Aula 4

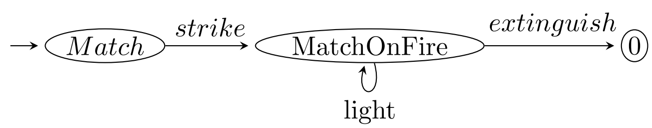
Definições Recursivas em CCS

- Usando *nomes* para sub-expressões e que podem ser usadas em expressões.
- Em vez de $strike.(light.0 + \tau.0)$

$$\begin{array}{rcl} Match & := & strike.MatchOneZeroLight \\ MatchOneZeroLight & := & light.0 + \tau.0 \end{array}$$

- sendo que agora podemos representar repetição (iteração)

$$\begin{array}{rcl} Match & := & strike.MatchOnFire \\ MatchOnFire & := & light.MatchOnFire + extinguish.0 \end{array}$$



- *Convenção*: nomes começam por letra maiúscula e ações por minúscula. Também são chamadas de constantes ou variáveis (?!).

CCS_0^ω : Sequencial com iteração

Seja Act um conjunto de ações e Var um conjunto de nomes (variáveis). As expressões do CCS_0^ω são

$$P \quad ::= \quad 0 \mid X \mid P + P \mid \alpha.P$$

onde

- $\alpha \in Act$,
- $X \in Var$ e
- um conjunto Γ de equações da forma $X := P$, representados por pares (X, P)

$$\Gamma = \{(X_1, P_1), \ldots, (X_n, P_n)\}$$

e

$$\Gamma(X_i) = P_i$$

Exemplo 4.1. *1. $X := a.b.X$ então temos $\Gamma = \{(X, a.b.X)\}$*

2. Se

$$\begin{array}{rcl} X & := & a.b.Y \\ Y & := & b.Z + a.Y \\ Z & := & a.Y \end{array}$$

temos $\Gamma = \{(X, a.b.Y), (Y, b.Z + a.Y), (Z, a.Y)\}$

Semântica do CCS_0^ω

Regras do CCS_0^ω

$$\text{Prefixo}\ \frac{}{\alpha.P \xrightarrow{\alpha} P}$$

$$\text{EscolhaE}\ \frac{P \xrightarrow{\alpha} P'}{P+Q \xrightarrow{\alpha} P'}$$

$$\text{EscolhaD}\ \frac{Q \xrightarrow{\alpha} Q'}{P+Q \xrightarrow{\alpha} Q'}$$

$$\text{Rec}\ \frac{P \xrightarrow{\alpha} P' \qquad \Gamma(X) = P}{X \xrightarrow{\alpha} P'}$$

Semântica do CCS_0^ω

A relação $\longrightarrow_\Gamma$ é a menor tal que, para todo $\alpha \in Act$ e $P, Q \in CCS_0^\omega$,

- $(\alpha.P, \alpha, P) \in \longrightarrow_\Gamma$
- $(P+Q, \alpha, P') \in \longrightarrow_\Gamma)$ se $(P, \alpha, P') \in \longrightarrow_\Gamma)$
- $(P+Q, \alpha, Q') \in \longrightarrow_\Gamma)$ se $(Q, \alpha, Q') \in \longrightarrow_\Gamma)$
- $(X, \alpha, P') \in \longrightarrow_\Gamma$ *se* $\Gamma(X) = P$ *e* $(P, \alpha, P') \in \longrightarrow_\Gamma$
- nada mais está em $\longrightarrow_\Gamma$

Regra prática

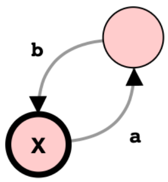
Tentar aplicar as regras a um P até encontrar uma nova transição $P \xrightarrow{\alpha} P'$. E repetir para P' caso não fosse conhecido ele ou o seu comportamento.

Exemplos

$[\![X]\!]_\Gamma$ e $\Gamma = \{(X, a.b.X)\}$

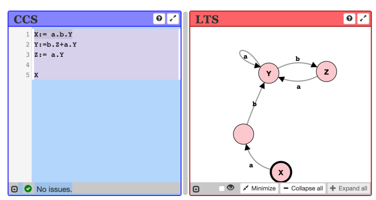
$$\text{Rec}\ \frac{\text{Prefixo}\ \dfrac{}{a.b.X \xrightarrow{a} b.X} \qquad \Gamma(X) = a.b.X}{X \xrightarrow{a} b.X}$$

$$\text{Prefixo}\ \frac{}{b.X \xrightarrow{b} X}$$



O segundo estado é etiquetado por $b.X$

$[\![Y]\!]_\Gamma$ e $\Gamma = \{(X, a.b.Y), (Y, b.Z + a.Y), (Z, a.Y)\}$



Exercício 4.1. *Determinar formalmente* $[\![Y]\!]_\Gamma$. $\diamond$

Exemplos

$[\![X]\!]_\Gamma$ e $\Gamma = \{(X, 0 + Y), (Y, a.X)\}$

$$\text{Rec}\;\dfrac{\text{EscolhaD}\;\dfrac{\text{Rec}\;\dfrac{\text{Prefixo}\;\dfrac{}{a.X \xrightarrow{a} X} \qquad \Gamma(Y) = a.X}{Y \xrightarrow{a} X}}{0 + Y \xrightarrow{a} X} \qquad \Gamma(X) = 0 + Y}{X \xrightarrow{a} X}$$

Desenha o LTS $[\![X]\!]_\Gamma$ e $[\![Y]\!]_\Gamma$

Exemplos

Como calcular:

- $[\![X]\!]_\Gamma$ e $\Gamma = \{(X, X)\}$
- $[\![X]\!]_\Gamma$ e $\Gamma = \{(X, X + a.0)\}$
- $[\![X_0]\!]_\Gamma$ e $\Gamma = \{(X_i, a.X_{i+1}) \mid i \geq 1\}$

Exercício 4.2. *Dado*

$BurningMatch := burn.BurningMatch + extinguish.0$

calcula o LTS do seguinte processo $[\![strike.(BurningMatch + smolder.0)]\!]$.

Usa também o Pseuco.com.

$\diamond$

Exemplo 4.2. *Tenta calcular* $[\![X]\!]$ *com* $\Gamma = (X, X + a.X)$

Se se usar a regra EscolhaE podemos ter árvores de dedução infinitas que não levam a novas transições. Assim iremos evitar esse tipo de expressão (embora sintaticamente correctas).

Expressões guardadas

- Uma variável X é *guardada* na expressão P se cada ocorrência de X em P ocorre numa expressão $\alpha.Q$
- caso contrário *não é guardada*
- Uma expressão é guardada se todas as suas variáveis são guardadas
- caso contrário *não é guardada*
- Ex. não guardadas: X, $a.0 + X$, $\tau.X + Y$, $a.X + Y$
- Ex. guardadas: $a.X$, $a.(X + Y)$, $\tau.X + a.Y$, $a.(X + b.Y)$
- Se $P \in CCS_0^\omega$ é guardada e os valores de Γ são guardados, é possível calcular $[\![P]\!]_\Gamma$ pela regra prática, isto é o processo termina.
- No pseuco.com só há expressões guardadas

Semântica do CCS Sequencial (III)

$$LTS_0^\omega = \{(CCS_0^\omega, T, P) \mid T \subseteq (CCS_0^\omega \times Act \times CCS_0^\omega, P \in CCS_0^\omega\}$$

Conjunto de todos os LTS sobre expressões do CCS_0^ω com um conjunto de variáveis Var. A semântica das expressões do CCS_0^ω é então

$$[\![\_]\!] : (Var \to CCS_0^\omega) \to CCS_0^\omega \to LTS_0^\omega$$

tal que

$$[\![P]\!]_\Gamma = (CCS_0^\omega, \longrightarrow_\Gamma, P)$$

com $\longrightarrow_\Gamma$ definida anteriormente.

Exercício 4.3. *Mostrar que para cada LTS estados-finito existe uma expressão* $P \in CCS_0^\omega$ *tal que* $TS \sim [\![P]\!]_\Gamma$. $\diamond$

Exemplo de $1 - Buffer$

$$Buffer \quad := \quad put?.get?.Buffer$$

Calcula $[\![Buffer]\!]_\Gamma$.

$$BufferM \quad := \quad put?.get?.BufferM + get?.put?.BufferM$$

Calcula $[\![put?.BufferM]\!]_\Gamma$.

$$\begin{aligned}
Buffer0 &:= put?.Buffer1 \\
Buffer1 &:= get?.Buffer2 + get?.Buffer0 \\
Buffer2 &:= get?.Buffer1
\end{aligned}$$

Calcula $[\![Buffer0]\!]_\Gamma$.