Problème de correspondance de Post : un problème indécidable simple !

2024

Emil Post (1946) : un jeu de dominos

1 Définition

Le problème PCP est défini par :
- Entrée : 2 listes de mots $u_1, u_2, \ldots, u_n \in \Sigma^*$ et $v_1, v_2, \ldots, v_n \in \Sigma^*$
- Question : Existe-t-il une suite $i_1, i_2, \ldots, i_k \in \{1, 2, \ldots, n\}$ telle que $u_{i_1} u_{i_2} \ldots u_{i_k} = v_{i_1} v_{i_2} \ldots v_{i_k}$.

Exemples :
- $u_1 = \begin{array}{|c|}\hline \text{a} \\ \hline \text{baa} \\ \hline \end{array}$, $u_2 = \begin{array}{|c|}\hline \text{ab} \\ \hline \text{aa} \\ \hline \end{array}$, $u_3 = \begin{array}{|c|}\hline \text{bba} \\ \hline \text{bb} \\ \hline \end{array}$ accepté pour $i_1 i_2 i_3 i_4 = 3231$ par exemple.
- $\begin{array}{|c|}\hline \text{aba} \\ \hline \text{a} \\ \hline \end{array}, \begin{array}{|c|}\hline \text{b} \\ \hline \text{b} \\ \hline \end{array}, \begin{array}{|c|}\hline \text{a} \\ \hline \text{ababa} \\ \hline \end{array}, \begin{array}{|c|}\hline \text{ab} \\ \hline \text{b} \\ \hline \end{array}$ accepté pour $i_1 i_2 i_3 i_4 i_5 = 12321$ par exemple
- $\begin{array}{|c|}\hline \text{aaa} \\ \hline \text{aab} \\ \hline \end{array}, \begin{array}{|c|}\hline \text{baa} \\ \hline \text{a} \\ \hline \end{array}, \begin{array}{|c|}\hline \text{ab} \\ \hline \text{abb} \\ \hline \end{array}, \begin{array}{|c|}\hline \text{b} \\ \hline \text{aa} \\ \hline \end{array}$ accepté avec plus petite solution de taille 302.

On va montrer que PCP est indécidable en passant par un intermédiaire et en utilisant 2 réductions consécutives. On utilise ici un problème plus contraint que PCP.

2 PCPM (modifié)

On ajoute une contrainte à PCP en fixant la première tuile utilisée. L'idée est de faire une réduction de l'arrêt Turing à PCPM en vérifiant un historique de calcul Turing.

Définition 1. Le problème PCPM est défini par :
- Entrée : 2 listes de mots $u_1, u_2, \ldots, u_n \in \Sigma^*$ et $v_1, v_2, \ldots, v_n \in \Sigma^*$
- Question : Existe-t-il une suite $i_2, \ldots, i_k \in \{2, \ldots, n\}$ telle que $u_1 u_{i_2} \ldots u_{i_k} = v_1 v_{i_2} \ldots v_{i_k}$.

Théorème 1. *PCPM est indécidable.*

Démonstration. On va montrer que l'on peut réduire le langage L_K à L_{PCPM}. Pour cela on va construire un jeu de dominos capable de simuler du calcul Turing et tel que la solution à PCPM correspond à un calcul qui s'arrête.

Objectif : Pour une machine de Turing M et une entrée w, on construit un ensemble de dominos $Dom(M, w) = \left(\begin{array}{|c|}\hline u_i \\ \hline v_i \\ \hline \end{array} \right)_i$ tel que M s'arrête sur w si et seulement si il existe une solution de PCPM pour $Dom(M, w)$.

Principe : On va construire un jeu de dominos qui écrit la suite des configurations traversées par la machine M sur l'entrée w (la trace) tout en vérifiant que chaque configuration est bien l'image de la précédente. Il faut de plus garantir que la configuration initiale correspond à w. L'idée est de prendre un premier domino qui écrit la configuration initiale C_0 “en bas” et rien “en haut”. Les dominos suivants devront donc écrire C_0 “en haut”, tout en écrivant son image “en bas”. Ainsi, on conserve une configuration d'avance en bas et on vérifie au passage que le calcul Turing est correct. On poursuit la vérification jusqu'à obtenir une configuration terminale.

La preuve. On définit une fonction Dom qui à chaque instance (M, w) du problème de l'arrêt fait correspondre un jeu de dominos $Dom(M, w)$. Soit M une MT et $w \in \Sigma^*$.

On se donne un premier domino $\left[\dfrac{}{\#q_0w_0w_1\dots w_n}\right]$ où $\#$ est choisi hors de Γ et de Q, et représente le début du codage d'une configuration Turing. Ensuite, il faut $\left[\dfrac{\#}{\#}\right]$ pour marquer le passage à une nouvelle configuration. Puis on propage le calcul en représentant chaque transition Turing :

— $(x,q) \to (y,q',\downarrow)$ est représentée par $\left[\dfrac{qx}{q'y}\right]$
— $(x,q) \to (y,q',\leftarrow)$ est représentée par $\left[\dfrac{zqx}{q'zy}\right]$ pour tout $z \in \Gamma$
— $(x,q) \to (y,q',\to)$ est représentée par $\left[\dfrac{qx}{yq'}\right]$

On gère les cas particuliers aux bords de la configuration :

— $(x,q) \to (y,q',\leftarrow)$ est représentée par $\left[\dfrac{\#qx}{\#q'By}\right]$
— $(x,q) \to (y,q',\to)$ est représentée par $\left[\dfrac{qx\#}{yq'B\#}\right]$

Tout le reste de la configuration (lorsque la tête n'est pas visible) est recopié tel quel à l'aide des dominos $\left[\dfrac{x}{x}\right]$ pour chaque $x \in \Gamma$.

Enfin, lorsque le calcul termine (état terminal atteint), il faut compléter $Dom(M,w)$ pour terminer la partie de dominos : les états terminaux "mangent" les lettres avec les dominos $\left[\dfrac{xq_f}{q_f}\right]$ et $\left[\dfrac{q_fx}{q_f}\right]$ pour chaque $x \in \Gamma$ et chaque $q_f \in Q_f$. On ajoute $\left[\dfrac{q_f\#}{}\right]$ pour chaque $q_f \in Q_f$ qui servira à clore la partie de dominos.

On dit qu'une suite d'indice $i_1, i_2, \dots, i_k$ réalise un couple de mots $\left[\dfrac{u}{v}\right]$ lorsque $u = u_{i_1}u_{i_2}\dots u_{i_k}$ et $v = v_{i_1}v_{i_2}\dots v_{i_k}$. On note $m = cod(q,u,k) \in (Q+\Sigma)^*$ le codage d'une configutaion Turing où $m = m_0m_1\dots m_{|u|}$ et :

— $m_0m_1\dots m_{k-1} = u_0u_1\dots u_{k-1}$
— $m_k = q$ où la tête de la MT pointe sur u_k
— $m_{k+1}\dots m_{|u|} = u_ku_{k+1}\dots u_{|u|-1}$

En particulier le premier domino est $\left[\dfrac{}{\#cod(q_0,w,0)}\right]$.

On commence par montrer un premier lemme :

Lemme 1. *Lemme de simulation : Soit C une configuration pour M. Les 2 propositions suivantes sont équivalentes :*

1. *la règle de M est applicable ($\exists C', C \vdash C'$)*
2. *il existe une suite d'indices qui réalisent $\left[\dfrac{cod(C)}{v}\right]$ ou $\left[\dfrac{\#cod(C)}{\#v}\right]$ ou $\left[\dfrac{cod(C)\#}{v\#}\right]$*

Dans ce cas, $v = cod(C')$.

Démonstration. On note $m = cod(C) = a_0a_{1-1}\dots a_lqb_0b_1\dots b_p$.

2⇒1 (contraposée) : Si la règle de M n'est pas applicable, il n'existe pas de domino contenant qb_0 "en haut".

1⇒2 : Dans le cas contraire, un des cas suivants s'applique :

1. Si $m = qb_0b_1\dots b_p$ et la règle de la MT est $(q,b_0) \mapsto (q',x,\leftarrow)$ alors il existe une suite d'indices de dominos réalisant $\left[\dfrac{\#qb_0}{\#q'Bx}\right]\left[\dfrac{b_1}{b_1}\right]\dots\left[\dfrac{b_p}{b_p}\right]$ c'est à dire $\left[\dfrac{\#m}{\#cod(C')}\right]$ pour C' telle que $C \vdash C'$.
2. Si $m = a_0\dots a_lqb_0$ et la règle de la MT est $(q,b_0) \mapsto (q',x,\to)$ alors il existe une suite d'indices de dominos réalisant $\left[\dfrac{a_0}{a_0}\right]\dots\left[\dfrac{a_l}{a_l}\right]\left[\dfrac{qb_0\#}{xq'B\#}\right]$ c'est à dire $\left[\dfrac{m\#}{cod(C')\#}\right]$ pour C' telle que $C \vdash C'$.

3. Si $m = a_0 \dots a_l q b_0 b_1 \dots b_p$ et la règle de la MT est $(q, b_0) \mapsto (q', x, \leftarrow)$ alors il existe une suite d'indices de dominos réalisant $\begin{array}{|c|}\hline a_0 \\ \hline a_0 \\ \hline \end{array} \dots \begin{array}{|c|c|c|}\hline a_{l-1} & a_l q b_0 & b_1 \\ \hline a_{l-1} & q' a_l x & b_1 \\ \hline \end{array} \dots \begin{array}{|c|}\hline b_p \\ \hline b_p \\ \hline \end{array}$ c'est à dire $\begin{array}{|c|}\hline m \\ \hline cod(C') \\ \hline \end{array}$ pour C' telle que $C \vdash C'$.
4. Si $m = a_0 \dots a_l q b_0 b_1 \dots b_p$ et la règle de la MT est $(q, b_0) \mapsto (q', x, \downarrow)$ alors il existe une suite d'indices de dominos réalisant $\begin{array}{|c|}\hline a_0 \\ \hline a_0 \\ \hline \end{array} \dots \begin{array}{|c|c|c|}\hline a_l & q b_0 & b_1 \\ \hline a_l & q' x & b_1 \\ \hline \end{array} \dots \begin{array}{|c|}\hline b_p \\ \hline b_p \\ \hline \end{array}$ c'est à dire $\begin{array}{|c|}\hline m \\ \hline cod(C') \\ \hline \end{array}$ pour C' telle que $C \vdash C'$.
5. Si $m = a_0 \dots a_l q b_0 b_1 \dots b_p$ et la règle de la MT est $(q, b_0) \mapsto (q', x, \rightarrow)$ alors il existe une suite d'indices de dominos réalisant $\begin{array}{|c|}\hline a_0 \\ \hline a_0 \\ \hline \end{array} \dots \begin{array}{|c|c|c|}\hline a_l & q b_0 & b_1 \\ \hline a_l & x q' & b_1 \\ \hline \end{array} \dots \begin{array}{|c|}\hline b_p \\ \hline b_p \\ \hline \end{array}$ c'est à dire $\begin{array}{|c|}\hline m \\ \hline cod(C') \\ \hline \end{array}$ pour C' telle que $C \vdash C'$.

Cela conclut la démonstration du lemme. $\square$

Retour à la preuve du théorème.

$\Rightarrow$ On suppose ici que M s'arrête sur w. Donc il existe une suite de configurations $C_0 \vdash C_1 \vdash \dots \vdash C_t$ avec $C_0 = (q_0, w, 0)$ et $C_t = (q_f, w_t, z_t)$ une configuration terminale. On note :
- $m_i = cod(C_i)$ pour $0 \leq i \leq t$, en particulier $m_t = a_0 a_{1-1} \dots a_l q_f b_0 b_1 \dots b_p$
- $m_{t+j} = a_0 a_1 \dots a_{l-j-1} q_f b_0 b_1 \dots b_p$ pour $1 \leq j \leq l-1$ et $m_{t+l} = q_f b_0 b_1 \dots b_p$
- $m_{t+l+j} = q_f b_j b_{j+1} \dots b_p$ pour $1 \leq j \leq p$ et $m_{t+l+p+1} = q_f$

On utilise le lemme de simulation 1 et on ajoute des $\begin{array}{|c|}\hline \# \\ \hline \# \\ \hline \end{array}$ pour montrer qu'il existe une suite d'indices réalisant $\begin{array}{|c|}\hline \# m_0 \# \dots \# m_{t-1} \# \\ \hline \# m_0 \# \dots \# m_{t-1} \# m_t \# \\ \hline \end{array}$. On note en particulier que le cas 2 de la démonstration du lemme ne peut se produire directement après le cas 1, ce qui justifie qu'on n'a pas besoin du cas $\begin{array}{|c|}\hline \# cod(C) \# \\ \hline \# v \# \\ \hline \end{array}$ dans le lemme.

Pour $0 \leq j \leq l-1$, il existe une suite d'indices réalisant $\begin{array}{|c|}\hline a_0 \\ \hline a_0 \\ \hline \end{array} \dots \begin{array}{|c|c|c|}\hline a_{l-j-2} & a_{l-j-1} q_f & b_0 \\ \hline a_{l-j-2} & q_f & b_0 \\ \hline \end{array} \dots \begin{array}{|c|}\hline b_p \\ \hline b_p \\ \hline \end{array}$ c'est à dire $\begin{array}{|c|}\hline m_{t+j} \\ \hline m_{t+j+1} \\ \hline \end{array}$.

Pour $0 \leq j \leq p$, il existe une suite d'indices réalisant $\begin{array}{|c|c|}\hline q_f b_j & b_{j+1} \\ \hline q_f & b_{j+1} \\ \hline \end{array} \dots \begin{array}{|c|}\hline b_p \\ \hline b_p \\ \hline \end{array}$ c'est à dire $\begin{array}{|c|}\hline m_{t+l+j} \\ \hline m_{t+l+j+1} \\ \hline \end{array}$.

En juxtaposant toutes ces suites d'indices, on réalise :

$$\begin{array}{|c|c|c|c|}\hline \# m_0 \# \dots \# m_{t-1} \# & m_t & \# & m_{t+1} \\ \hline \# m_0 \# \dots \# m_{t-1} \# m_t \# & m_{t+1} & \# & m_{t+2} \\ \hline \end{array} \dots \begin{array}{|c|c|c|}\hline m_{t+l+p} & \# & q_f \# \\ \hline m_{t+l+p+1} & \# & \\ \hline \end{array}$$

C'est bien une solution de PCPM pour $Dom(M, w)$.

$\Leftarrow$ On suppose maintenant qu'il existe une solution à l'instance $Dom(M, w)$ de PCPM. Il existe donc une suite d'indices $i_1 = 1, i_2, \dots i_k$ qui réalise $\begin{array}{|c|}\hline Sol \\ \hline Sol \\ \hline \end{array}$ où $Sol = \# m_0 \# m_1 \# \dots \# m_l$ ou $Sol = \# m_0 \# m_1 \# \dots \# m_l \#$ en prenant tous les m_i dans $(Q \cup \Gamma)^*$ de sorte qu'ils ne contiennent pas de $\#$.

On montre successivement plusieurs résultats intermédiaires.

Fait 1 : Il existe $e \leq l$ tel que m_e contient un symbole $q_f \in Q_f$.

Démo 1 : Pour chaque domino $i > 1$, si u_i ne contient pas de q_f, $|u_i| \leq |v_i|$. Or, on a $|u_{i_1}| < |v_{i_1}|$ donc s'il n'existe pas de tel e, alors pour tout j, $|u_{i_j}| \leq |v_{i_j}|$, et la suite de dominos $(i_j)_j$ produit deux mots de longueurs différentes, et ne peut pas être une solution.

On note e le plus petit tel indice et on s'intéresse à $m_0 \dots m_e$. On remarque aussi que si u_i commence par $\#$, alors v_i aussi, donc nécessairement v_{i_2} commence par $\#$ et $m_0 = q_0 w$.

Fait 2 : Il existe un découpage de $i_2, \dots i_k$ en des suites d'indices qui réalisent $\begin{array}{|c|}\hline m_j \\ \hline m_{j+1} \\ \hline \end{array}$ ou $\begin{array}{|c|}\hline m_j \# \\ \hline m_{j+1} \# \\ \hline \end{array}$ ou $\begin{array}{|c|}\hline \# m_j \\ \hline \# m_{j+1} \\ \hline \end{array}$ pour chaque $0 \leq j \leq e-1$.

Démo 2 : Comme m_i avec $i < e$ ne contient pas de symbole de Q_f, chaque domino utilisé contenant un symbole $\#$ le contient en première ou en dernière position à la fois "en haut" et "en bas". On coupe la séquence d'indices avant chaque symbole $\#$ si possible et après sinon.

On note $i_1, i_2, \dots, i_q$ la suite d'indices réalisant $\begin{array}{|c|}\hline \# m_0 \# m_1 \# \dots \# m_{e-1} \\ \hline \# m_0 \# m_1 \# \dots \# m_e \\ \hline \end{array}$ ou $\begin{array}{|c|}\hline \# m_0 \# m_1 \# \dots \# m_{e-1} \# \\ \hline \# m_0 \# m_1 \# \dots \# m_e \# \\ \hline \end{array}$.

Fait 3 : Chaque m_j avec $j \leq e$ contient exactement un symbole de Q.
Démo 3 : Par récurrence. Pour $j = 0$, on a $m_0 = q_0 w$. Si la propriété est vraie pour j entre 0 et $e-1$, on voit que les dominos conservent tous le nombre de symboles dans Q donc la suite d'indices réalisant $\boxed{\frac{m_j}{m_{j+1}}}$ ou $\boxed{\frac{m_j\#}{m_{j+1}\#}}$ ou $\boxed{\frac{\#m_j}{\#m_{j+1}}}$ aussi (puisque q_f ne peut apparaître dans m_j). Par conséquent, la propriété est vraie pour $j+1$.
On note $m_j = cod(C_j)$ pour tout $0 \leq j \leq e$. Le Fait 3 suffit à prouver l'existence de la configuration C_j (un seul symbole d'état).

Des applications répétées du lemme de simulation 1 démontrent que pour tout $0 \leq j \leq e-1$, la règle de M est applicable à la configuration C_j et $(C_j) \vdash (C_{j+1})$.

Ce dernier résultat montre donc que les m_j codent des configurations C_j avec :
- C_0 la configuration initiale pour M sur l'entrée w
- $\forall j \leq e-1, C_j \vdash C_{j+1}$
- C_e une configuration terminale.

Ce qui signifie que M s'arrête sur l'entrée w.

Conclusion : La fonction Dom est totale calculable et de plus

$$\forall (M,w), (M,w) \in L_K \Leftrightarrow Dom(M,w) \in L_{PCPM}$$

C'est à dire $L_K \leq_m L_{PCPM}$. Comme L_K n'est pas récursif, d'après la proposition du cours, L_{PCPM} non plus.
□

3 Indécidabilité de PCP

Pour conclure, on fait une réduction de PCPM à PCP.

Théorème 2. *PCP est indécidable*

Démonstration. Il suffit de montrer que $L_{PCPM} \leq_m L_{PCP}$. Pour cela on construit une fonction f calculable qui à tout jeu de dominos A associe un jeu de dominos B tel que $A \in L_{PCPM} \Leftrightarrow B \in L_{PCP}$.

Pour tout mot $w = w_0 w_1 \dots w_k \in \Sigma^*$, on note :
- $g(w) = \#w_0\#w_1\#\dots\#w_k$ et $g(\epsilon) = \epsilon$
- $d(w) = w_0\#w_1\#\dots\#w_k\#$ et $d(\epsilon) = \epsilon$

On peut montrer aisément que pour tous mots $w_1, w_2, \dots, w_k$, on a
- $g(w_1)g(w_2)\dots g(w_k) = g(w_1 w_2 \dots w_k)$
- et $d(w_1)d(w_2)\dots d(w_k) = d(w_1 w_2 \dots w_k)$

On a aussi :

$$\text{Si } g(u)\# = \#d(v) \text{ alors } u = v.$$

Soit A un jeu de dominos $\left(\boxed{\frac{u_i}{v_i}}\right)_i$. On construit la réduction f qui à A associe un jeu de dominos B contenant :
- $\boxed{\frac{g(u_1)}{\#d(v_1)}}$
- $\boxed{\frac{g(u_i)}{d(v_i)}}$ pour chaque i
- et $\boxed{\frac{\#}{}}$

On montre successivement les points suivants :

1. Si $A \in L_{PCPM}$, alors $B \in L_{PCP}$
   Soit une solution pour A $\boxed{\frac{u_1}{v_1}}\boxed{\frac{u_{i_2}}{v_{i_2}}}\dots\boxed{\frac{u_{i_n}}{v_{i_n}}}$. Clairement $\boxed{\frac{g(u_1)}{\#d(v_1)}}\boxed{\frac{g(u_{i_2})}{d(v_{i_2})}}\dots\boxed{\frac{g(u_{i_n})}{d(v_{i_n})}}\boxed{\frac{\#}{}}$ est une solution pour B.

2. Toute solution pour B commence par $\begin{array}{|c|}\hline g(u_1) \\ \hline \#d(v_1) \\ \hline \end{array}$.

   Tous les dominos $\begin{array}{|c|}\hline g(u_i) \\ \hline d(v_i) \\ \hline \end{array}$ commencent par des lettres différentes ($\#$ et une lettre de Σ). Si une solution commence par un domino $\begin{array}{|c|}\hline \# \\ \hline \\ \hline \end{array}$, il faut poursuivre avec un autre domino dont le mot du bas commence par $\#$ et pas celui du haut, ce qui n'existe pas.

3. Toute solution pour B termine par $\begin{array}{|c|}\hline \# \\ \hline \\ \hline \end{array}$

   Tous les autres dominos ont des lettres terminales différentes en haut et en bas ($\#$ et une lettre de Σ).

4. S'il existe une solution pour B, alors il en existe une qui ne contient qu'une seule fois le domino $\begin{array}{|c|}\hline \# \\ \hline \\ \hline \end{array}$ (en dernière position) et qu'une seule fois le domino $\begin{array}{|c|}\hline g(u_1) \\ \hline \#d(v_1) \\ \hline \end{array}$ (en première position).

   Tous les mots $g(u_i)$ et $\#$ commencent par $\#$ donc si le domino $\begin{array}{|c|}\hline \# \\ \hline \\ \hline \end{array}$ n'est pas en dernière position, le mot construit contient $\#\#$. Symétriquement, ce motif $\#\#$ ne peut apparaître qu'avec le domino $\begin{array}{|c|}\hline \# \\ \hline \\ \hline \end{array}$ pour le premier $\#$. De la même manière, ce motif ne peut apparaître en bas dans $\#d(v_1)d(v_{i_2})d(v_{i_3})\dots d(v_{i_k})$ qu'en réutilisant $\begin{array}{|c|}\hline g(u_1) \\ \hline \#d(v_1) \\ \hline \end{array}$ pour le second $\#$. En particulier, les préfixes lus jusqu'à la première apparition du motif $\#\#$ doivent coïncider donc $\left(\text{en coupant entre les deux } \#\text{, c'est à dire après } \begin{array}{|c|}\hline \# \\ \hline \\ \hline \end{array} \text{ et avant } \begin{array}{|c|}\hline g(u_1) \\ \hline \#d(v_1) \\ \hline \end{array}\right)$, on obtient $g(u_1)g(u_{i_2})\dots g(u_{i_n})\# = \#d(v_1)d(v_{i_2})\dots d(v_{i_n})$. C'est bien une solution pour B ne contenant qu'une fois $\begin{array}{|c|}\hline \# \\ \hline \\ \hline \end{array}$ (en dernière position) et qu'une fois le domino $\begin{array}{|c|}\hline g(u_1) \\ \hline \#d(v_1) \\ \hline \end{array}$ (en première position).

5. S'il existe une solution pour B, alors il existe une solution pour A

   Par les points précédents 2, 3 et 4, toute solution pour B commence par

   $$\begin{array}{|c|c|}\hline g(u_1) & g(u_{i_1}) \\ \hline \#d(v_1) & d(v_{i_1}) \\ \hline \end{array} \dots \begin{array}{|c|c|}\hline g(u_{i_n}) & \# \\ \hline d(v_{i_n}) & \\ \hline \end{array}$$

   Donc $g(u_1u_{i_1}\dots u_{i_n})\# = \#d(v_1v_{i_1}\dots v_{i_n})$, par conséquent $\begin{array}{|c|c|}\hline u_1 & u_{i_1} \\ \hline v_1 & v_{i_1} \\ \hline \end{array} \dots \begin{array}{|c|}\hline u_{i_n} \\ \hline v_{i_n} \\ \hline \end{array}$ est une solution pour A.

La fonction f est totale calculable. Au jeu de dominos A, elle associe le jeu de dominos $B = f(A)$ tel que

$$A \in L_{PCPM} \Leftrightarrow B \in L_{PCP}$$

C'est à dire $L_{PCPM} \leq_m L_{PCP}$. $\square$