Feuille d'exercices n°5.1 - Classes de complexité P et NP

Notions abordées

- modélisation, problèmes de décision
- problèmes NP
- réduction polynomiale d'un problème à un autre
- plusieurs problèmes NP-difficiles classiques

Exercice 1 : Problèmes de partitions

On décrit en français 4 problèmes de décisions NP-complets classiques.

SubsetSum Étant donnés n entiers $w_1, w_2, \ldots, w_n$, et un entier W, on se demande si on peut sélectionner une partie des w_i dont la somme est exactement W.

Partition Étant donnés n entiers $w_1, w_2, \ldots, w_n$, on se demande s'il est possible de les partitionner en deux ensembles de même somme♣.

Knapsack Étant donnés n objets de poids $p_1, p_2, \ldots, p_n$ et de valeurs $v_1, v_2, \ldots, v_n$ ainsi qu'un poids maximal P et une valeur objectif K, on se demande s'il est possible de trouver un sous-ensemble d'objets dont la somme des valeurs est au moins K, sans dépasser le poids P.

BinPacking Étant donnés n objets de taille $t_1, t_2, \ldots, t_n$, C la capacité des boîtes, et K un nombre maximum de boîtes, on se demande s'il est possible de ranger les n objets dans au plus K boîtes en respectant la contrainte de capacité.

Q. 1 Proposer une définition formelle des quatre problèmes de décision décrits ci-avant.

Solution

$$\textsc{SubsetSum} : \begin{cases} \textbf{Entrée :} \text{ Un entier } n \in \mathbb{N}, \text{ une suite finie } (w_i)_{i\in[\![1,n]\!]} \in \mathbb{N}^n, \text{ un entier } W \\ \textbf{Sortie :} \text{ Existe-t-il } I \subseteq [\![1,n]\!] \text{ tel que } \sum_{i\in I} w_i = W \text{ ?} \end{cases}$$

$$\textsc{Partition} : \begin{cases} \textbf{Entrée :} \text{ Un entier } n \in \mathbb{N}, \text{ une suite finie } (w_i)_{i\in[\![1,n]\!]} \in \mathbb{N}^n \\ \textbf{Sortie :} \text{ Existe-t-il } I \subseteq [\![1,n]\!] \text{ tel que } \sum_{i\in I} w_i = \sum_{i\notin I} w_i \text{ ?} \end{cases}$$

$$\textsc{Knapsack} : \begin{cases} \textbf{Entrée :} \text{ Un entier } n \in \mathbb{N}, \text{ deux suites finies } (p_i)_{i\in[\![1,n]\!]} \in \mathbb{N}^n \text{ et } (v_i)_{i\in[\![1,n]\!]} \in \mathbb{N}^n, \\ \qquad\qquad \text{un entier } P \in \mathbb{N} \text{ et un seuil } K \in \mathbb{N} \\ \textbf{Sortie :} \text{ Existe-t-il } I \subseteq [\![1,n]\!] \text{ tel que } \sum_{i\in I} p_i \leqslant P \text{ et } \sum_{i\in I} v_i \geqslant K \text{ ?} \end{cases}$$

$$\textsc{BinPacking} : \begin{cases} \textbf{Entrée :} \text{ Un entier } n \in \mathbb{N}, (t_i)_{i\in[\![1,n]\!]} \in \mathbb{N}^n, C \in \mathbb{N}^* \text{ et un seuil } K \in \mathbb{N} \\ \textbf{Sortie :} \text{ Existe-t-il } \varphi : [\![1,n]\!] \to [\![1,K]\!] \text{ telle que } \forall i \in [\![1,K]\!], \sum_{j\in\varphi^{-1}(\{i\})} t_j \leqslant C \text{ ?} \end{cases}$$

♣. On note que cette somme est alors nécessairement la moitié de la somme totale des w_i.

Q. 2 On dit de manière informelle qu'un problème Q **est un cas particulier** d'un autre problème R, ou que R est une généralisation de Q, lorsque Q se réduit "très simplement"[♣] au problème R. C'est par exemple le cas lorsque la fonction de réduction est l'identité. Montrer que :

a. le problème SubsetSum est un cas particulier du problème Knapsack,
b. le problème Partition est un cas particulier du problème Knapsack,
c. le problème Partition est un cas particulier du problème BinPacking.

Solution

a. Soit $w = (w_i)_{i \in [\![1,n]\!]} \in \mathbb{N}n$ une instance du problème SubsetSum.
Une entrée du problème Knapsack est constituée de deux suites finies de même taille et deux entiers. On définit à partir de w l'entrée de Knapsack suivante.

$$\begin{aligned} p_1, p_2, \dots, p_n &\overset{\text{déf}}{=} w_1, w_2, \dots, w_n \\ v_1, v_2, \dots, v_n &\overset{\text{déf}}{=} w_1, w_2, \dots, w_n \\ K &\overset{\text{déf}}{=} W \\ P &\overset{\text{déf}}{=} W \end{aligned}$$

On remarque alors que :

$$\begin{aligned} (p, v, K, P) \in \text{Knapsack}^+ &\Leftrightarrow \exists I \subseteq [\![1,n]\!], \sum_{i \in I} x_i \leqslant P \text{ et } \sum_{i \in I} v_i \geqslant K \\ &\Leftrightarrow \exists I \subseteq [\![1,n]\!], \sum_{i \in I} w_i \leqslant W \text{ et } \sum_{i \in I} w_i \geqslant W \\ &\Leftrightarrow \exists I \subseteq [\![1,n]\!], \sum_{i \in I} w_i = W \\ &\Leftrightarrow w \in \text{SubsetSum}^+ \end{aligned}$$

Le problème SubsetSum est donc un cas particulier du problème Knapsack.
De plus cette réduction est clairement calculable en temps polynomial.

b. Soit $w = (w_i)_{i \in [\![1,n]\!]} \in \mathbb{N}n$ une instance du problème Partition. On pose $S = \sum_{i=1}^{n} w_i$.
Une entrée du problème Knapsack est constituée de deux suites finies de même taille et deux entiers. On définit à partir de w l'entrée de Knapsack suivante.

$$\begin{aligned} p_1, p_2, \dots, p_n &\overset{\text{déf}}{=} w_1, w_2, \dots, w_n \\ v_1, v_2, \dots, v_n &\overset{\text{déf}}{=} w_1, w_2, \dots, w_n \\ P &\overset{\text{déf}}{=} \begin{cases} \frac{1}{2}S & \text{si } S \text{ est paire} \\ -1 & \text{sinon} \end{cases} \\ K &\overset{\text{déf}}{=} S \end{aligned}$$

On remarque alors que :

- Si S est impaire, nécessairement w est une instance négative de Partition, et (p, v, K, P) est une instance négative de Knapsack car la contrainte de poids avec $P = -1$ est insatisfiable vu que les p_i sont positifs.

[♣]. et donc en particulier en temps polynomial

- Sinon, *i.e.* si S est paire, on a $K = P = \frac{1}{2}S$, d'où les équivalences suivantes.

$$\begin{aligned}
(p, v, K, P) \in \textsc{Knapsack}^+ &\Leftrightarrow \exists I \subseteq [\![1, n]\!], \sum_{i \in I} x_i \leqslant P \text{ et } \sum_{i \in I} v_i \geqslant K \\
&\Leftrightarrow \exists I \subseteq [\![1, n]\!], \sum_{i \in I} w_i \leqslant \frac{1}{2}S \text{ et } \sum_{i \in I} w_i \geqslant \frac{1}{2}S \\
&\Leftrightarrow \exists I \subseteq [\![1, n]\!], \sum_{i \in I} w_i = \frac{1}{2}S \\
&\Leftrightarrow \exists I \subseteq [\![1, n]\!], \sum_{i \in I} w_i = \sum_{i \notin I} w_i \\
&\Leftrightarrow w \in \textsc{Partition}^+
\end{aligned}$$

Dans tous les cas on a $(p, v, K, P) \in \textsc{Knapsack}^+ \Leftrightarrow w \in \textsc{Partition}^+$, donc le problème Partition est un cas particulier du problème Knapsack.
De plus cette réduction est clairement calculable en temps polynomial.

c. Soit $w = (w_i)_{i \in [\![1,n]\!]} \in \mathbb{N}n$ une instance du problème Partition. On pose $S = \sum_{i=1}^{n} w_i$.
Une entrée du problème BinPacking est constituée de deux entiers C et K et d'une suite finie (t_i) d'entiers. On définit à partir de w l'entrée de BinPacking suivante.

$$\begin{aligned}
t_1, t_2, \dots, t_n &\overset{\text{déf}}{=} w_1, w_2, \dots, w_n \\
C &\overset{\text{déf}}{=} \frac{1}{2}S \\
K &\overset{\text{déf}}{=} 2
\end{aligned}$$

On remarque alors que $S = \sum_{j \in [\![1,n]\!]} t_j$:

$$\begin{aligned}
(t, C, k) \in \textsc{BinPacking}^+ &\Leftrightarrow \exists \varphi \in [\![1, K]\!]^{[\![1,n]\!]}, \forall i \in [\![1, K]\!], \sum_{j \in \varphi^{-1}(i)} t_j \leqslant C \\
&\Leftrightarrow \exists \varphi \in \{1, 2\}^{[\![1,n]\!]}, \forall i \in [\![1, 2]\!], \sum_{j \in \varphi^{-1}(i)} t_j \leqslant \frac{1}{2}S \\
&\Leftrightarrow \exists I \subseteq [\![1, n]\!], \sum_{j \in I} t_j \leqslant \frac{1}{2}S \text{ et } \sum_{j \notin I} t_j \leqslant \frac{1}{2}S \\
&\Leftrightarrow \exists I \subseteq [\![1, n]\!], \sum_{j \in I} t_j = \frac{1}{2}S \text{ et } \sum_{j \notin I} t_j = \frac{1}{2}S \\
&\Leftrightarrow \exists I \subseteq [\![1, n]\!], \sum_{j \in I} t_j = \sum_{j \notin I} t_j \\
&\Leftrightarrow w \in \textsc{Partition}^+
\end{aligned}$$

Le problème Partition est donc un cas particulier du problème Knapsack.
De plus cette réduction est clairement calculable en temps polynomial.

Q. 3 Montrer que SubsetSum $\preccurlyeq_P$ Partition.

Solution

Soit $\big((w_i)_{i \in [\![1,n]\!]}, W\big)$ une entrée de SubsetSum. On pose $S = \sum_{i=1}^{n} w_i$.
Un entrée du problème Partition est une suite finie d'entiers. On définit à partir de w l'entrée

de Partition suivante.

$$\begin{aligned} w'_1, w'_2, \dots, w'_n, , w'_{n+2} &\overset{\text{déf}}{=} w_1, w_2, \dots, w_n \\ w'_{n+1} &\overset{\text{déf}}{=} 2S - W \\ w'_{n+2} &\overset{\text{déf}}{=} S + w \end{aligned}$$

On remarque alors que $\sum\limits_{i=1}^{n+2} w'_i = 4S$, ainsi $I \subseteq [\![1, n+2]\!]$ est une solution de cette instance de Partition si $\sum\limits_{i\in I} = \sum\limits_{i\notin I} = 2S$. En particulier on a :

- si $n+1 \notin I$ et $n+2 \notin I$ alors $\sum_{i\notin I} w'_i \geqslant 3S > 2S$, donc I n'est pas solution.
- si $n+1 \in I$ et $n+2 \in I$ alors $\sum_{i\in I} w'_i \geqslant 3S > 2S$, donc I n'est pas solution.

Finalement :

$$\begin{aligned} ((w_i)_{i\in[\![1,n]\!]}, W) \in \textsc{SubsetSum} &\Leftrightarrow \exists I' \subseteq [\![1, n+2]\!], \sum_{i\in I'} w'_i = \sum_{i\notin I'} w'_i \\ &\Leftrightarrow \exists I' \subseteq [\![1, n+2]\!], \sum_{i\in I'} w'_i = 2S \\ &\Leftrightarrow \exists I' \subseteq [\![1, n+2]\!], \sum_{i\in I'} w'_i = 2S \text{ et } (n+1 \in I' \text{ et } n+2 \notin I') \\ &\qquad \text{ou } \sum_{i\in I'} w'_i = 2S \text{ et } (n+1 \in I' \text{ et } n+2 \notin I') \\ &\Leftrightarrow \exists I \subseteq [\![1, n]\!], \sum_{i\in I} w_i = 2S - 2S + W = W \\ &\qquad \text{ou } \sum_{i\in I} w_i = 2S - S - W = S - W \\ &\Leftrightarrow \exists I \subseteq [\![1, n]\!], \sum_{i\in I} w_i = W \end{aligned}$$

Finalement le problème SubsetSum se réduit au problème Partition et la réduction est clairement polynomiale.
On peut aussi définir l'instance suivante :

$$\begin{aligned} w'_1, w'_2, \dots, w'_n, , w'_{n+2} &\overset{\text{déf}}{=} w_1, w_2, \dots, w_n \\ w'_{n+1} &\overset{\text{déf}}{=} S \\ w'_{n+2} &\overset{\text{déf}}{=} 2W \end{aligned}$$

Dans ce cas $\sum\limits_{i=1}^{n+2} w'_i = S+S+2W = 2(S+W)$, pour $I \subseteq [\![1, n+2]\!]$ une solution de cette instance on a nécessairement $(n+1, n+2) \in I \times I^C$ ou $(n+1, n+2) \in I^C \times I$. Cette affirmation permet de mener les mêmes équivalences que plus haut et de conclure de même.

Q. 4 On souhaite démontrer que les 4 problèmes SubsetSum, Partition, Knapsack et BinPacking sont NP-difficiles. Quelle réduction polynomiale permettrait d'obtenir ce résultat ?
On ne demande pas de construire cette réduction car c'est l'objet de la question suivante.

Solution

Il suffit d'exhiber une réduction polynomiale d'un problème qu'on sait être NP-difficile à SUBSETSUM par exemple de 3-SAT à SUBSETSUM, la NP-difficulté de PARTITION, KNAPSACK et BINPACKING s'en déduiront vue les réductions préalablement démontrées (et résumées ci-dessous).

$$\begin{tikzcd}
& \text{KNAPSACK} & & \text{BINPACKING} \\
\text{SUBSETSUM} \arrow[ur, "1.a"] \arrow[rr, "2"] & & \text{PARTITION} \arrow[ul, "1.b"'] \arrow[ur, "1.c"] &
\end{tikzcd}$$

Réduction polynomiale de 3-SAT à SUBSETSUM. On fixe une entrée du problème 3-SAT représentée par une famille de m 3-clauses $(c_j)_{j\in[\![0,m[\![}$ sur l'ensemble des variables propositionnelles $\mathcal{Q} = \{x_1, x_2, \ldots, x_n\}$. Pour $i \in [\![1,n]\!]$ et $j \in [\![0,m[\![$, on note $x_i \in c_j$ le fait que le littéral x_i apparaisse dans la clause c_j, et on note $\neg x_i \in c_j$ le fait que le littéral $\neg x_i$ apparaisse dans la clause c_j. On considère les entiers $(a_i)_{i\in[\![1,n]\!]}$ et $(b_i)_{i\in[\![1,n]\!]}$ définis comme suit.

$$a_i = 10^{m-1+i} + \sum_{j=0}^{m-1} \mathbb{1}_{x_i\in c_j} 10^j$$

$$b_i = 10^{m-1+i} + \sum_{j=0}^{m-1} \mathbb{1}_{\neg x_i\in c_j} 10^j$$

Q. 5 Pour l'instance de 3-SAT suivante : $(x_1 \vee \neg x_2 \vee x_4) \wedge (\neg x_1 \vee x_3 \vee \neg x_4) \wedge (x_2 \vee x_3 \vee \neg x_4)$, donner les valeurs de :

- n ;
- m ;
- $(a_i)_{i\in[\![1,n]\!]}$;
- $(b_i)_{i\in[\![1,n]\!]}$.

Afin de rendre apparent le rôle joué par les chiffres des nombres $(a_i)_{i\in[\![1,n]\!]}$ et $(b_i)_{i\in[\![1,n]\!]}$, on s'efforcera de représenter les nombres comme des tableaux de chiffres et de bien séparer les m chiffres de poids faibles, des n chiffres de poids forts.

Solution

$n = 4$, $m = 3$,

- $a_1 = \begin{array}{|c|c|c|c||c|c|c|}\hline 0&0&0&1&0&0&1\\\hline\end{array}$: x_1 est dans la clause c_1, pas dans les autres.
- $a_2 = \begin{array}{|c|c|c|c||c|c|c|}\hline 0&0&1&0&1&0&0\\\hline\end{array}$: x_2 est dans la clause c_3, pas dans les autres.
- $a_3 = \begin{array}{|c|c|c|c||c|c|c|}\hline 0&1&0&0&1&1&0\\\hline\end{array}$: x_3 est dans les clauses c_2 et c_3, pas dans c_1.
- $a_4 = \begin{array}{|c|c|c|c||c|c|c|}\hline 1&0&0&0&0&0&0\\\hline\end{array}$: x_4 est dans la clause c_1.
- $b_1 = \begin{array}{|c|c|c|c||c|c|c|}\hline 0&0&0&1&0&1&0\\\hline\end{array}$: $\neg x_1$ est dans la clause c_2, pas dans les autres.
- $b_2 = \begin{array}{|c|c|c|c||c|c|c|}\hline 0&0&1&0&0&0&1\\\hline\end{array}$: $\neg x_2$ est dans la clause c_1, pas dans les autres.
- $b_3 = \begin{array}{|c|c|c|c||c|c|c|}\hline 0&1&0&0&0&0&0\\\hline\end{array}$: $\neg x_3$ n'est dans aucune clause.
- $b_4 = \begin{array}{|c|c|c|c||c|c|c|}\hline 1&0&0&0&1&1&0\\\hline\end{array}$: $\neg x_4$ est dans les clauses c_2 et c_3, pas dans c_1.

Dans la suite de cet exercice, pour $(k,j) \in \mathbb{N}^2$ nous noterons la notation $[k]_j$ le j-ème chiffre de k, de sorte que pour tout $k \in \mathbb{N}$, $k = \sum_{i=0}^{+\infty} [k]_j 10^j$.

Q. 6 Démontrer que le calcul de $\sum_{i=1}^n a_i + \sum_{i=1}^n b_i$ ne conduit pas à une retenue.

Solution

- Soit $j \in [\![0, m-1]\!]$, par définition des (a_i) et (b_i) : $[a_i]_j = \mathbb{1}_{x_i\in c_j}$ et $[b_i]_j = \mathbb{1}_{\neg x_i\in c_j}$. Or, pour tout $j \in [\![0, m-1]\!]$ la clause c_j contient exactement 3 littéraux, ainsi $\sum_{i=1}^n [a_i]_j + \sum_{i=1}^n [b_i]_j = 3$. Finalement les m chiffres de poids faibles de la somme $\sum_{i=1}^n a_i + \sum_{i=1}^n b_i$ ne conduisent pas à une retenue.
- Soit $j \in [\![m, m+n-1]\!]$, par définition des (a_i) et (b_i) : $[a_i]_j = \delta_{m-1+i,j} = [b_i]_j$, ainsi

$\sum_{i=1}^n [a_i]_j + \sum_{i=1}^n [b_i]_j = 2$. Finalement les n chiffres de poids forts de la somme $\sum_{i=1}^n a_i + \sum_{i=1}^n b_i$ ne conduisent pas à une retenue.

Q. 7 Montrer que s'il existe $\rho \in \mathbb{B}^{\mathcal{Q}}$ qui est un modèle de toutes les clauses $(c_j)_{j\in[\![0,m-1]\!]}$, alors il existe deux sous-ensembles A et B partitionnant $[\![1,n]\!]$ tels que $\sum_{i\in A} a_i + \sum_{i\in B} b_i$ est un entier dont l'écriture en base 10 est de la forme $w_{m+n-1}w_{m+n-2}\dots w_m w_{m-1}\dots w_1 w_0$ avec $\forall j \in [\![0,m[\![, w_j \in \{1,2,3\}$ et $\forall i \in [\![1,n]\!], w_{m-1+i} = 1$.

Solution

Soit $\rho \in \mathbb{B}^n$ un modèle de toutes les clauses $(c_j)_{j\in[\![0,m-1]\!]}$. On pose $A = \{i \in [\![1,n]\!] \mid \rho(x_i) = \mathsf{V}\}$ et $B = \{i \in [\![1,n]\!] \mid \rho(x_i) = \mathsf{F}\}$. Montrons que de tels ensembles conviennent.

- $A \sqcup B = [\![1,n]\!]$
- Soit $j \in [\![0,m-1]\!]$, $[\sum_{i\in A}^n a_i + \sum_{i\in B}^n b_i]_j = \sum_{i\in A}[a_i]_j + \sum_{i\in B}[b_i]_j \leqslant 3$ de la question précédente. Par ailleurs ρ est un modèle des clauses $(c_j)_{j\in[\![0,m-1]\!]}$ ainsi il existe une variable propositionnelle x_i de c_j telle que $x_i \in c_j$ et $\rho(x_i) = \mathsf{V}$ ou $\neg x_i \in c_j$ et $\rho(x_i) = \mathsf{F}$. Dans le premier cas $[a_i]_j = 1$ et $i \in A$, dans le second $[b_i]_j = 1$ et $i \in B$, d'où $\sum_{i\in A}[a_i]_j + \sum_{i\in B}[b_i]_j \geqslant 1$.
- Soit $j \in [\![m, m+n-1]\!]$, $[\sum_{i\in A}^n a_i + \sum_{i\in B}^n b_i]_j = \sum_{i\in A}[a_i]_j + \sum_{i\in B}[b_i]_j = \sum_{i\in A}\delta_{m-1+i,j} + \sum_{i\in B}\delta_{m-1+i,j} = \mathbb{1}_{i\in A} + \mathbb{1}_{i\in B} = 1$.

Q. 8 Proposer deux familles d'entiers $(d_i)_{i\in[\![0,m[\![}$ et $(e_i)_{i\in[\![0,m[\![}$ telles qu'il existe un modèle de toutes les clauses $(c_j)_{j\in[\![0,m-1]\!]}$ si et seulement s'il existe une suite extraite de $(a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_n, d_0, d_1, \dots, d_{m-1}, e_0, e_1, \dots, e_{m-1})$ dont la somme des termes vaut l'entier dont l'écriture décimale est $\underbrace{11\dots1}_{n}\underbrace{33\dots3}_{m}$.

Solution

D'après la question précédente, les sommes "bien choisies" de a_i et b_i conduisent à des chiffres de poids faibles de valeurs dans $\{1,2,3\}$. Ainsi afin de permettre "d'atteindre" des chiffres 3 on fournit des entiers $d_j = 10^j$ pour $j \in [\![0, m-1]\!]$ et $e_j = d_j$. Dans notre exemple :

- $d_1 = \begin{array}{|c|c|c|c||c|c|c|}\hline 0&0&0&0&0&0&1\\\hline\end{array}$
- $e_1 = \begin{array}{|c|c|c|c||c|c|c|}\hline 0&0&0&0&0&0&1\\\hline\end{array}$
- $d_2 = \begin{array}{|c|c|c|c||c|c|c|}\hline 0&0&0&0&0&1&0\\\hline\end{array}$
- $e_2 = \begin{array}{|c|c|c|c||c|c|c|}\hline 0&0&0&0&0&1&0\\\hline\end{array}$
- $d_3 = \begin{array}{|c|c|c|c||c|c|c|}\hline 0&0&0&0&1&0&0\\\hline\end{array}$
- $e_3 = \begin{array}{|c|c|c|c||c|c|c|}\hline 0&0&0&0&1&0&0\\\hline\end{array}$

Remarquons tout d'abord que le résultat de la question 6 est toujours vérifié malgré l'ajout des (d_i) et (e_i).

Montrons alors les deux sens de l'équivalence demandée.

$\Rightarrow$ Soit ρ un modèle de chacune des clauses $(c_j)_{j\in[\![0,m-1]\!]}$. Soit A et B tels que définis dans la question précédente. Notons alors $S = \sum_{i\in A} a_i + \sum_{i\in B} b_i$. D'après la question précédente : pour tout $j \in [\![0, m-1]\!]$, $[S]_j \in [\![1,3]\!]$ et pour tout $j \in [\![m, m+n-1]\!]$, $[S]_j = 1$. Soit alors $D = \{j \in [\![0,m-1]\!] \mid [S]_j \in \{1,2\}\}$ et $E = \{j \in [\![0,m-1]\!] \mid [S]_j = 1\}$. Finalement en notant $T = \sum_{i\in A} a_i + \sum_{i\in B} b_i + \sum_{i\in D} d_i + \sum_{i\in E} e_i$:

- pour tout $j \in [\![0, m-1]\!]$, $[T]_j = [S]_j + \mathbb{1}_{S_j\in\{1,2\}} + \mathbb{1}_{S_j=1} = 3$;
- pour tout $j \in [\![m, m+n-1]\!]$, $[T]_j = [S]_j = 1$.

D'où le résultat avancé.

$\Leftarrow$ Soient A, B, D, E des sous-ensembles de $[\![1,n]\!]$ (pour A et B) et $[\![1,m]\!]$ (pour D et E) tels que $T \overset{\text{déf}}{=} \sum_{i\in A} a_i + \sum_{i\in B} b_i + \sum_{i\in D} d_i + \sum_{i\in E} e_i = \underbrace{11\dots1}_{n}\underbrace{33\dots3}_{m}$. Soit $j \in [\![m, m+n-1]\!]$, par construction des (c_i) et (d_i) : $[T]_j = \sum_{i\in A}[a_i]_j + \sum_{i\in B}[b_i]_j = \mathbb{1}_{i\in A} + \mathbb{1}_{i\in B} = 1$. Ainsi $A \sqcup B = [\![1, n]\!]$. Soit alors l'environnement propositionnel ρ défini pour tout $i \in [\![1, n]\!]$ par $\rho(x_i) = \mathsf{V}$ si et seulement si $i \in A$. Soit alors $j \in [\![0, m-1]\!]$, par construction $\sum_{i\in D}[d_i]_j + \sum_{i\in E}[e_i]_j \leqslant 2$, ainsi $\sum_{i\in A}[a_i]_j + \sum_{i\in B}[b_i]_j \geqslant 1$, par disjonction de cas :

- Si il existe $i \in A$ tel que $[a_i]_j = 1$ alors $x_i \in c_j$, et par définition de ρ, $\rho(x_i) = \mathsf{V}$, ainsi $[\![c_j]\!]^\rho = \mathsf{V}$.
- Si il existe $i \in B$ tel que $[a_i]_j = 1$ alors $\neg x_i \in c_j$, et par définition de ρ, $\rho(x_i) = \mathsf{F}$, ainsi $[\![c_j]\!]^\rho = \mathsf{V}$.

Finalement $[\![c_j]\!]^\rho = \mathsf{V}$, et ceci étant vrai pour tout $j \in [\![0, m-1]\!]$ on en déduit que ρ est un modèle de toutes les clauses $(c_j)_{j\in[\![0,m-1]\!]}$.

Q. 9 Conclure quant à la NP-difficulté des problèmes.

Solution

La fonction f associant, à une instance $(c_i)_{i\in[\![1,m]\!]}$ de 3-SAT, la famille d'entiers $((a_i)_{i\in[\![1,n]\!]}, (d_i)_{i\in[\![1,n]\!]}, (c_i)_{i\in[\![1,m]\!]}, (d_i)_{i\in[\![1,m]\!]})$ est calculable en temps polynomial. Par ailleurs la question précédente nous assure que pour toute instance $(c_i)_{i\in[\![1,m]\!]}$ de 3-SAT : $(c_i)_{i\in[\![1,m]\!]} \in$ 3-SAT$^+$ $\Leftrightarrow f((c_i)_{i\in[\![1,m]\!]}) \in$ SubsetSum$^+$. Ainsi SubsetSum est np-difficile car 3-SAT l'est. D'après les réductions résumée à la **Q. 4**, les problèmes Partition, Knapsack et BinPacking le sont aussi.

Q. 10 Montrer que ces problèmes sont NP-complets.

Solution

Il nous reste à justifier que ces problèmes sont bien dans la classe NP. On le justifie en donnant pour chaque problème l'ensemble des certificats et un problème de vérification qui conviennent.

- SubsetSum. On choisit l'ensemble des certificats $\mathcal{C} = \bigcup \mathcal{C}_n$ où $\mathcal{C}_n = \mathcal{P}([\![1, n]\!])$. Les éléments de $\mathcal{C}_n$ admettent une représentation de taille polynomiale en n. On choisit alors le problème de vérification :

  Vérif : $\begin{cases} \textbf{Entrée :} \ I \in \mathcal{C}, \text{ un entier } n \in \mathbb{N}, \text{ une suite finie } (w_i)_{i\in[\![1,n]\!]} \in \mathbb{N}^n, \text{ un entier } W \\ \textbf{Sortie :} \ \sum_{i\in I} w_i = W\,? \end{cases}$

  Vérif $\in$ P.
- Partition. On choisit l'ensemble des certificats $\mathcal{C} = \bigcup \mathcal{C}_n$ où $\mathcal{C}_n = \mathcal{P}([\![1, n]\!])$. Les éléments de $\mathcal{C}_n$ admettent une représentation de taille polynomiale en n. On choisit alors le problème de vérification :

  Vérif : $\begin{cases} \textbf{Entrée :} \ I \in \mathcal{C}, \text{ un entier } n \in \mathbb{N}, \text{ une suite finie } (w_i)_{i\in[\![1,n]\!]} \in \mathbb{N}^n \\ \textbf{Sortie :} \ \sum_{i\in I} w_i = \sum_{i\notin I} w_i\,? \end{cases}$

  Vérif $\in$ P.
- Knapsack. On choisit l'ensemble des certificats $\mathcal{C} = \bigcup \mathcal{C}_n$ où $\mathcal{C}_n = \mathcal{P}([\![1, n]\!])$. Les éléments de $\mathcal{C}_n$ admettent une représentation de taille polynomiale en n. On choisit alors le problème de vérification :

  Vérif : $\begin{cases} \textbf{Entrée :} \ I \in \mathcal{C}, \text{ un entier } n \in \mathbb{N}, \text{ deux suites finies } (p_i)_{i\in[\![1,n]\!]} \in \mathbb{N}^n \text{ et } \\ \qquad (v_i)_{i\in[\![1,n]\!]} \in \mathbb{N}^n, \text{ un entier } P \in \mathbb{N} \text{ et un seuil } K \in \mathbb{N} \\ \textbf{Sortie :} \ \sum_{i\in I} p_i \leqslant P \text{ et } \sum_{i\in I} v_i \geqslant K\,? \end{cases}$.

  Vérif $\in$ P.

- BinPacking. On choisit l'ensemble des certificats $\mathcal{C} = \bigcup \mathcal{C}_n$ où $\mathcal{C}_n = [\![1, n]\!]^{[\![1,n]\!]}$. Les éléments de $\mathcal{C}_n$ admettent une représentation de taille polynomiale en n. On choisit alors le problème de vérification :

$$\text{VÉRIF} : \begin{cases} \textbf{Entrée :} \ \varphi \in \mathcal{C}, \text{ un entier } n \in \mathbb{N}, (t_i)_{i \in [\![1,n]\!]} \in \mathbb{N}^n, C \in \mathbb{N}^* \text{ et un seuil } K \in \mathbb{N} \\ \textbf{Sortie :} \ \forall i \in [\![1, K]\!], \sum_{j \in \varphi^{-1}(\{i\})} t_j \leqslant C\,? \end{cases}$$

VÉRIF $\in$ P.