

# Informatique Théorique 3

## 2004/05

### *Semaine 11*

Voir [HMU03] §8 (en particulier §8.2).

Voir [HMU03] §8.2.6 et [Sch95] §2.2.

```
main( )  
{  
    printf("hello, world");  
}
```

```
int exp(int i n)
/* computes i to the power n */
...

main()
{
    int n, total, x, y, z;
    scanf("%d", &n);
    total = 3;
    while (1) {
        for (x=1; x<=total-2; x++)
            for (y=1; y<=total-x-1; y++) {
                z = total - x - y;
                if ( exp(x,n) + exp(y,n) == exp(z,n) )
                    printf("hello, world")
            }
        total++
    }
}
```

### 4.1.1 Définition (TM) Une *machine de Turing* (TM) est un septuplet

$$M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$$

où

- $Q$  est un ensemble fini d'états,
- $\Sigma$  est un ensemble fini de symboles, *l'alphabet d'entrée*,
- $\Gamma \supset \Sigma$  est un ensemble fini de symboles, *l'alphabet du ruban*,
- $\delta : (Q \times \Gamma) \rightarrow (Q \times \Gamma \times \{-1, 0, 1\})$  est la *fonction de transition*,
- $q_0 \in Q$  est l'état initial (ou de départ),
- $B \in \Gamma \setminus \Sigma$  est le *symbole blanc*,
- $F \subseteq Q$  est l'ensemble des états accepteurs (ou finaux).

En général,  $\delta$  est une fonction partielle.

Il existe aussi des versions des TM où  $\delta$  est de type

<i>multi/k-rubans</i> :	$(Q \times \Gamma^k) \rightarrow (Q \times \Gamma^k \times \{-1, 0, 1\}^k)$
<i>non-déterministe</i> :	$(Q \times \Gamma) \rightarrow \mathcal{P}(Q \times \Gamma \times \{-1, 0, 1\})$

## 4.1.4 Définition (Configurations et calculs)

Soit  $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$  une TM.

1. Une *configuration* de  $M$  est un triplet  $(q, i, \text{rub})$  où
    - $q \in Q$  est l'état dans lequel se trouve la machine  $M$ ,
    - $i \in \mathbb{Z}$  est l'emplacement courant de la tête de lecture/écriture, et
    - $\text{rub} : \mathbb{Z} \rightarrow \Gamma$  est une application représentant l'état du ruban telle que  $\text{rub}(k) \neq B$  pour au plus un nombre fini d'entiers  $k$ .
- Si  $q \in F$ , on dit que  $(q, i, \text{rub})$  est une *configuration acceptante* de  $M$ .

2. Soit  $c = (q, i, \text{rub})$  une configuration de  $M$  et  $X = \text{rub}(i)$ .

Si  $\delta(q, X) = (q', Y, d)$ ,

alors la configuration  $c$  se réduit en une étape de calcul

en la configuration  $c' = (q', i + d, \text{rub}\{i \mapsto Y\})$ , noté  $c \vdash_M c'$ , avec

$$\begin{aligned} \text{rub}\{i \mapsto Y\} : \mathbb{Z} &\rightarrow \Gamma \\ k &\mapsto Y && \text{si } k = i \\ k &\mapsto \text{rub}(k) && \text{sinon} \end{aligned}$$

Comme d'habitude, on note  $\vdash_M^*$  la fermeture réflexive et transitive de la relation  $\vdash_M$  ainsi définie.

3. On identifie tout mot  $w \in \Gamma^*$  avec le ruban  $w : \mathbb{Z} \rightarrow \Gamma$  défini par :

$$\begin{aligned} w : \mathbb{Z} &\rightarrow \Gamma \\ k &\mapsto (w)_k && \text{si } 1 \leq k \leq |w| \\ k &\mapsto B && \text{sinon} \end{aligned}$$

4. Soit  $w \in \Sigma^*$ . Un **calcul de  $M$  pour  $w$**  est une séquence

$$c_0 \vdash_M c_1 \vdash_M \dots \vdash_M c_i \vdash_M \dots$$

où pour tout  $k$ ,  $c_k$  est une configuration de  $M$  et  **$c_0 = (q_0, 1, w)$**  est la configuration initiale du calcul.

5. Soit  $c$  un configuration de  $M$ . S'il n'existe aucune configuration  $c'$  telle que  $c \vdash_M c'$ , on note  **$c \not\vdash_M$** .

Lorsque la machine  $M$  dont on parle est claire dans le contexte, on écrit simplement  $\vdash$  à la place de  $\vdash_M$ .

#### 4.1.5 Définition (Langage accepté par une TM)

Soit  $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$  une TM.

Le *langage accepté par  $M$*  est défini par :

$$L(M) \triangleq \{ w \in \Sigma^* \mid \text{il existe } c \text{ acceptante telle que } (q_0, 1, w) \vdash_M^* c \}$$

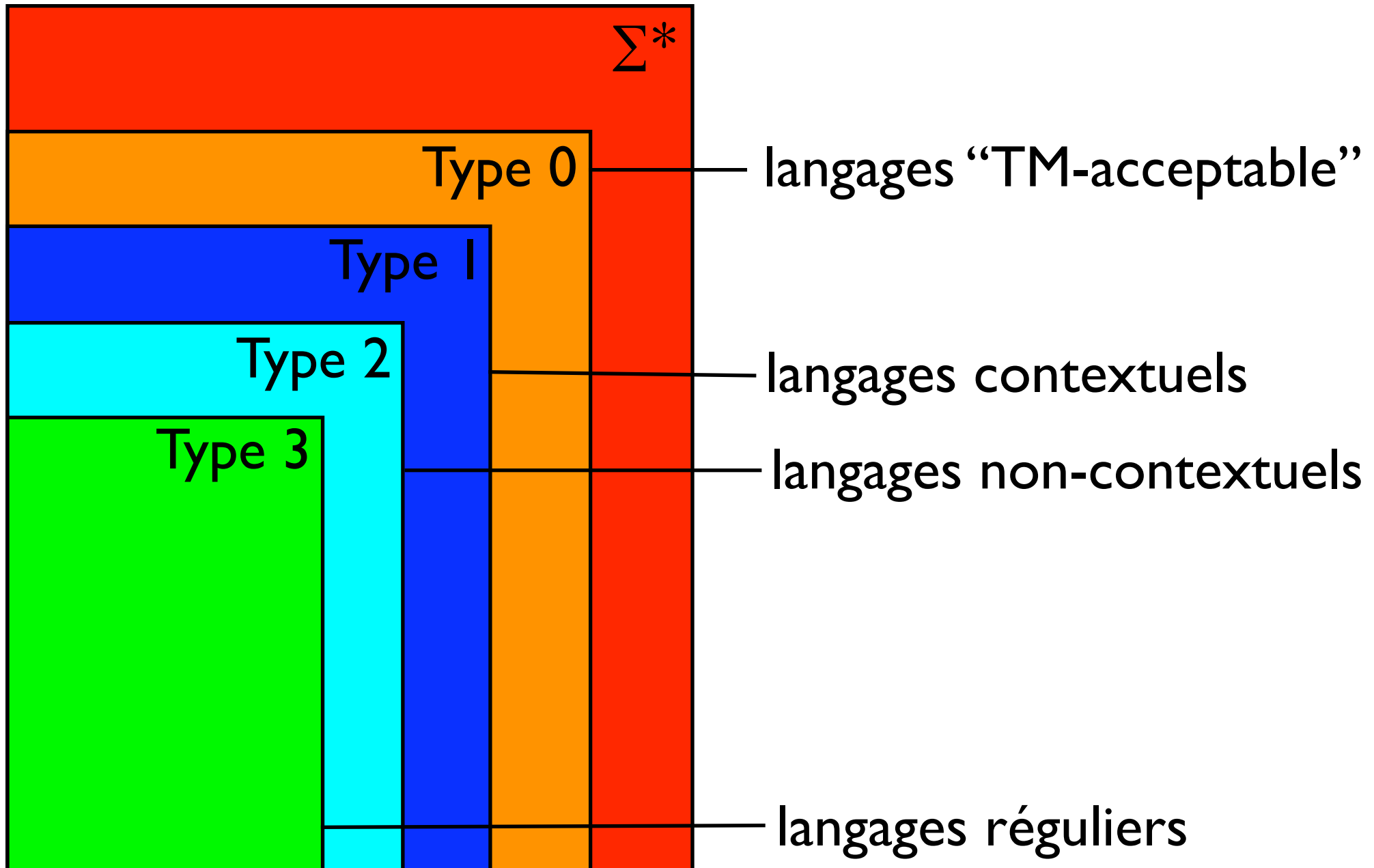
#### 4.1.6 Théorème Soit $\Sigma$ un alphabet et $L \subseteq \Sigma^*$ .

Alors les deux propositions suivantes sont équivalentes :

1. Il existe une TM  $M$  telle que  $L = L(M)$ .
2. Il existe une grammaire  $G$  de type 0 telle que  $L = L(G)$ .



# Hiérarchie de Chomsky ...



### 4.1.3 Notation (Graphe d'une TM)

Si  $\delta(q, X) = (q', Y, d)$ ,

on dessine une flèche étiquetée par  ~~$X, Y/d$~~  de  $q$  à  $q'$ .

$X / Y, d$

### 4.2.1 Définition (Arrêt)

Soit  $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$  une TM.

On dit que  $M$  s'arrête pour  $w \in \Sigma^*$

s'il existe une configuration  $c$  telle que  $(q_0, 1, w) \vdash_M^* c \not\vdash_M$ .

Si  $M$  s'arrête pour tout  $w \in \Sigma^*$ , alors  $M$  est dite *totale*.

**4.2.2 Notation** Soit  $M = (Q, \Sigma, \Gamma, \delta, q_0, B, F)$  une TM.

1. Si l'on ne s'intéresse pas au langage accepté par une TM  $M$ , mais seulement à la question de l'arrêt, on peut poser  $F = \emptyset$  ou bien omettre  $F$ .
2. Si l'on veut imposer qu'une TM s'arrête au moment où elle accepte (c-à-d, où elle passe dans un état accepteur), on peut restreindre le type de  $\delta$  à

$$((Q \setminus F) \times \Gamma) \rightarrow (Q \times \Gamma \times \{-1, 0, 1\})$$

1. Une fonction  $f : \Sigma^* \rightarrow \Sigma^*$  est dite *Turing-calculable*, s'il existe une TM  $M = (Q, \Sigma, \Gamma, \delta, q_0, B, \emptyset)$  telle que pour tout  $x, y \in \Sigma^*$

$$f(x) = y$$

ssi

$$(q_0, 1, x) \vdash^* (q, 1, y) \nmid$$

2. Soit  $\text{rep}(n)$  la suite  $1 \dots 1$  de longueur  $n \in \mathbb{N}$ .  
Une fonction  $f : \mathbb{N}^k \rightarrow \mathbb{N}$  est dite *Turing-calculable*, s'il existe une TM  $M = (Q, \{1\}, \{1, \#\} \cup \Gamma, \delta, q_0, B, \emptyset)$  telle que pour tout  $n_1, \dots, n_k, m \in \mathbb{N}$

$$f(n_1, \dots, n_k) = m$$

ssi

$$(q_0, 1, \text{rep}(n_1)\# \dots \# \text{rep}(n_k)) \vdash^* (q, 1, \text{rep}(m)) \nmid$$

## 4.2.4 Thèse (Church-Turing)

Tout *algorithme*, c-à-d toute *procédure effective*, peut être représenté par une TM.

**4.3.1 Définition** Soit  $L$  un langage sur l'alphabet  $\Sigma$ .

1.  $L$  est dit **décidable (ou : récursif)**  
s'il existe une TM  $M$  telle que  $L(M) = L$  et  $M$  est totale.
2.  $L$  est dit **semi-décidable (ou : récursivement énumérable, ou : r.e.)**  
s'il existe une TM  $M$  telle que  $L(M) = L$ .
3.  $L$  est dit **co-semi-décidable (ou : co-r.e.)**  
si son complément  $\bar{L}$  est semi-décidable.
4.  $L$  est dit **indécidable**  
si  $L$  n'est pas décidable.
5.  $L$  est dit **non-semi-décidable (ou : non-r.e.)**  
si  $L$  n'est pas semi-décidable.

**4.3.2 Proposition** Soit  $L$  un langage sur l'alphabet  $\Sigma$ .

1. Si  $L$  est décidable, alors son complément  $\bar{L}$  est aussi décidable.
2. Si  $L$  et  $\bar{L}$  sont semi-décidables, alors  $L$  est décidable.

# Hiérarchie de Chomsky ...

