5N-3

Deterministic Parsing of Simple Syntax-Directed Translateors

Hiroyuki Anzai, Tomonari Doi and Hidehiko Eguchi

(Kyusyu Institute of Technology)

The director sets are used for the top-down type of parsers to make the behavior deterministic. This paper presents a method to compute the similar kinds of director sets for a recursive-descent syntax-directed translator[1].

1. Preparation

An input symbol set is denoted by T, an output symbol set by Δ, a nonterminal symbol set by N, and $T \cup \Delta \cup N$ by V. A set of all nonterminals generating the empty string ε is denoted by N'. For a given set S, the power set of S is written by PS. The set $\{\varepsilon\}$ is written by λ and the empty set by ϕ. For sets x, $y \in S$, x + y and x·y(usually written by xy) are interpreted as the set union and the set concatination, respectively.

For a given set S, a matrix $A = (a_{ij}) = ([A]_{ij})$, $a_{ij} \in PS$, is called an S-matrix. The λ-matrix of which each diagonal element is λ and the others are all ϕ is called the identity matrix and written by E. The sum and the product of matrices are defined as the same as the case in usual algebra. Vectors are all written by Gothic letters.

A function ∂: $V \times PV \rightarrow P\lambda$ is defined as $\partial_s S = \lambda$ if $s \in S$ and $\partial_s S = \phi$ if otherwise. Then, the definition is extended for a V-matrix A = (aij) as $\partial_s A = (\partial_s a_{ij})$. For a V-matrix A, $C = \sum_{s \in V} \partial_s A$ and $C' = \sum_{s \in N' \cup \Delta} \partial_s A$ are called the adjacent and the ε-adjacent matrix of A, respectively.

2. Translation Scheme and Transition Matrix

One of the methods to define simple syntax-directed translation is to regard a terminal symbol set of extended BNF as a union of T and Δ. Each word w defined by this grammar is a string generated from $T \cup \Delta$. For the string w, replace every output symbols by ε, then we have an input string w_i, and similarly an outout string w_0 associated with w_i. Thus it can be said that w shows the translation $t(w_i) = w_0$. We call this translation scheme "regular translation form" or RTF in short. The following simple example defines the translation from a simple arithmetic expression to the postfix notation.

<E> = <T> ('+' <T> [+])* [;]

<T> = ('a'[a] + '(' <E> ')') [;]

Where N = { <E>, <T> }, T = {'+', 'a', '(', ')'} and Δ = {[+], [a], [;]}. The output symbol [;] (written usually without brackets) have special role as a control symbol(for pop-up operation), outputs ε and must be always placed at the right end of each RTF equation.

For each nonterminal X, the right part of the X-defining RTF equation is a regular expression R_X generared from V, and is able to be represented as a finite automaton $\mathfrak{M}_X$ or an $n_X \times n_X$ transition matrix such that $R_X = [A^*]_{1n_X}$, where n_X is the number of states of $\mathfrak{M}_X$, as follows:

$$A_E = \begin{vmatrix} \phi & \langle T\rangle & \phi & \phi & \phi \\ \phi & \phi & '+' & \phi & \phi \\ \phi & \phi & \phi & \langle T\rangle & \phi \\ \phi & [+] & \phi & \phi & [;] \\ \phi & \phi & \phi & \phi & \phi \end{vmatrix} \qquad A_T = \begin{vmatrix} \phi & 'a' & \phi & '(' & \phi & \phi \\ \phi & \phi & [a] & \phi & \phi & \phi \\ \phi & \phi & \phi & \phi & \phi & [;] \\ \phi & \phi & \phi & \phi & \langle E\rangle & \phi \\ \phi & \phi & ')' & \phi & \phi & \phi \\ \phi & \phi & \phi & \phi & \phi & \phi \end{vmatrix}$$

For $\mathfrak{M}_X = (V, \chi, x_1, x_{n_X}, \tau_X)$, where V: a set of transition symbols, χ: a set of states $\{x_1, \cdots, x_{n_X}\}$, x_1: the initial state, x_{n_X}: the final state, τ_X: the transition function: $\chi \times V \rightarrow \chi$, $\tau_X(x_i, s) = x_j$ iff $s \in [A_X]_{ij}$ iff $[\partial_s A_X]_{ij} = \lambda$.

For each nonterminal X, $\mathfrak{M}_X$ thus constructed is regarded as a kind of sequential transducer. A special machine called the monitor $\mathfrak{M}_M$ such that <M> = <X₀> '#' is added to them, where X_0 is the starting nonterminal and # is the end marker. The whole of them are linked together by means of the recursive call mechanism as shown below and works as a syntax-directed translator. This system called the SDT $\mathfrak{M}$ starts from $\mathfrak{M}_M$. Let the currently active machine be $\mathfrak{M}_X$, the current state be x_{X_i} and the look-ahead symbol be t. Then, for the state transition $\tau_X(x_{X_i}, s) = x_{X_j}$, $\mathfrak{M}_X$ performs one of the followings.

(1) When $s = t' \in T$ and $t = t'$, $\mathfrak{M}_X$ transits to x_{X_j} and inputs a new look-ahead symbol as t.

(2) When $s = \delta (\neq [;]) \in \Delta$, $\mathfrak{M}_X$ outputs δ and transits to x_{X_j}.

(3) When $s = Y \in N$, $\mathfrak{M}_X$ calls $\mathfrak{M}_Y$, i.e. the current state becomes x_{Y_1}, the initial state of $\mathfrak{M}_Y$, and $\mathfrak{M}_X$ pauses.

(4) When s = [;], $\mathfrak{M}_X$ returns control to the machine(say $\mathfrak{M}_Z$) which has called $\mathfrak{M}_X$. Let the state transition where $\mathfrak{M}_Z$ has called $\mathfrak{M}_X$ be $\tau_Z(x_{Z_i}, X) = x_{Z_j}$, then the next state becomes x_{Z_j}.

3. First sets

We give a function $\gamma: N \times V \to P\lambda$:

$$\gamma_{Xs} = [C'_X{}^* \partial_s A_X C_X{}^*]_{1 n_X},$$

which means that if there exits a string $\xi s w$ in $[A_X{}^*]_{1 n_X}$ such that $\xi \in N'^*$ and $w \in V^*$, then $\gamma_{Xs} = \lambda$, otherwise $\gamma_{Xs} = \phi$.

For each $X \in N$, First set of X is defined as

$$\mathrm{First}(X) = \sum_{Y \in N} \gamma_{XY} \mathrm{First}(Y) + \sum_{t \in T} \gamma_{Xt} \{t\}$$

In order to solve the equation, put

$u = (u_X)$, $u_X = \mathrm{First}(X)$

$d = (d_X)$, $d_X = \sum_{t \in T} \gamma_{Xt}\{t\}$

$\Gamma = (\gamma_{XY})$.

Then, we have the solution, as follows:

$$u = \Gamma u + d = \Gamma^* d.$$

The definition of the first set is extended as

First(t) = {t} for $t \in T$,

First(δ) = ϕ for $\delta \in \Delta$.

4. Follow Sets

A function $d: V \times V \times N \to P\lambda$ is given as

$$d_{ss'Y} = [C_Y{}^* \partial_s A_Y C'_Y{}^* \partial_{s'} A_Y C_Y{}^*]_{1 n_Y},$$

which means that if there exists a string $w s \xi s' w'$ in $[A_Y{}^*]_{1 n_Y}$ such that $w, w' \in V^*$ and $\xi \in N'^*$, then $d_{ss'Y} = \lambda$, otherwise $d_{ss'Y} = \phi$.

And a function $\theta: V \times N \to P\lambda$ is defined as

$$\theta_{sY} = [C_Y{}^* \partial_s A_Y C'_Y{}^*]_{1 n_Y},$$

which means that if there exits a string $w s \xi$ in $[A_X{}^*]_{1 n_Y}$ such that $w \in V^*$ and $\xi \in N'^*$, then $\theta_{sY} = \lambda$, otherwise $\theta_{sY} = \phi$.

For each $X \in N$, Follow set of X is defined as

$$\mathrm{Follow}(X) = \sum_{Y \in N} \theta_{XY} \mathrm{Follow}(Y) + \sum_{Y \in N} \sum_{s \in V} d_{XsY} \mathrm{First}(s)$$

In order to solve the above equation, put

$u = (u_X)$, $u_X = \mathrm{Follow}(X)$

$d = (d_X)$, $d_X = \sum_{Y \in N} \sum_{s \in V} d_{XsY} \mathrm{First}(s)$

$\Theta = (\theta_{XY})$.

Then, we have the solution, as follows:

$$u = \Theta u + d = \Theta^* d$$

For $\delta \in \Delta$, the definition is extended as

$$\mathrm{Follow}(\delta) = \sum_{Y \in N} \theta_{\delta Y} \mathrm{Follow}(Y) + \sum_{Y \in N} \sum_{s \in V} d_{\delta s Y} \mathrm{First}(s)$$

5. Director Sets

For $s \in V$, a set of terminals used for deterministic parsing is defined as follows:

$$\mathrm{Director}(s) = \mathrm{First}(s) \cup \rho(s) \mathrm{Follow}(s)$$

where $\rho(s)$ is λ if $s \in N' \cup \Delta$, otherwise ϕ.

For each machine $\mathfrak{M}_X$ and for each state x_{X_i} in $\mathfrak{M}_X$, if there are no two state transitions $\tau_X(x_{X_i}, s)$ and $\tau_X(x_{X_i}, s')$ such that $\mathrm{Director}(s) \cap \mathrm{Director}(s') \neq \phi$, the parser of the SDT $\mathfrak{M}$ is called LL(1). In this case, we can make the move of the SDT $\mathfrak{M}$ deterministic in such a manner that for a look-ahead input symbol t at state x_{X_i} in $\mathfrak{M}_X$, we make $\mathfrak{M}$ perform the state transition $\tau_X(x_{X_i}, s)$ if $t \in \mathrm{Director}(s)$, and if there is no such transition and furthermore if the state x_{X_i} is the final state $x_{X_{n_X}}$ of $\mathfrak{M}_X$, then we make $\mathfrak{M}$ perform the return operation.

6. Remarks

Similarly, we can easily obtain the more detailed director set associated with not only a symbol but the state where the set is used.

[1] H. Anzai: A theory of recursive descent translator generator, Proc. Int'l Comp. Symp., 1980(Vol.II), pp.1171-1182.