Review:

$$CTM = coin . (\overline{tea} . CTM + \overline{coff} . CTM)$$

denotes an infinite process execution - an infinite tree

What about:

$$CM = coin . (\overset{\text{halt}}{0} + \overline{coff} . CM)$$

can it run forever? will it?

Related question - grammar:

$$N ::= 1 \mid 0N$$

what string set does this define?

(Sheila Greibach)

Usual (lfp / inductive) interpretation:

$$n_0 = \{\}$$

$$n = \bigcup_{i \geq 0} n_i \qquad n_{i+1} = \{1\} \cup \{0 \cdot s \mid s \in n_i\}$$

$$= \{0^k 1 \mid k \geq 0\}$$

no infinite strings ☹

gfp / coinductive interpretation:

$$m_0 = \{s \mid s \text{ is a finite or infinite sequence of chars in } \{0, 1, 2, \cdots\}\}$$

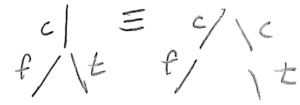
$$m = \bigcap_{i \geq 0} m_i \qquad m_{i+1} = \{1\} \cup \{0 \cdot s \mid s \in m_i\}$$

exercise: calculate the m_is.

$$= \{0^k 1 \mid k \geq 0\} \cup \{0^\omega\}$$

000000000000000000 . . .

In Linear-time model (meanings are paths/strings):

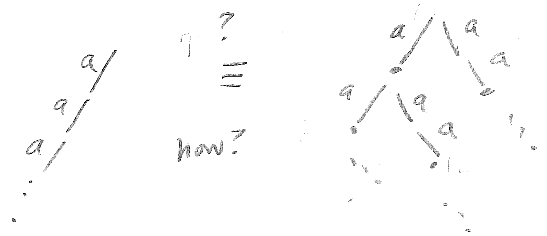


In Branching-time model (" " are trees): $\not\equiv$

And trees can be infinite: $CTM = c.(f.CTM + t.CTM)$ How?

Ex: $T = a.T$ $\quad U = a.U + \underbrace{a.a.U}_{U'}$

The Trees?



$\begin{cases} U = a.U + a.U' \\ U' = a.U \end{cases}$

must prove $T \approx U$ and $T \approx U'$ simultaneously.

R. Milner: "depth-equivalence":

$T \approx U$ iff $\forall i \geq 0$: $T \approx_i U$

where: $X \approx_0 Y$ for every comb. X, Y

$X \approx_{i+1} Y$ iff $X \xrightarrow{m} X'$ then $Y \xrightarrow{m} Y'$ and $X' \approx_i Y'$

and vice versa

leave off — then simulation

"(Co)Induction principle":

replace $\approx_k$ by some 'R':

$X \, R \, Y$ iff $X \xrightarrow{m} X'$ then $Y \xrightarrow{m} Y'$ and $X' \, R \, Y'$

is this the same as $\approx$?

D. Park: yes — if we formalize trees coinductively.

For tree eqn: $T = \ldots T \ldots$

define $\lim_{i>0} T_i = \bigcap_{i>0}^{\text{overlay}} T_i$

$\begin{cases} T_0 = \text{infinite chaos tree} \text{ — all branches forever!} \\ T_{i+1} = \ldots T_i \ldots \end{cases}$

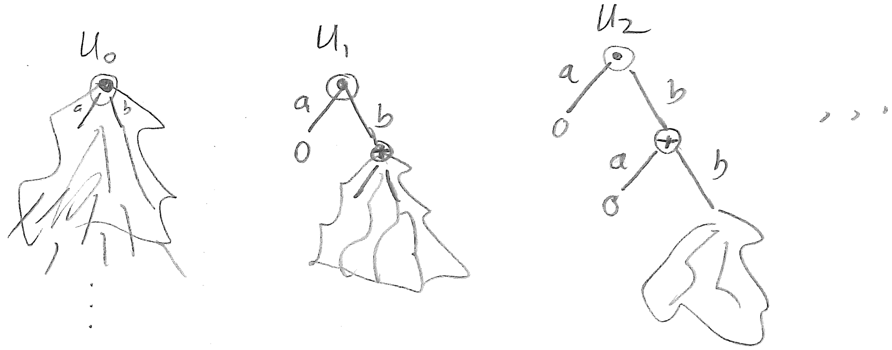
Ex: let alphabet be $\{a, b, c\}$. Then chaos tree is

infinitely deep

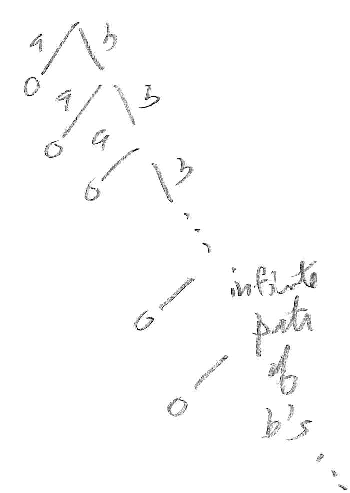
U_0 = chaos

$U_{i+1} = a.0 + b.U_i$

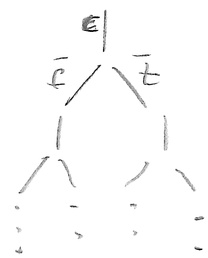
Ex: $U = a.0 + b.U$



the overlay of all U_i's:



$M = c.(\bar{f}.M + \bar{t}.M)$



Prove a prop of such a tree: infinite tree needs recursive prop.

noSoda(t) iff $(t \xrightarrow{c}\xrightarrow{m} t', \ m \neq \text{soda}) \wedge$ noSoda(t')

Define set of trees:

$$NS = \{ t \mid t \xrightarrow{c}\xrightarrow{m} t', \ m \neq \text{soda}, \ t' \in NS \}$$

$$NS = \bigcap_{i \geq 0} NS_i \qquad NS_0 = \text{all trees}$$

$$NS_{i+1} = \{ t \mid t \xrightarrow{c}\xrightarrow{m} t', \ m \neq \text{soda}, \ t' \in NS_i \}$$

To prove $M \in NS$, show $M_i \in NS_i \quad \forall i \geq 0$

hence $M = \bigcap M_i \in \bigcap_i NS_i = NS$

Follows from: assuming $M \in NS$, prove: $M \xrightarrow{c}\xrightarrow{m} t', \ m \neq \text{soda}, \ t' \in NS$

Def binary relns, like $\approx \ \subseteq \text{Tree} \times \text{Tree}$, similarly.