– Compléments pour l'exercice 4 –

Considérons le langage

$$L := \{\langle M_1, M_2\rangle \in \Sigma^\star \mid \mathscr{L}(M_1) = \mathscr{L}(M_2)\}.$$

But. Montrons que L n'est pas décidable, ni Turing-reconnaissable, ni co-Turing-reconnaissable.

On admet que les langages

$$A_{\mathsf{TM}} = \{\langle M, w\rangle \in \Sigma^\star \mid M \text{ accepte } w\}$$
$$\bar{A}_{\mathsf{TM}} = \{\langle M, w\rangle \in \Sigma^\star \mid M \text{ n'accepte pas } w\} = \Sigma^\star \setminus A_{\mathsf{TM}}$$

sont indécidables. On admet également que A_{TM} est Turing-reconnaissable mais pas co-Turing-reconnaissable. Réciproquement, $\bar{A}_{\mathsf{TM}}$ est co-Turing-reconnaissable mais pas Turing-reconnaissable.

Par la suite, on considère deux machines :
- la machine $M_\emptyset$ qui rejette toutes les entrées ;
- la machine $M_{w,N}$ dont l'implémentation est ci-dessous :

(1) On ignore l'entrée.
(2) On simule N sur w.
(3) Si N accepte w, alors on accepte.

Son langage est

$$\mathscr{L}(M_{w,N}) = \begin{cases} \Sigma^\star & \text{si } N \text{ accepte } w \\ \emptyset & \text{sinon.} \end{cases}$$

I. Une première réduction.

On procède par réduction au langage $\bar{A}_{\mathsf{TM}}$. On considère la fonction calculable

$$\begin{aligned} f : \quad \Sigma^\star &\longrightarrow \Sigma^\star \\ \langle N, w\rangle &\longmapsto \langle M_{w,N}, M_\emptyset\rangle. \end{aligned}$$

et on a l'équivalence :

$$\begin{aligned}\langle M_{w,N}, M_\emptyset\rangle \in L &\iff \mathscr{L}\big(M_{w,N}\big) = \mathscr{L}(M_\emptyset) = \emptyset \\ &\iff N \text{ n'accepte pas } w \\ &\iff \langle N, w\rangle \in \bar{A}_{\mathsf{TM}}.\end{aligned}$$

De cette réduction, par la question 2 de l'exercice 1, et parce que le langage $\bar{A}_{\mathsf{TM}}$ n'est pas décidable, L n'est pas décidable.

Mais, on sait également que L n'est pas Turing-reconnaissable en appliquant le lemme ci-dessous.

Lemme

Si A se réduit à B alors :
- si B est Turing-reconnaissable, alors A aussi ;
- si A n'est pas Turing-reconnaissable, alors B non plus.

Preuve (C'est une preuve quasi-identique à celle de Q2 dans l'exercice 1).
- Supposons que A se réduit (via une fonction f calculable) à B et que B est Turing-reconnaissable par une machine R. On a donc l'équivalence $w \in A \iff f(w) \in B$. On construit la machine ci-dessous qui reconnait A.

  (1) On calcule $f(w)$.
  (2) On simule R sur l'entrée $f(w)$.
  (3) Si R accepte $f(w)$,
  alors on accepte.

- Contraposée du premier point.

□

II. Une seconde réduction.

On va démontrer que le langage $\bar{A}_{\mathsf{TM}}$ se réduit à

$$\bar{L} = \{\langle M_1, M_2\rangle \mid \mathscr{L}(M_1) \neq \mathscr{L}(M_2)\}.$$

On considère la fonction calculable

$$\begin{aligned} g : \quad \Sigma^\star &\longrightarrow \Sigma^\star \\ \langle N, w\rangle &\longmapsto \langle M_{w,N}, M_{\Sigma^\star}\rangle. \end{aligned}$$

et on a l'équivalence :

$$\begin{aligned} \langle M_{w,N}, M_{\Sigma^\star}\rangle \in \bar{L} &\iff \mathscr{L}\big(M_{w,N}\big) \neq \mathscr{L}(M_{\Sigma^\star}) = \Sigma^\star \\ &\iff \mathscr{L}\big(M_{w,N}\big) = \emptyset \\ &\iff N \text{ n'accepte pas } w \\ &\iff \langle N, w\rangle \in \bar{A}_{\mathsf{TM}}. \end{aligned}$$

Ainsi, parce que $\bar{A}_{\mathsf{TM}}$ n'est pas Turing-reconnaissable, que $\bar{L}$ n'est pas Turing-reconnaissable (lemme précédent). On en conclut que le langage L n'est pas co-Turing-reconnaissable.