mid \varepsilon \end{array} \qquad \begin{array}{lcl} L(A) & = & \{a^n b^n\} \\ L(B) & = & \{a^n b^{2n}\} \\ L(X) & = & (a \mid b)^* \end{array}$$

- the language is not standard LL($k$) for any $k$
- way out: use right contexts to peek the last symbol of the string

# Grammars with contexts (B, Okhotin, 2012)

$$A \to \alpha_1 \,\&\, \ldots \,\&\, \alpha_k \,\&\, \rhd\beta_1 \,\&\, \ldots \,\&\, \rhd\beta_m \,\&\, \trianglerighteq\gamma_1 \,\&\, \ldots \,\&\, \trianglerighteq\gamma_n$$

Given a string $x = u\,w\,v$:

- each $\alpha_i$ defines $w$
- each $\beta_i$ defines $v$ (the right context of $w$)
- each $\gamma_i$ defines $wv$ (the extended right context of $w$)

**Semantics by *logical deduction* of elementary propositions**

$[X, \langle w \rangle v]$ : "a string $w$ written in right context $v$ has the property $X$"

**$\{a^n b^n \mathbf{c}\} \cup \{a^n b^{2n} \mathbf{d}\}$**

$$\begin{array}{lcl} S & \to & \trianglerighteq Xc \,\&\, A\mathbf{c} \mid \trianglerighteq Xd \,\&\, B\mathbf{d} \\ A & \to & aAb \mid \varepsilon \\ B & \to & aBbb \mid \varepsilon \\ X & \to & aX \mid bX \mid \varepsilon \end{array} \qquad \begin{array}{lcl} L(A) & = & \{a^n b^n\} \\ L(B) & = & \{a^n b^{2n}\} \\ L(X) & = & (a \mid b)^* \end{array}$$

- the language is not standard LL($k$) for any $k$
- way out: use right contexts to peek the last symbol of the string

# Results

Recursive descent parser:

# Results

Recursive descent parser:

- conjunction in rules: scan the substring multiple times

# Results

Recursive descent parser:

- conjunction in rules: scan the substring multiple times
- right contexts ($\triangleright, \trianglerighteqslant$): to choose suitable alternatives of rules

# Results

Recursive descent parser:

- conjunction in rules: scan the substring multiple times
- right contexts ($\triangleright, \trianglerighteqslant$): to choose suitable alternatives of rules
- backtracking: to go over the alternatives

# Results

Recursive descent parser:

- conjunction in rules: scan the substring multiple times
- right contexts ($\triangleright, \trianglerighteqslant$): to choose suitable alternatives of rules
- backtracking: to go over the alternatives
- memoization: linear time instead of exponential

# Results

Recursive descent parser:

- conjunction in rules: scan the substring multiple times
- right contexts ($\triangleright, \trianglerighteqslant$): to choose suitable alternatives of rules
- backtracking: to go over the alternatives
- memoization: linear time instead of exponential

Theory:

- mathematically sound definition of right contexts ("*look-ahead*") within recursive descent

# Results

Recursive descent parser:

- conjunction in rules: scan the substring multiple times
- right contexts ($\triangleright, \trianglerighteqslant$): to choose suitable alternatives of rules
- backtracking: to go over the alternatives
- memoization: linear time instead of exponential

Theory:

- mathematically sound definition of right contexts (*"look-ahead"*) within recursive descent
- higher expressive power, but still linear time

# Results

Recursive descent parser:

- conjunction in rules: scan the substring multiple times
- right contexts ($\triangleright, \trianglerighteqslant$): to choose suitable alternatives of rules
- backtracking: to go over the alternatives
- memoization: linear time instead of exponential

Theory:

- mathematically sound definition of right contexts (*"look-ahead"*) within recursive descent
- higher expressive power, but still linear time
- parsing algorithm proven correct

# Results

Recursive descent parser:

- conjunction in rules: scan the substring multiple times
- right contexts ($\triangleright, \trianglerighteqslant$): to choose suitable alternatives of rules
- backtracking: to go over the alternatives
- memoization: linear time instead of exponential

Theory:

- mathematically sound definition of right contexts (*"look-ahead"*) within recursive descent
- higher expressive power, but still linear time
- parsing algorithm proven correct

Practice:

- implemented as a prototype software

# Results

Recursive descent parser:

- conjunction in rules: scan the substring multiple times
- right contexts ($\triangleright, \trianglerighteqslant$): to choose suitable alternatives of rules
- backtracking: to go over the alternatives
- memoization: linear time instead of exponential

Theory:

- mathematically sound definition of right contexts (*"look-ahead"*) within recursive descent
- higher expressive power, but still linear time
- parsing algorithm proven correct

Practice:

- implemented as a prototype software