SYNTAX ANALYSIS BASED ON MULTIPLE METHODS

Jaromír SMRČEK, Bachelor Degree Programme (3)
Dept. of Information Systems, FIT, VUT
E-mail: xsmrce00@stud.fit.vutbr.cz

Supervised by: Prof. Alexander Meduna

ABSTRACT

We investigate the optimization of syntax analysis using multiple analysis methods. Specifically, we examine the use of predictive and precedential syntax analysis and the connection and switching between two methods. In addition, we discuss the pros and cons of resulting method.

1 ÚVOD

Syntaktická analýza je základním kamenem pro analýzu textů na úrovni programovacích jazyků a následnou tvorbu překladačů či interpretů. Máme několik metod syntaktické analýzy, z nichž každá má svá specifika. Použití jedné metody na základě potřebných vlastností je dlouho používaný přístup při tvorbě překladačů. Avšak v současné době je trendem používat kombinované metody, které využívají těch nejlepších vlastností z několika metod. Takto kombinované metody jsou často nazývané *gramatickými systémy*.

2 SYNTAKTICKÁ ANALÝZA (PARSING)

Jedná se o proces rozpoznávání *struktury* zdrojového jazyka na základě transformace vstupního řetězce (často již rozloženého na jednotlivé lexikální prvky) na datovou strukturu (tzv. syntaktický strom). Pravidla této transformace jsou sestavena pomocí dané *gramatiky* jazyka a program je ukládá do paměti ve formě tabulek.

Proces transformace lze podle jeho směru (viz. dále) dělit na dvě základní skupiny:

2.1 ANALÝZA SHORA DOLŮ

Tato metoda vychází z počátečního stavu (tzv. počátečního neterminálu) a postupnými derivačními kroky se snaží vygenerovat vstupní řetězec. Při vygenerování řetězce

totožného s řetězcem vstupním, považujeme syntaktickou analýzu za úspěšnou a vstupní řetězec spadá do daného jazyka. V opačném případě vstupní řetězec tzv. odmítneme.

Výhodou tohoto přístupu je tvorba poměrně malých tabulek a nepříliš velké zanoření v zásobníku. Jsou výhodné k analýze základní kostry programů, avšak při analýze výrazů je třeba vytvářet *faktorizaci* a odstranit *levou rekurzi*, přičemž se takto upravená gramatika stává složitou.

Do této kategorie řadíme:

- Analýza rekurzivním sestupem,
- prediktivní analýza,
- Packratova analýza.

2.2 ANALÝZA ZDOLA NAHORU

Opačným přístupem je pak analýza zdola nahoru, tedy transformace vstupního řetězce postupnými redukcemi na počáteční neterminál. Pokud pak nelze provést žádnou redukci, která by byla v mezích pravidel, jedná se o chybu a vstupní řetězec nespadá do daného jazyka.

Zde problém s výrazy nenastává, spíše naopak, ale k analýze je zapotřebí poměrně dosti velkých tabulek, což omezuje výkon analyzátoru.

Do této kategorie řadíme:

- LR-analýza (mnoho typů, např. LALR, SLR, . . .),
- CYK analýza,
- precedenční analýza.

2.3 KOMBINOVANÉ METODY

V současnosti velmi diskutovanou a studovanou metodou syntaktické analýzy je spolupráce několika základních typů analýzy. Jde o získání kladů a eliminací záporů jednotlivých metod. Obě části procesu pak mezi sebou musejí komunikovat, aby nedošlo k porušení celistvosti analýzy.

3 ZVOLENÉ ŘEŠENÍ

Rozhodl jsem se kombinovat metodu *prediktivní analýzy* pro kostru programu a *precedenční analýzy* pro výrazy. Hlavními body této kombinace je rozdílnost ve směru (shora dolů – zdola nahoru), struktuře zásobníku, pořadí výstupu a samozřejmě použité tabulce. Oproti analýze jednou metodou je třeba se postarat o *přepnutí metody* a předávání mezivýsledků metodami dosažených.

3.1 PREDIKTIVNÍ ANALÝZA

Jedná se o metodu typu shora dolů. Ke své práci využívá LL-tabulku, která se sestavuje podle dané gramatiky, přičemž je třeba dávat si pozor na levou rekurzi v gramatice obsaženou a případně ji odstranit. Tato rekurze vzniká převážně při tvorbě výrazů, a proto není metoda pro jejich analýzu vhodná. Je však dobře použitelná pro analýzu kostry programovacího jazyka a nevyžaduje příliš rozsáhlou tabulku či složitý analyzátor. Ke své funkci dále potřebuje zásobník, v němž jsou řetězce uloženy pozpátku (typické pro analýzu shora dolů). Výstupem je pak levý rozbor.

3.2 PRECEDENČNÍ ANALÝZA

Tato metoda zdola nahoru, obecně velmi slabá, je velmi kvalitní co se týče analýzy výrazů. Je také jednou z nejlehčích metod z pohledu implementace. Využívá precedenční tabulku, která je sestavena na základě precedence operátorů a jejich asociativitě. Výstupem této metody je tzv. *pravý rozbor*, tedy řetězec uplatněných pravidel, v pořadí opačném, než je zapotřebí je aplikovat na vstupní řetězec.

3.3 SPOJENÍ METOD

Je třeba v analýze vytvořit tzv. *stop-bod*, tedy místo, kde se jedna metoda zastaví, předá informace o současném stavu analýzy metodě druhé a předá jí řízení. Někdy je tento postup interpretován *přepínacím pravidlem*, tedy speciálním pravidlem, které se namísto derivace/redukce stará o přepnutí metody.

3.4 PROGRAMOVÁ ČÁST

Pro výsledný syntaktický analyzátor jsem sestavil gramatiku jazyka, který je vhodný pro demonstraci vlastností analyzátoru. Výstup programu má výkladovou charakteristiku a bude použit jako součást mé bakalářské práce. Základem analyzátoru jsou dvě tabulky pravidel (LL-tabulka a precedenční tabulka), dle kterých se analyzátor řídí. Interpret obsahuje jak funkci o LL-analýzu, tak funkci pro operátorově-precedenční analýzu a tyto funkce se navzájem volají, pokud se vyskytne stop-bod. Tyto stop-body jsou celkem čtyři a to: počátek a konec výrazu, počátek a konec volání funkce ve výrazu. Při startu analýzy se zavolá funkce pro LL-analýzu, která může svou činnost pozastavit a předat řízení funkci pro precedenční analýzu. Výhodou tohoto přístupu je možnost zavolat kdykoli danou funci a zadat jí počáteční neterminál (čehož využívám při analýze funkcí ve výraze).

4 ZÁVĚR

Výsledky ukazují řešení moderní problematiky syntaktické analýzy, potažmo překladačů programovacích jazyků. Práce dokumentuje zajímavé problémy, přínosy a výhody spolupráce dvou poměrně rozdílných metod. Dalším rozšířením projektu je možné zaimplementování optimalizace daného spojení metod popř. porovnání s ostatními metodami (jejich kombinacemi).