Máquinas de Turing y lenguajes:

Def:

Suponga que $M = (K, \Sigma, \delta, s, H)$ es una máquina de Turing con $H = \{ Y, N \}$ (Y: acepta, N: rechaza)

Si $w \in (\Sigma \setminus \{\sqcup, \rhd\})^*$ decimos que

(1) M **acepta** a w si

$$(s, \rhd \underline{\sqcup} w) \vdash_M^* (y, \rhd u)$$

(2) M **rechaza** a w si

$$(s, \rhd \underline{\sqcup} w) \vdash_M^* (N, \rhd u)$$

sin importar qué ($\rhd u$)

Def: Si $L \subseteq \Sigma_0^*$ decimos que **M decide a L** si $\forall w \in \Sigma_0^*$ se cumplen las siguientes:

(i) $w \in L \Rightarrow$ M acepta a w

(ii) $w \notin L \Rightarrow$ M rechaza a w

Def: **$L \subseteq \Sigma_0^*$ es un lenguaje recursivo** si existe $\Sigma \supseteq \Sigma_0$ y $M = (K, \Sigma, \delta, s, \{Y, N\})$ tal que M decide a L.

Nota que, incluso si $H = \{Y, N\}$, una máquina de Turing podría no responder a la pregunta $w \in L$, pues podría NO parar nunca con input $(s, \rhd \underline{\sqcup} w) \vdash_M^* \cdots$ (nunca llega a un halting state)

Def: Una máquina de Turing $M = (K, \Sigma, \delta, s, H)$ __semidecide__ un lenguaje L si $\forall w \in \Sigma_0^*$ se tiene

(i) $w \in L \Rightarrow (s, \rhd \underline{\sqcup} w) \vdash_M^* (h, u)$ para $h \in H$

(ii) $w \notin L \Rightarrow (s, \rhd \underline{\sqcup} w) \vdash_M^* \cdots$ NO PARA.

Def: $L \subseteq \Sigma_0^*$ es __recursivamente enumerable__ si existe una máquina de Turing M que semidecida a L.

No es difícil ver que

$$\left\{ \begin{array}{l} \text{Lenguajes} \\ \text{recursivamente enumerables} \end{array} \right\} \subseteq \left\{ \begin{array}{c} \text{Lenguajes recursivos} \\ (\text{ó decidibles}) \end{array} \right\}$$

La inclusión es ESTRICTA, como veremos después...

Máquinas de Turing NO deterministas

$M = (K, \Sigma, \Delta, s, H)$ es una MTND

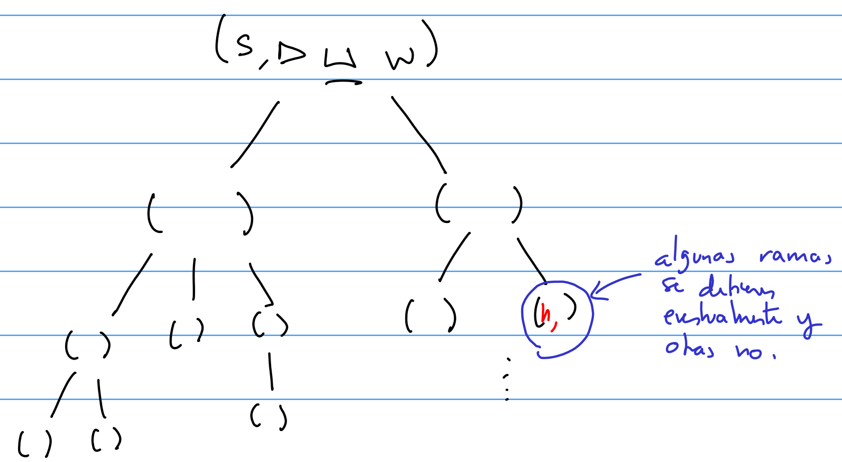
si

$$\Delta \subseteq (K \setminus H) \times \Sigma \times K \times (\Sigma \cup \{\leftarrow, \rightarrow\})$$

Δ es solo una relación y no una función así que se permite

$$(p, a, q, b)$$
$$(p, a, q', b')$$

Si dos reglas (o más) pueden aplicarse consideramos ambas ejecuciones como futuros posibles y seguimos adelante... esto lleva a un multiverso (árbol) de ejecuciones



Def: Sea $M = (K, \Sigma, \Delta, s, H)$ una máquina de Turing NO DET.

- Si $L \subseteq \Sigma_0^*$ decimos que M semidecide a L si $\forall w \in \Sigma_0^*$ se tiene

  (i) $w \in L \Rightarrow M$ acepta a w (es decir, si M inicia con $(s, \triangleright\underline{\sqcup}\, w)$ existe una rama que termina en un estado del tipo $(h, \triangleright u)$ arbitrario para algún $h \in H$. Note que

M acepta a w incluso si en otras ramas no para.)

(ii) $w \notin L$ $\Rightarrow$ Toda rama del arbol de ejecucion, iniciando en $(s, \rhd \underline{\sqcup} w)$ es infinita (no para en ningún mundo)

con $H = \{Y, N\}$

Def: Si $L \subseteq \Sigma_0^*$ decimos que M decide a L si $\forall w \in (\Sigma \setminus \{\rhd, \sqcup\})^*$ tenemos

(i) El árbol de cómputo iniciado en $(s, \rhd \underline{\sqcup} w)$ es finito y

(ii) $w \in L \iff$ el árbol contiene algún estado

$(Y, \rhd m)$

(incluso si contiene otras ramas que terminan en $(N, \rhd m)$)

Teorema Si una máquina de Turing NO DET semidecide o decide a un lenguaje L entonces existe una máquina DETERMINISTA que semidecide o decide a L

(la máquina determinista es, en todas las construcciones conocidas exponencialmente más lenta).