Automi a Pila e Grammatiche Libere dal Contesto

Automi a Pila

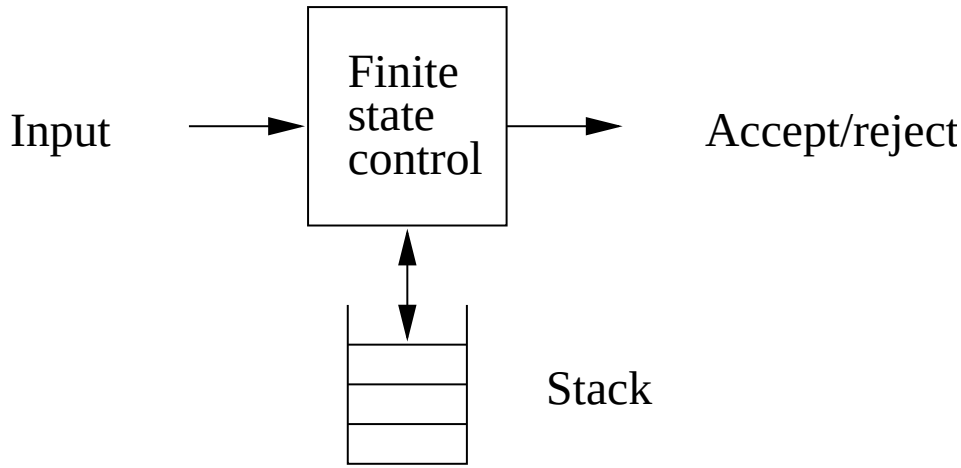
- Un **automa a pila (PDA)** e' una estensione degli automi a stati finiti, che ha una memoria (una pila)
- Vedremo due modi equivalenti per definire il linguaggio accettato da un PDA
- Vedremo che la classe dei linguaggi accettati da PDA (nondeterministici) e' esattamente quella dei CFL

Determismo e Non-Determismo

- Vedremo come nel caso dei PDA la versione deterministica e non deterministica non siano equivalenti
- Gli automi a pila deterministici operano in modo simile ai parser dei compilatori, e' importante capire quali costrutti linguistici non possono essere catturati.

Descrizione Intuitiva

Un automa a pila (PDA) e' in pratica un ϵ-NFA con una pila, la pila contiene simboli ordinati LIFO



Mosse dell'automa

- Il PDA in un certo stato puo' (al solito) leggere un simbolo di input o eseguire una ϵ-transizione
- Leggere il *simbolo memorizzato al top della pila*
- Quando si muove
  1. puo' cambiare stato
  2. *rimpiazzare il top della pila con una stringa* (la pila rimane invariata, o viene eliminato il top della pila, o viene aggiunta una stringa in cima alla pila)

Esempio

Consideriamo il linguaggio CFL

$$L_{wwr} = \{ww^R : w \in \{0,1\}^*\},$$

con "grammatica"

$$P \rightarrow 0P01P1|\epsilon$$

Possiamo definire un PDA che utilizza la pila nel seguente modo:

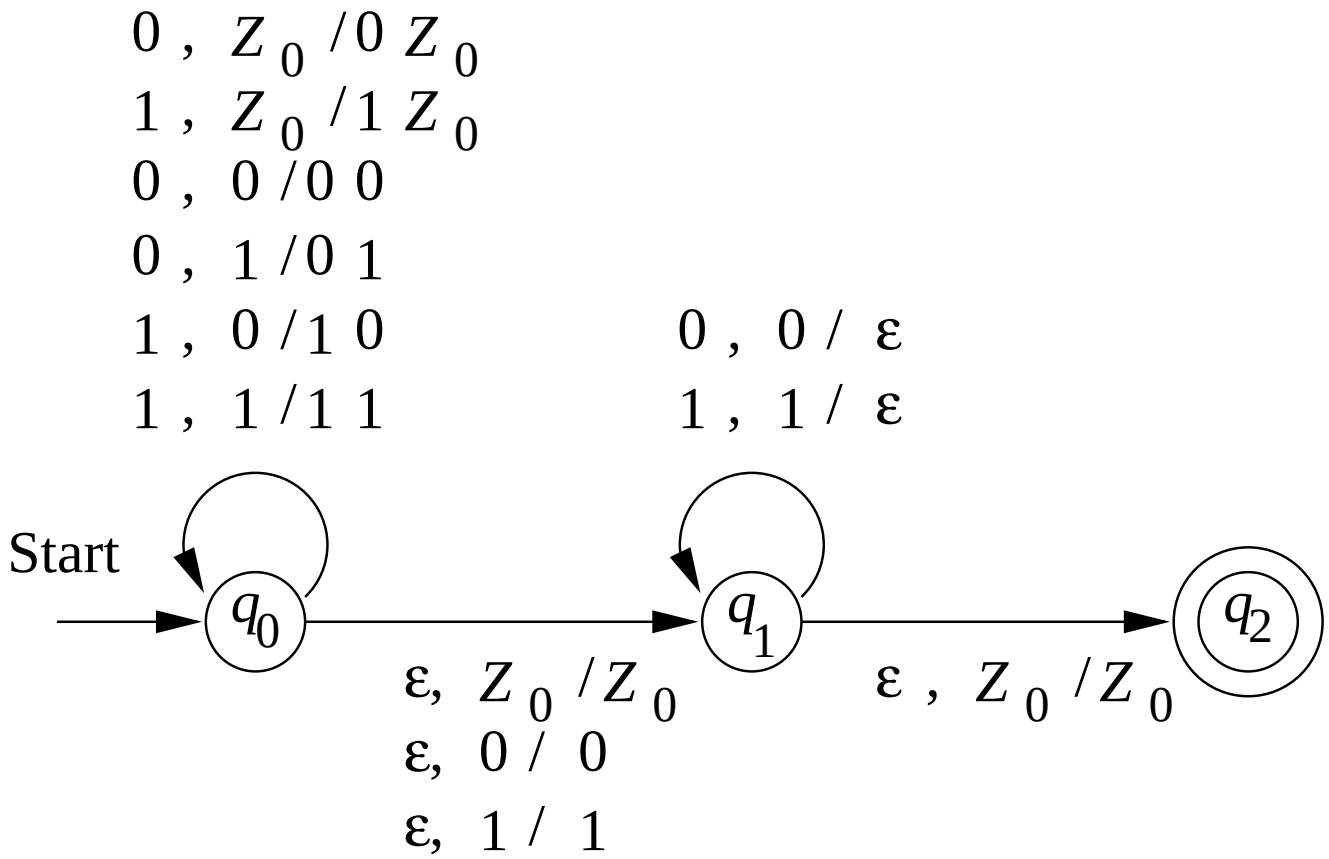
1. fino a che legge i simboli di w memorizza i simboli letti nella pila
2. quando inizia w^R, svuota la pila, se il simbolo memorizzato al top e' uguale a quello in input

PDA: Descrizione Intuitiva

Un PDA per L_{wwr} ha tre stati, e funziona come segue:

- 0: Scommette che sta leggendo w. Rimane nello stato 0, e mette il simbolo di input sulla pila.
- 0: Scommette che sta nel mezzo di ww^R. Va spontaneamente nello stato 1.
- 1: Sta leggendo la testa di w^R. La paragona al top della pila. Se sono uguali, fa un pop della pila, e rimane nello stato 1. Se non sono uguali, si ferma.
- 1: Se la pila e' vuota, va nello stato 2 e accetta la stringa.

PDA: Diagramma di Transizione



Commenti

- Gli archi del diagramma sono etichettati con $a, Z/\alpha$ che rappresentano:
  1. il simbolo in input (o ϵ)
  2. Z il simbolo al top della pila
  3. α la stringa che viene messa al top della pila (al posto di Z)
- Z_0 e' il simbolo memorizzato inizialmente nella pila

Commenti

- in q_0 qualsiasi simbolo legga in input lo pusha al top della pila
- in q_0 si puo' muovere in q_1 senza leggere input e lasciando la pila invariata
- in q_1 cancella (rimpiazzandolo con ϵ) il simbolo al top della pila se combacia con l'input
- in q_1 si muove in q_2 se la pila e' vuota (contiene Z_0)

Definizione Formale di PDA

Un PDA e' una tupla di 7 elementi:

$$P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F),$$

dove

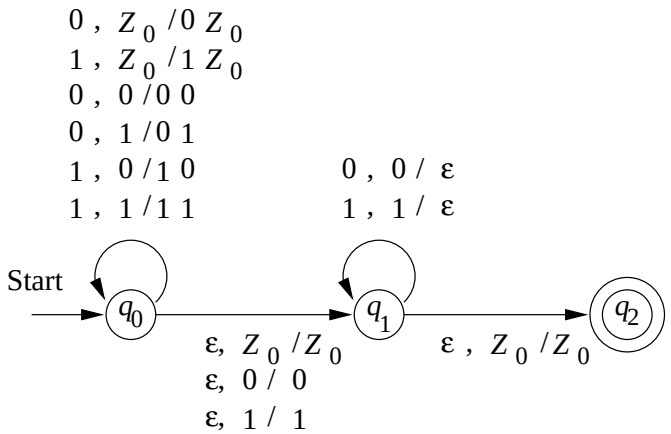
- Q e' un insieme finito di stati,
- Σ e' un *alfabeto finito di input*,
- Γ e' un *alfabeto finito di pila*,
- $\delta : Q \times \Sigma \cup \{\epsilon\} \times \Gamma \longrightarrow 2^{Q \times \Gamma^*}$ e' la *funzione di transizione*,
- q_0 e' lo *stato iniziale*,
- $Z_0 \in \Gamma$ e' il *simbolo iniziale* per la pila, e
- $F \subseteq Q$ e' l'insieme di *stati di accettazione*.

Funzione di Transizione

- la funzione di transizione $\delta : Q \times \Sigma \cup \{\epsilon\} \times \Gamma \longrightarrow 2^{Q \times \Gamma^*}$ restituisce un insieme di coppie (q, α) dove q e' uno stato e α e' la stringa nell'alfabeto della pila che deve rimpiazzare il simbolo al top
- l'automa e' nondeterministico, esistono ovviamente varie scelte

Esempio

Il PDA



e' la 7-tupla

$$P = (\{q_0, q_1, q_2\}, \{0, 1\}, \{0, 1, Z_0\}, \delta, q_0, Z_0, \{q_2\}),$$

dove δ e' data dalla tavola seguente:

La Funzione di Transizione

	$0, Z_0$	$1, Z_0$	$0,0$	$0,1$	$1,0$	$1,1$	ϵ, Z_0	$\epsilon, 0$	$\epsilon, 1$
$\rightarrow q_0$	$q_0, 0Z_0$	$q_0, 1Z_0$	$q_0, 00$	$q_0, 01$	$q_0, 10$	$q_0, 11$	q_1, Z_0	$q_1, 0$	$q_1, 1$
q_1			q_1, ϵ			q_1, ϵ	q_2, Z_0		
$\star q_2$									

Descrizioni Istantanee

- Per descrivere le computazioni di un PDA dobbiamo descrivere la configurazione dell'automa ad un certo punto Usiamo delle *descrizioni istantanee* (ID) del PDA.
- Una ID e' una tripla

$$(q, w, \gamma)$$

dove $q \in Q$ e' lo stato, $w \in \Sigma^*$ l'input rimanente, e $\gamma \in \Gamma^*$ e' una stringa, il contenuto della pila.

- Nota: la pila e' rappresentata da una stringa $\gamma = a\alpha$ a e' il top

Computazione

- Per descrivere le computazioni di un PDA introduciamo una nozione di $\vdash$ tra configurazioni istantanee,

$$ID_1 \vdash ID_2$$

l'automa nella configurazione ID_1 si muove in un passo nella configurazione ID_2

- Al solito usiamo $\overset{*}{\vdash}$ la chiusura riflessiva e transitiva di $\vdash$.

Definizione: Passo di Computazione

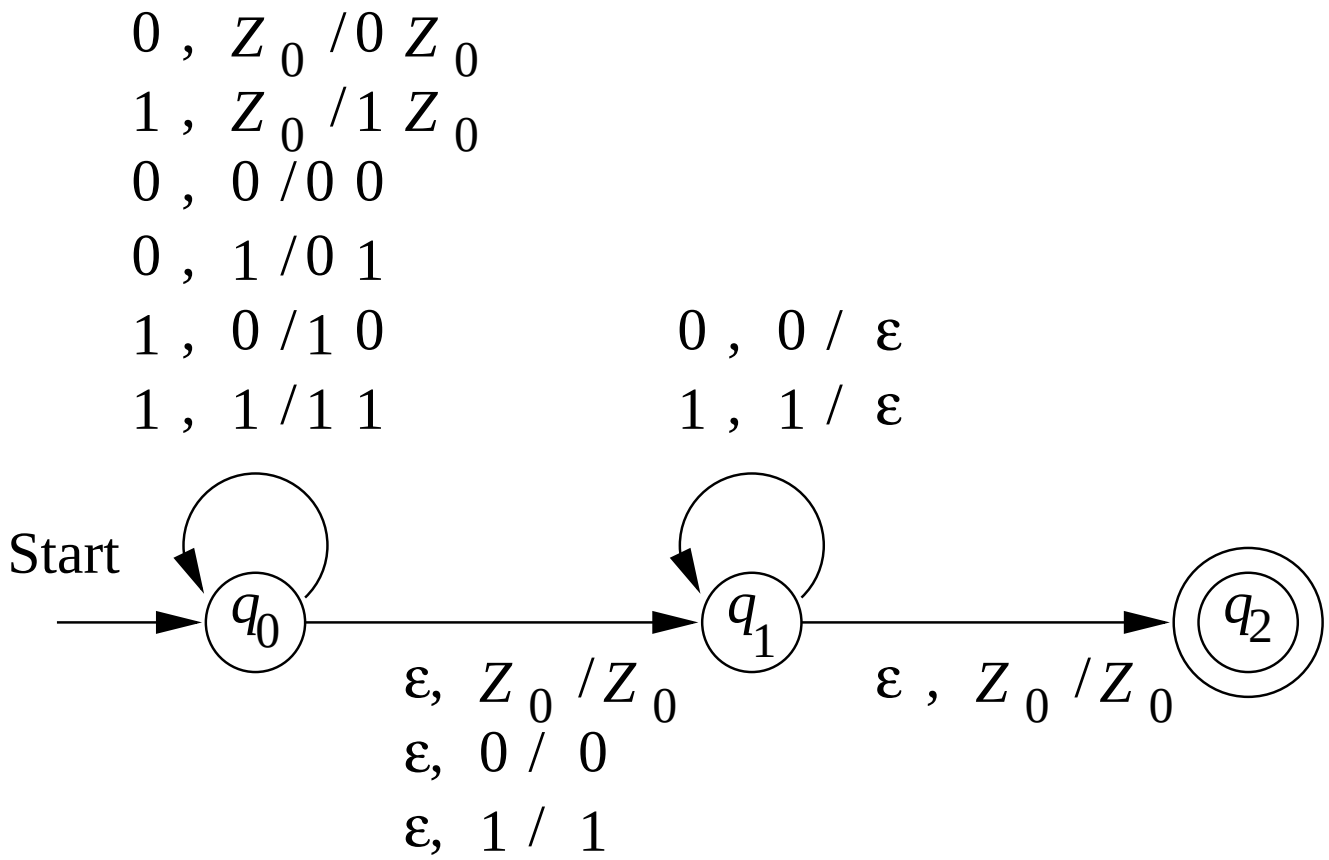
Sia $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ un PDA.
Allora $\forall w \in \Sigma^*, \beta \in \Gamma^*$:

$$(p, \alpha) \in \delta(q, a, X) \Rightarrow (q, aw, X\beta) \vdash (p, w, \alpha\beta).$$

dove $q \in Q$, $a \in \Sigma \cup \{\epsilon\}$ e $X \in \Gamma$.

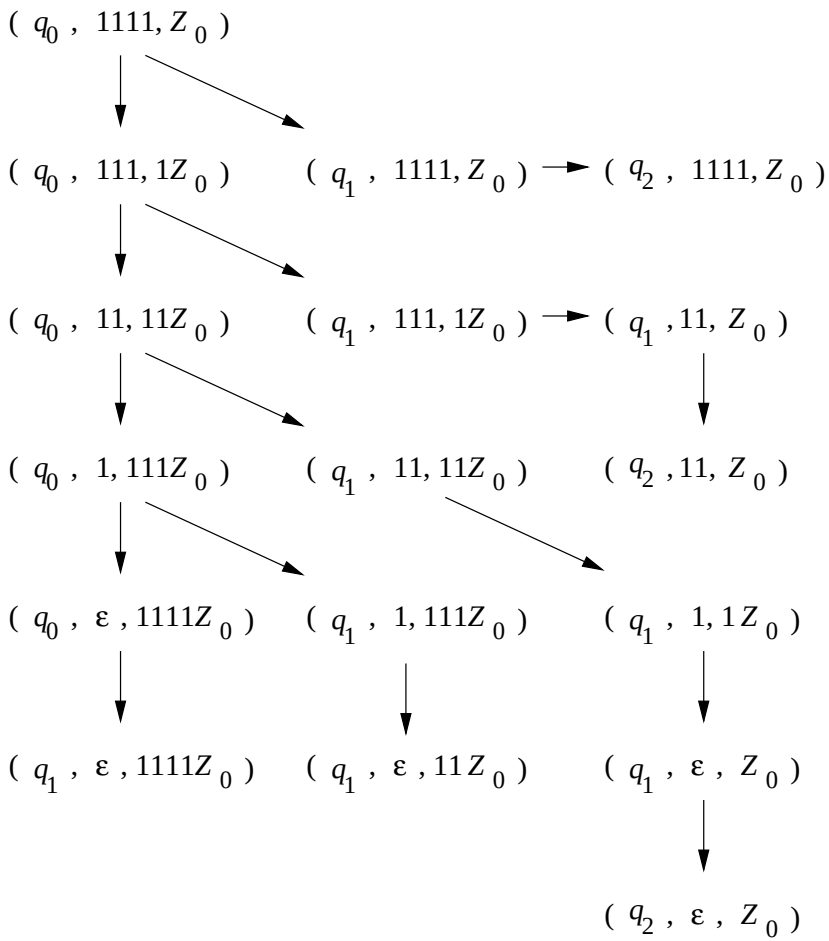
Esempio

Su input 1111 il PDA



ha le seguenti sequenze di computazioni (a causa del nondeterminismo)

Esempio


(q_0 , 1111, Z_0)
(q_0 , 111, 1Z_0)
(q_1 , 1111, Z_0)
(q_2 , 1111, Z_0)
(q_0 , 11, 11Z_0)
(q_1 , 111, 1Z_0)
(q_1 , 11, Z_0)
(q_0 , 1, 111Z_0)
(q_1 , 11, 11Z_0)
(q_2 , 11, Z_0)
(q_0 , ε , 1111Z_0)
(q_1 , 1, 111Z_0)
(q_1 , 1, 1 Z_0)
(q_1 , ε , 1111Z_0)
(q_1 , ε , 11 Z_0)
(q_1 , ε , Z_0)
(q_2 , ε , Z_0)

Non Determinismo

A causa del nondetermiminsmo e delle ϵ-transizioni

1. alcune computazioni non arrivano nello stato finale q_2
2. alcune computazioni arrivano nello stato finale q_2, ma non avendo consumato l'input
3. alcune computazioni arrivano nello stato finale q_2 avendo consumato l'input (quindi la stringa viene accettata)

Accettazione per stato finale

Sia $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ un PDA. Il *linguaggio accettato da P per stato finale* e'

$$L(P) = \{w : (q_0, w, Z_0) \overset{*}{\vdash} (q, \epsilon, \alpha),\ q \in F\}.$$

- la configurazione iniziale contiene q_0, l'input w e Z_0 sulla pila
- la configurazione finale contiene q accettante, input ϵ, e sulla pila una stringa arbitaria α

Esempio

Sia P l'automa visto. Proviamo che $L(P) = L_{wwr}$.
(direzione $\supseteq$.) Sia $x \in L_{wwr}$. Allora $x = ww^R$, e la seguente e' una sequenza di computazione legale

$$(q_0, ww^R, Z_0) \overset{*}{\vdash} (q_0, w^R, w^R Z_0) \vdash (q_1, w^R, w^R Z_0) \overset{*}{\vdash}$$

$$(q_1, \epsilon, Z_0) \vdash (q_2, \epsilon, Z_0).$$

(direzione $\subseteq$.)

Osserviamo che l'unico modo in cui il PDA puo' andare in q_2 e' se e' nello stato q_1 con la pila vuota.

Quindi e' sufficiente mostrare che se $(q_0, x, Z_0) \stackrel{*}{\vdash} (q_1, \epsilon, Z_0)$ allora $x = ww^R$, per una qualche stringa w.
Mostreremo per induzione su $|x|$ che

$$(q_0, x, \alpha) \stackrel{*}{\vdash} (q_1, \epsilon, \alpha) \quad \Rightarrow \quad x = ww^R.$$

Base: Se $x = \epsilon$ allora x e' una palindrome.
Induzione: Supponiamo che $x = a_1 a_2 \cdots a_n$, dove $n > 0$, e che l'ipotesi induttiva valga per stringhe piu' corte.
Ci sono due mosse per il PDA da ID (q_0, x, α):

(direzione $\subseteq$.)

Mossa 1: La mossa spontanea $(q_0, x, \alpha) \vdash (q_1, x, \alpha)$. Allora $(q_1, x, \alpha) \overset{*}{\vdash} (q_1, \epsilon, \beta)$ implica che $|\beta| < |\alpha|$, che implica $\beta \neq \alpha$.
Mossa 2: $(q_0, a_1 a_2 \cdots a_n, \alpha) \vdash (q_0, a_2 \cdots a_n, a_1 \alpha)$.
In questo caso c'e' una sequenza
$(q_0, a_1 a_2 \cdots a_n, \alpha) \vdash (q_0, a_2 \cdots a_n, a_1 \alpha) \vdash \cdots \vdash (q_1, a_n, a_1 \alpha) \vdash (q_1, \epsilon, \alpha)$.
Quindi $a_1 = a_n$ e

$$(q_0, a_2 \cdots a_n, a_1 \alpha) \overset{*}{\vdash} (q_1, a_n, a_1 \alpha).$$

Per il teorema 6.6 possiamo rimuovere a_n. Quindi

$$(q_0, a_2 \cdots a_{n-1}, a_1 \alpha \overset{*}{\vdash} (q_1, \epsilon, a_1 \alpha).$$

Allora, per ipotesi induttiva $a_2 \cdots a_{n-1} = yy^R$. Allora $x = a_1 yy^R a_n$ e' una palindrome.

Accettazioe per pila vuota

Sia $P = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ un PDA. Il *linguaggio accettato da P per pila vuota* e'

$$N(P) = \{w : (q_0, w, Z_0) \overset{*}{\vdash} (q, \epsilon, \epsilon)\}.$$

Nota La configurazione iniziale e' la stessa, mentre quella finale contiene uno stato q qualsiasi, ϵ come input,e la pila vuota (ϵ).

Domande

1. come modificare il PDA per le palindromi per accettare lo stesso linguaggio per pila vuota?
2. esiste un metodo generale e sistematico?
3. esiste un metodo generale e sistematico per fare anche la trasformazione opposta?

Da pila vuota a stato finale

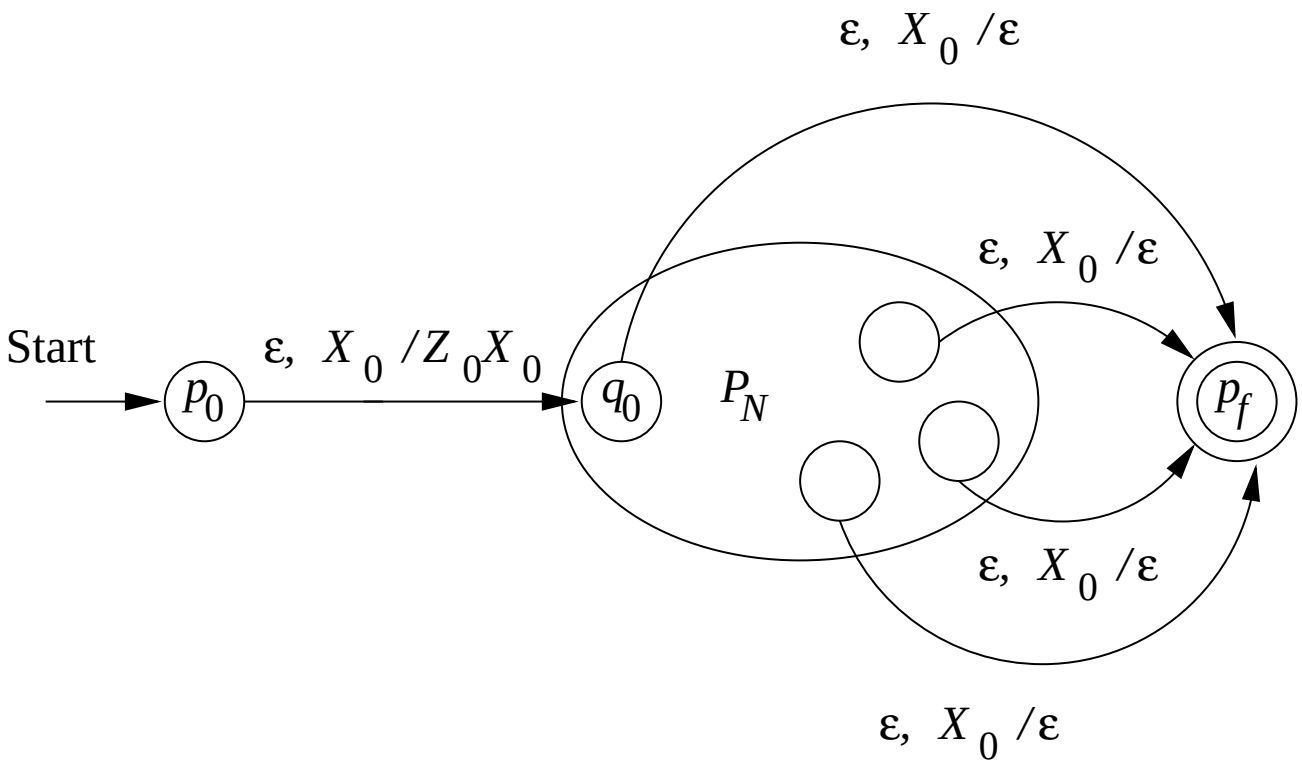
Teorema 6.9: Se $L = N(P_N)$ per un PDA $P_N = (Q, \Sigma, \Gamma, \delta_N, q_0, Z_0)$, allora $\exists$ PDA P_F, tale che $L = L(P_F)$.
Prova: Costruiamo una automa

$$P_F = (Q \cup \{p_0, p_f\}, \Sigma, \Gamma \cup \{X_0\}, \delta_F, p_0, X_0, \{p_f\})$$

dove

- $\delta_F(p_0, \epsilon, X_0) = \{(q_0, Z_0X_0)\}$, per un nuovo stato iniziale p_0
- per ogni $q \in Q, a \in \Sigma \cup \{\epsilon\}, Y \in \Gamma : \delta_F(q, a, Y) = \delta_N(q, a, Y)$
- per ogni $q \in Q$, $(p_f, \epsilon) \in \delta_F(q, \epsilon, X_0)$ per un nuovo stato p_f accettante

L'automa P_F



Intuitivamente

L'automa P_F inizia nel nuovo stato iniziale p_0

1. da p_0 si muove nello stato iniziale di q_0 di P_N impilando Z_0
2. in q_0 (trovando Z_0al top della pila) simula eseguendo le mosse di P_N, fino a quando non arriva alla pila vuota in un certo stato q
3. da q trovando X_0 sulla pila si muove nel nuovo stato accettante p_f

Prova della correttezza della costruzione

Valgono le seguenti proprieta':

1. Se una sequenza di ID e' una computazione legale per un PDA, allora lo e' anche la sequenza ottenuta aggiungendo una stringa alla fine della seconda componente (input).
2. Se una sequenza di ID e' una computazione legale per un PDA, allora lo e' anche la sequenza ottenuta aggiungendo una stringa alla fine della terza componente.
3. Se una sequenza di ID e' una computazione legale per un PDA, e la coda di un input non e' consumata, allora rimuovendola da tutte le ID si ottiene una computazione lecita.

Teorema 6.5: $\forall w \in \Sigma^*, \beta \in \Gamma^*$:

$$(q, x, \alpha) \overset{*}{\vdash} (p, y, \beta) \Rightarrow (q, xw, \alpha\gamma) \overset{*}{\vdash} (p, yw, \beta\gamma).$$

Prova: Induzione sulla lunghezza della sequenza sulla destra.
Nota: Se $\gamma = \epsilon$ abbiamo la proprieta' 1, e se $w = \epsilon$ abbiamo la proprieta' 2.
Nota: L'inverso del teorema e' falso.
Dalla proprieta' 3 abbiamo
Teorema 6.6:

$$(q, xw, \alpha) \overset{*}{\vdash} (p, yw, \beta) \Rightarrow (q, x, \alpha) \overset{*}{\vdash} (p, y, \beta).$$

Prova del Teorema 6.9

Dobbiamo mostrare che $L(P_F) = N(P_N)$.
(direzione $\supseteq$) Sia $w \in N(P_N)$. Allora

$$(q_0, w, Z_0) \overset{*}{\underset{N}{\vdash}} (q, \epsilon, \epsilon),$$

per un qualche q. Dal teorema 6.5 abbiamo

$$(q_0, w, Z_0X_0) \overset{*}{\underset{N}{\vdash}} (q, \epsilon, X_0).$$

Dato che $\delta_N \subset \delta_F$, abbiamo

$$(q_0, w, Z_0X_0) \overset{*}{\underset{F}{\vdash}} (q, \epsilon, X_0).$$

Concludiamo che

$$(p_0, w, X_0) \underset{F}{\vdash} (q_0, w, Z_0X_0) \overset{*}{\underset{F}{\vdash}} (q, \epsilon, X_0) \underset{F}{\vdash} (p_f, \epsilon, \epsilon).$$

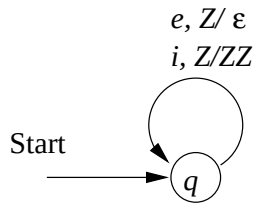
(direzione $\subseteq$) Basta esaminare il diagramma.

Esempio

Vogliamo definire un P_N per trovare errori in stringhe della grammatica *if-else* G

$$S \rightarrow \epsilon | SS | iS | iSe.$$

Problema, deve essere possibile abbinare ogni else con l'if che lo precede.
Ad esempio, $\{ieie, iie, iei\} \subseteq G$, e $\{ei, ieeii\} \cap G = \emptyset$.



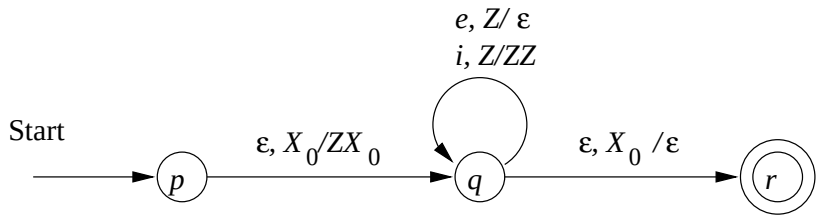
$$P_N = (\{q\}, \{i, e\}, \{Z\}, \delta_N, q, Z),$$

dove $\delta_N(q, i, Z) = \{(q, ZZ)\}$ e $\delta_N(q, e, Z) = \{(q, \epsilon)\}$.
L'automa ha un solo stato q e un solo simbolo di stack Z, che viene usato per contare la differenza tra il numero di i ed e letti. Inseriamo una Z quando troviamo una i, cancelliamo una Z quando troviamo una i. La pila parte con Z, quindi se ad un certo punto abbiamo Z^n sulla pila la differenza tra i ed e e' $n-1$. Se la pila e' vuota, allora abbiamo trovato l'errore, il numero degli e e' diventato illecito.

Da P_N otteniamo

$$P_F = (\{p, q, r\}, \{i, e\}, \{Z, X_0\}, \delta_F, p, X_0, \{r\}),$$

dove
$\delta_F(p, \epsilon, X_0) = \{(q, ZX_0)\}$,
$\delta_F(q, i, Z) = \delta_N(q, i, Z) = \{(q, ZZ)\}$,
$\delta_F(q, e, Z) = \delta_N(q, e, Z) = \{(q, \epsilon)\}$, and
$\delta_F(q, \epsilon, X_0) = \{(r, \epsilon)\}$
Il diagramma per P_F e'



Da stato finale a pila vuota

Teorema 6.11: Sia $L = L(P_F)$, per un PDA $P_F = (Q, \Sigma, \Gamma, \delta_F, q_0, Z_0, F)$. Allora $\exists$ PDA P_n, tale che $L = N(P_N)$.

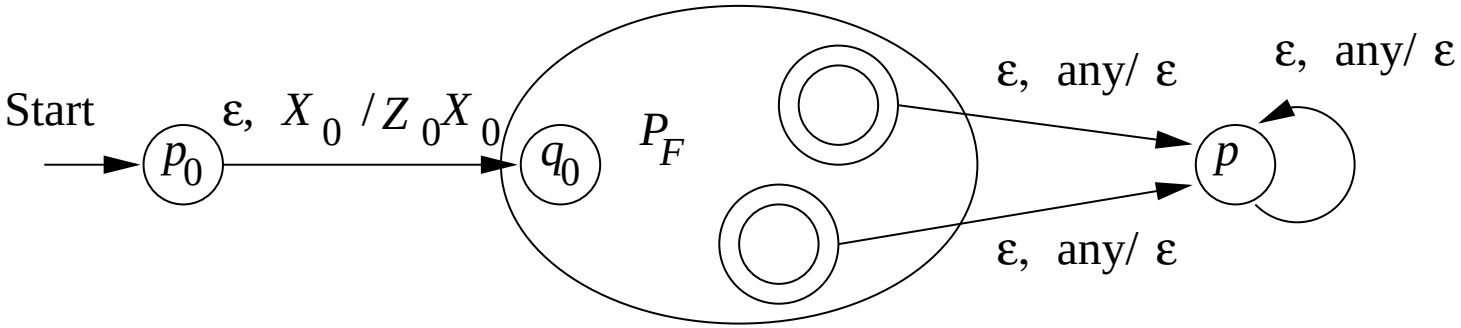
Prova: Costruiamo

$$P_N = (Q \cup \{p_0, p\}, \Sigma, \Gamma \cup \{X_0\}, \delta_N, p_0, X_0)$$

dove

- $\delta_N(p_0, \epsilon, X_0) = \{(q_0, Z_0X_0)\}$ per un nuovo stato iniziale p_0
- per tutti i $q \in Q$, $a \in \Sigma \cup \{\epsilon\}$, $Y \in \Gamma$, $\delta_N(q, a, Y) = \delta_F(q, a, Y)$
- $\delta_N(p, \epsilon, Y) = \{(p, \epsilon)\}$, per un nuovo stato p per $Y \in \Gamma \cup \{X_0\}$,
- per tutti i $q \in F$ e $Y \in \Gamma \cup \{X_0\}$ $(p, \epsilon) \in \delta_N(q, \epsilon, Y)$.

L'automa P_N



Intuitivamente

L'automa P_N inizia nel nuovo stato iniziale p_0

1. da p_0 si muove nello stato iniziale di q_0 di P_F impilando Z_0
2. in q_0 (trovando Z_0 al top della pila) simula eseguendo le mosse di P_F, fino a quando non arriva in uno stato accettante $q \in F$
3. da $q \in F$ si muove nel nuovo stato p, in cui la pila viene svuotata (senza leggere input)

Dobbiamo mostrare che $N(P_N) = L(P_F)$.
(direzione $\subseteq$) Basta esaminare il diagramma.
(direazione $\supseteq$) Sia $w \in L(P_F)$. Allora

$$(q_0, w, Z_0) \overset{*}{\underset{F}{\vdash}} (q, \epsilon, \alpha),$$

per un $q \in F, \alpha \in \Gamma^*$. Dato che $\delta_F \subseteq \delta_N$, e teorema 6.5 dice che X_0 puo' essere infilato sotto la pila, otteniamo

$$(q_0, w, Z_0X_0) \overset{*}{\underset{N}{\vdash}} (q, \epsilon, \alpha X_0).$$

Il P_N puo' calcolare:

$$(p_0, w, X_0) \underset{N}{\vdash} (q_0, w, Z_0X_0) \overset{*}{\underset{N}{\vdash}} (q, \epsilon, \alpha X_0) \overset{*}{\underset{N}{\vdash}} (p, \epsilon, \epsilon).$$