

Dispensa 4

4.1 Traduzione guidata dalla sintassi: generalità

Per “traduzione guidata dalla sintassi” si intende il processo di tradurre un input in un certo output durante il parsing. Sostanzialmente, una grammatica context free viene arricchita con una parte semantica in modo da permettere la traslazione di costrutti del linguaggio sorgente mentre se ne fa l’analisi sintattica. I due principali formalismi usati per fare traduzione guidata dalla sintassi sono le *Definizioni Dirette dalla Sintassi* e gli *Schemi di Traslazione*.

4.2 Definizioni Dirette dalla Sintassi

Informalmente, una *Definizione Diretta dalla Sintassi* (da ora in poi *DDS* per semplicità) è una grammatica context free in cui ogni simbolo grammaticale (terminali e non-terminali) viene associato ad attributi, e ogni produzione viene estesa con regole semantiche.

Un attributo può essere un numero, una stringa, una locazione di memoria, etc. o qualsiasi altro tipo di valore in base alla specifica traduzione. La cosa importante da tenere presente è che il valore dell’attributo di un token è fornito dall’analizzatore lessicale, mentre il valore dell’attributo di un non-terminale è calcolato da regola semantica.

Gli attributi definiti dalle regole semantiche possono essere di due tipi: *attributi sintetizzati* ed *attributi ereditati*.

Definizione (attributi sintetizzati):

Il valore di un attributo sintetizzato a un nodo del parse tree è calcolato dai valori degli attributi dei figli di quel nodo nel parse tree

In altre parole, data una produzione $X \rightarrow X_1 \dots X_i \dots X_n$ il valore di un attributo sintetizzato di X dipende dagli attributi di $X_1 \dots X_i \dots X_n$.

Definizione (attributi ereditati):

Il valore di un attributo ereditato a un nodo del parse tree è calcolato dai valori degli attributi ai nodi padre e/o fratelli di quel nodo nel parse tree

In altre parole, data una produzione $X \rightarrow X_1 \dots X_i \dots X_n$ il valore di un attributo ereditato di X_i dipende dagli attributi di X_k (per qualche k) e/o di X .

A questo punto possiamo dare la definizione formale di DDS:

Definizione (DDS):

Una DDS è una grammatica context free in cui ogni produzione $A \rightarrow \alpha$ ha associato un insieme di regole semantiche della forma $b := f(c_1 \dots c_k)$ dove f è una funzione e:

caso1) b è un attributo sintetizzato di A e $c_1 \dots c_k$ sono attributi appartenenti a simboli nella parte destra α della produzione

caso2) b è un attributo ereditato di un simbolo della parte destra α e $c_1 \dots c_k$ sono attributi o di A o di simboli nella parte destra α della produzione

Come semplice esempio di DDS si pensi a una grammatica per espressioni aritmetiche dove i simboli grammaticali sono associati ad attributi (VAL) che ne denotano i valori numerici, e le regole semantiche permettono la valutazione delle espressioni attraverso il computo di operazioni aritmetiche quali la somma, il prodotto, etc. Per esempio, la produzione che specifica il costrutto dell'addizione $E \rightarrow E_1 + E_2$ potrebbe essere estesa con la regola semantica $E.VAL := E_1.VAL + E_2.VAL$. In questo esempio, la funzione f della definizione precedente è la somma $+$, e la dipendenza di b (nell'esempio $E.VAL$) da $c_1 \dots c_k$ (nell'esempio $E_1.VAL$ e $E_2.VAL$) è definita dal caso 1) della definizione essendo $E.VAL$ un attributo sintetizzato.

4.3 Valutazione delle regole semantiche: il grafo delle dipendenze

Quando si usa una DDS per tradurre una stringa sorgente in un certo output bisogna eseguire le regole semantiche durante il parsing dell'input. Dato il parse tree della stringa input l'ordine di valutazione delle regole semantiche deve garantire la seguente proprietà:

Proprietà:

“se un attributo b a un nodo del parse tree dipende da un attributo c (e quindi c deve essere disponibile al momento in cui si calcola b) allora la regola semantica per b deve essere valutata dopo la regola semantica che definisce c ”

Per preservare questa proprietà si può associare a un parse tree un grafo diretto (chiamato *grafo delle dipendenze*) che mantiene traccia delle corrette interdipendenze fra gli attributi ai vari nodi del parse tree. L'algoritmo di costruzione del grafo è il seguente:

Algoritmo di costruzione del grafo delle dipendenze

Input: il parse tree di una stringa sorgente (annotato con le regole semantiche)

Output: il grafo delle dipendenze

Metodo:

FOR ogni nodo n del parse tree **DO**

FOR ogni attributo a del simbolo grammaticale al nodo n **DO**
 costruisci un nodo a nel grafo

FOR ogni nodo n del parse tree **DO**

FOR ogni regola semantica $b := f(c_1 \dots c_k)$ associata alla produzione usata al nodo n **DO**

FOR $i=1$ to k **DO**

 costruisci un arco dal nodo c_i al nodo b

Nell'algoritmo il primo blocco di FOR inserisce i nodi nel grafo (uno per ogni attributo dei vari simboli grammaticali nel parse tree), il secondo blocco di FOR inserisce gli archi in modo tale che la direzione di un arco mantiene traccia della dipendenza fra gli attributi.

Ora, per garantire la proprietà descritta sopra basta prendere un qualsiasi ordinamento topologico del grafo delle dipendenze per ottenere un ordine di valutazione corretto delle regole semantiche, dove un ordinamento topologico è definito come segue:

Definizione (Ordinamento Topologico):

Un ordinamento topologico di un grafo diretto aciclico è un qualsiasi ordinamento $m_1 \dots m_n$ dei nodi del grafo tale che: “se $m_i \rightarrow m_j$ è un arco del grafo allora m_i appare prima di m_j nell'ordinamento”

Ricapitolando, la linea generale del metodo per valutare le regole semantiche di una DDS e fare quindi traduzione guidata dalla sintassi di una stringa input è la seguente:

stringa \Rightarrow parse tree \Rightarrow grafo delle dipendenze \Rightarrow ordinamento topologico \Rightarrow output

Si noti però che poiché la definizione di ordinamento topologico si applica solo a grafi diretti aciclici, questo metodo è applicabile soltanto nel caso in cui il grafo delle dipendenze non contiene cicli. In caso contrario la traduzione guidata dalla sintassi fallisce e la DDS si dice *circolare*.

4.4 Definizioni con S-attributi e Definizioni con L-attributi

Esistono due casi particolari di DDS per le quali la valutazione delle regole semantiche è più semplice e non fa uso del metodo generale del grafo delle dipendenze, queste DDS si chiamano *Definizioni con S-attributi* e *Definizioni con L-attributi*, e sono di seguito brevemente illustrate.

Definizione (Definizione con S-attributi):

Una Definizione con S-attributi è una DDS che usa solo attributi sintetizzati

La valutazione delle regole semantiche nelle Definizioni con S-attributi è molto semplice e immediata, basta infatti valutare gli attributi sul parse tree in modo bottom-up man mano che l'input viene parsato. Infatti, poiché esistono solo attributi sintetizzati (e quindi le dipendenze sono sempre del tipo “attributo di nodo padre dipende da attributo di nodo figlio”), la valutazione fatta in stile bottom-up garantisce che gli attributi di cui ha bisogno il calcolo di una regola semantica sono sempre disponibili, e quindi la Proprietà illustrata in 4.3 è preservata.

Definizione (Definizione con L-attributi):

Una Definizione con L-attributi è una DDS con la seguente restrizione sugli attributi ereditati:

in ogni produzione $A \rightarrow X_1 \dots X_i \dots X_n$ il simbolo X_i può avere attributi ereditati solo da $A, X_1 \dots X_{i-1}$

In altre parole, una Definizione con L-attributi ha attributi sintetizzati normali, mentre ha un vincolo sugli attributi ereditati nel senso che questi possono dipendere solo dal padre e/o fratelli alla sinistra.

Il metodo di valutazione delle regole semantiche di una Definizione con L-attributi è dato dal seguente algoritmo ricorsivo, applicato ai nodi n del parse tree di una stringa input e innescato sulla radice:

```
df_visit(n)
BEGIN
  FOR ogni figlio  $m$  di  $n$ , da sinistra a destra, DO
    BEGIN
      valuta gli attributi ereditati di  $m$ 
      df_visit(m)
    END
  valuta gli attributi sintetizzati di  $n$ 
END
```

Questo algoritmo tocca i nodi del parse tree in stile depth-first e impone il seguente ordine di valutazione: “appena tocca un nodo ne calcola gli attributi ereditati, i sintetizzati invece li calcola quando, a fine ricorsione, risale su quel nodo”

Grazie al sottostante stile depth-first e alla restrizione sugli attributi ereditati che non possono dipendere da quelli dei fratelli alla destra, è ancora una volta garantita la proprietà che gli attributi di cui ha bisogno il calcolo di una regola semantica siano sempre disponibili.

4.5 Schemi di traslazione: cenni

Per concludere, diamo dei cenni all'altro formalismo per fare traduzione guidata dalla sintassi, gli Schemi di Traslazione.

Definizione (Schema di Traslazione):

Uno Schema di Traslazione è una grammatica context-free in cui i simboli grammaticali (terminali e non-terminali) sono associati ad attributi, e azioni semantiche, racchiuse tra graffe { }, sono inserite nella parte destra delle produzioni

Questa definizione può apparentemente sembrare equivalente a quella di DDS (al di là dell'uso differente di terminologia tra "regole semantiche" ed "azioni semantiche"), tuttavia c'è una differenza sostanziale fra DDS e Schemi di Traslazione. Nelle DDS, infatti, l'associazione fra produzione e regola semantica è logica, ossia va concepita come un 'annotamento'; viceversa, negli Schemi di Traslazione l'associazione fra produzione e azione semantica è fisica, nel senso che le azioni semantiche sono esplicitamente inserite nella parte destra, ed è quindi significativa la loro particolare posizione fisica.

In altre parole, gli Schemi di Traslazione sono un formalismo un pò più a "basso livello" (cioè meno astratto/più implementativo) rispetto alle DDS in quanto l'ordine di valutazione delle azioni semantiche è esplicitamente mostrato dalla loro posizione nella parte destra della produzione. Mentre quest'ordine non era esplicitamente visibile nelle DDS (e quindi si doveva ricorrere alla metodologia del grafo delle dipendenze per stabilire un corretto ordine di valutazione), il fatto che sia esplicito negli Schemi di Traslazione permette di stabilire immediatamente un ordine di valutazione corretto, che è quello depth-first e può essere così formalizzato:

"negli Schemi di Traslazione un output per una stringa input è generato eseguendo le azioni semantiche nell'ordine in cui appaiono in una visita depth-first del parse tree di quella stringa"