Definitions

A “handle” is the right side of the most recently applied production in a right derivation of a right sentential form.

An “item” (or an “LR(0) item”) is a production with a mark on the right hand side. (I use the symbol “ˆ” for the mark.)

A “viable prefix” is a prefix of a right-sentential form which ends no further right than the end of the handle.

A “complete item” is an item in which the mark is at the far right hand side of the item.

The set of viable prefixes for a context free language form a regular set; hence, there is a finite state machine that recognizes them. The transition symbols are language symbols (terminal and non-terminal); the states are identified by items or sets of items. There are two ways to produce it:

(I assume that the start symbol appears on the left of only one production; to make that happen, we could introduce a new start symbol S’ and a production S’ → S, where S is the old start symbol.)

non-deterministic: (the states are items)

1. The start state is the state S’ → S.

2. For any state A → α ˆ B β, B a non-terminal, α and β strings of terminals and non-terminals (possibly empty), we have a transition on ϵ to all states B → γ, γ a string of language symbols.

3. For any state A → α ˆ Γ β, α and β strings of terminals and non-terminals, Γ a single symbol (terminal or non-terminal), we have a transition on Γ to the state A → α Γ ˆ β.

Then we can transform into the deterministic FSM.

deterministic (directly): (the states are sets of items)

1. Start Rule The start state contains the item S’ → S.

2. Competition Rule If a state contains an item A → α ˆ B β, B a non-terminal, α and β strings of terminals and non-terminals (possibly empty), then the state also contains all items of the form B → γ, γ a string of language symbols.

3. Read Rule If a state contains a subset of items A_i → α_i ˆ Γ β_i, Γ a language symbol (the same for all such states), α_i and β_i strings of terminals and non-terminals (possibly empty), then there is a transition of Γ to a state containing all A_i → α_i Γ ˆ β_i (the mark moves one position).

If the deterministic machine built as above for the language from its grammar has the property that any set containing a complete item has only that item (i.e., is a singleton set), then the grammar is said to be an LR(0) grammar.