Examen du cours Complexité (L3)

Date : 18 janv. 2018 / Durée : 2 heures

Partie 1

On considère des conjonctions de 3-clauses de la forme $\phi = \bigwedge_{i=1}^{m} C_i$ où chaque clause $C_i = \bigvee_{j=1}^{3} l_{i,j}$ est une disjonction d'*exactement* 3 littéraux, chaque littéral étant une variable booléenne x ou sa négation $\neg x$. On notera $\mathit{Var}(\phi)$ pour l'ensemble $\{x_1, \ldots, x_k\}$ des variables apparaissant dans ϕ. Sans être plus précis, on sait que la taille $n = |\phi|$ de ces formules est en $m^{O(1)}$.

Par ailleurs, le sujet utilise parfois des formules booléennes avec quantificateurs : il s'agit exactement des formules considérées pour le problème QBF —aussi appelé QSAT— vu en cours.

1A. Alors que 3SAT est le problème de savoir s'il existe une valuation $v : \mathit{Var}(\phi) \to \{\mathbf{true}, \mathbf{false}\}$ qui valide une formule ϕ, le problème 1-in-3SATconsiste à décider, étant donné ϕ, s'il existe une valuation v qui valide *exactement* un littéral par 3-clause de ϕ.

1. Est-ce que $x \vee y \vee \neg x$ est dans 1-in-3SAT ? Quelles valuations conviennent ?

   Solution:
   Oui. Toute valuation telle que $v(y) = \mathbf{false}$ convient, et seulement celles là.

2. Expliquez ce que veut dire « 1-in-3SAT $\subseteq$ 3SAT » et justifiez cette inclusion.

   Solution:
   Il s'agit d'une inclusion de langages : si $x \in$ 1-in-3SAT est une instance positive, alors $x = \phi$ est une conjonction de 3-clauses et il existe une valuation v qui valide exactement un littéral de chaque clause de x. Donc en particulier, v valide x, établissant $x \in$ 3SAT.

3. Montrez que le problème 1-in-3SAT est dans NP.

   Solution:
   La valuation v est un témoin de taille polynomiale (en n) qu'on peut donc deviner en NP. Une fois que v est deviné, tester s'il valide bien exactement un littéral par clause est une simple boucle logspace donc en temps polynomial.

On note $R(x, y, z)$ le prédicat qui est vrai si et seulement si exactement une des trois propositions x, y, z est vraie.

4. Démontrez que la formule $x \vee y \vee z$ est équivalente à la formule existentiellement quantifiée :

$$\exists a, b, c, d, e, f : R(x, a, d) \wedge R(y, b, d) \wedge R(z, c, \mathbf{false}) \wedge R(a, b, e) \wedge R(c, d, f) .$$

   Solution:
   Soit v une valuation de $\{x, y, z\}$ et v' une extension de v sur $\{x, y, z, a, b, c, d, e, f\}$. Il faut montrer que (1) tout v validant $(\phi \equiv)\ x \vee y \vee z$ peut être étendu en un v' validant $(\psi \equiv)\ R(x, a, d) \wedge \cdots$, et réciproquement que (2) si $v(\phi) = \mathbf{false}$ alors $v'(\psi) = \mathbf{false}$ pour toute extension v' de v.

Supposons que $v(\phi) = \textbf{true}$ et $v(x) = v(y) = v(z) = \textbf{false}$. Alors $v'(c) = \textbf{true}$ pour valider le 3e atome de ψ, donc $v'(d) = v'(f) = \textbf{false}$ (5e atome) donc $v'(a) = v'(b) = \textbf{true}$ (1er et 2e littéraux) et donc le 4e atome de ψ n'est pas validé : on vient de prouver par l'absurde l'implication (2).

Pour prouver (1) on suppose $v(\phi) = \textbf{true}$ et on montre comment construire une extension v' en considérant plusieurs cas. D'abord on fixe $v'(c) = \neg v(z)$ pour valider le 3e atome. Si $v(x) = \textbf{true}$ on prend $v'(a) = v'(d) = \textbf{false}$ pour valider le 1e atome, puis $v'(b) = \neg v'(y)$ pour le 2e atome, et enfin $v'(e) = \neg v'(b)$ et $v'(f) = \neg v'(c)$ pour les deux derniers littéraux. Si $v(y) = \textbf{true}$ on fait une construction symétrique en partant de $v'(b) = v'(d) = \textbf{false}$. Le cas restant est $v(x) = v(y) = \textbf{false}$ et $v(z) = \textbf{true}$ (donc $v'(c) = \textbf{false}$) : on prend $v'(a) = v'(b) = v'(f) = \textbf{true}$ et $v'(d) = v'(e) = \textbf{false}$.

5. Montrez que la formule $R(x, y, \textbf{false})$ est équivalente à la formule existentiellement quantifiée : $\exists g : R(x, y, g) \wedge R(g, g, \neg g)$.

Solution:
$R(g, g, \neg g)$ équivaut à $g = \textbf{false}$, et donc $\exists g : R(x, y, g) \wedge g = \textbf{false}$ équivaut à $R(x, y, \textbf{false})$.

6. En vous servant des questions 4 et 5, construisez une réduction de 3SAT vers 1-in-3SAT. Validez sa correction. Déduisez-en la complexité de 1-in-3SAT.

Solution:
À $\phi = \bigwedge_i C_i$ on associe $\phi' = \bigwedge_i \bigwedge_{j=1}^{6} C'_{i,j}$. Si $C_i = l_{i,1} \vee l_{i,2} \vee l_{i,3}$, on lui associe une conjonction $C'_{i,1} \wedge \cdots \wedge C'_{i,6}$ de 6 clauses de la forme

$$(l_{i,1} \vee a_i \vee d_i) \wedge (l_{i,2} \vee b_i \vee d_i) \wedge (l_{i,3} \vee c_i \vee g_i) \wedge (g_i \vee g_i \vee \neg g_i) \wedge (a_i \vee b_i \vee e_i) \wedge (c_i \vee d_i \vee f_i).$$

On a donc remplacé C_i, c.-à-d. $l_{i,1} \vee l_{i,2} \vee l_{i,3}$ par la formule $R(l_{i,1}, a_i, d_i) \wedge \cdots$ de la question 4, en se passant de la constante **false** grâce au gadget de la question 5. Puis le prédicat R est remplacé par une disjonction. Finalement, ϕ' est bien une conjonction de 3-clauses ne contenant que des littéraux qu'on construit facilement en logspace.

Il reste à montrer la correction. Si ϕ est satisfaisable, disons par une valuation v, alors, pour chaque i, v peut être étendu sur $\{a_i, \ldots, g_i\}$ de façon à ce que chaque $C'_{i,j}$ soit validé et qui plus est de façon à ce que exactement un littéral par $C'_{i,j}$ soit validé (c'est ce qu'on a montré à la question 4). Puisque les 6 clauses $C'_{i,j}$ associées à C_i utilisent des variables booléennes $a_i, b_i, \ldots, g_i$ qui ne sont pas partagées avec les autres clauses de ϕ', ces extensions de v sont toutes compatibles et on peut donc satisfaire ϕ' en ne satisfaisant par ailleurs qu'un littéral par clause.

Réciproquement, si ϕ n'est pas satisfaisable et si on considère une valuation v quelconque des variables de ϕ', il existe une clause C_i telle que $v(C_i) = \textbf{false}$. Donc grâce à la question 4, on déduit que v ne peut pas valider $C'_{i,1} \wedge \cdots \wedge C'_{i,6}$ sans valider au moins 2 littéraux dans une même clause $C'_{i,j}$. Finalement, ϕ est une instance positive de 3SAT ssi ϕ' est une instance positive de 1-in-3SAT.

Cette réduction montre que 1-in-3SAT est NP-difficile puis 3SAT l'est. Avec la question 3 on a montré qu'il était NP-complet.

1B. Le problème distinct-3SAT, donné par une conjonction de m clauses de trois littéraux *distincts* (parmi k propositions ou leur négation), consiste à décider s'il existe une interprétation v des propositions telle que toute clause soit vraie.

7. Montrez que le problème distinct-3SAT est dans NP.

Solution:
Étant donnée une instance, il faut vérifier qu'elle a bien la forme requise (une conjonction de 3-clauses, chacune sans littéral répété) ce qui se fait en temps polynomial (et même en logspace). On peut ensuite décider la satisfaisabilité en NP, exactement comme pour 3SAT.

8. Est ce que l'inclusion distinct-3SAT $\subseteq$ 3SAT entre langages est vérifiée ?

Solution:
Oui. $x \in$ distinct-3SAT implique $x \in$ 3SAT.

9. Construisez une réduction de 3SAT vers distinct-3SAT. Validez sa correction. Déduisez-en la complexité de distinct-3SAT.
Indication : Pour une clause de la formule originelle qui contient deux littéraux distincts on construira deux clauses à trois littéraux distincts à l'aide d'une nouvelle variables booléenne. Pour une clause de la formule originelle qui contient un seul littéral on construira quatre clauses à trois littéraux distincts à l'aide de deux nouvelles variables booléennes.

Solution:
Soit $\phi = \bigwedge_i C_i$ une instance de 3SAT, on lui associe ϕ' obtenu en modifiant les clauses de ϕ. Soit donc une clause C_i. Si les littéraux de C_i sont distincts, on garde C_i. Si $C_i = l \vee l \vee l$ contient trois fois le même littéral, on remplace C_i par

$$(l \vee a_i \vee b_i) \wedge (l \vee a_i \vee \neg b_i) \wedge (l \vee \neg a_i \vee b_i) \wedge (l \vee \neg a_i \vee \neg b_i) .$$

Pour v donné il existe une clause, parmi ces quatre clauses, où v ne valide ni le 2e littéral (on choisit a_i ou $\neg a_i$ en fonction de $v(a_i)$) ni le 3e. Donc v valide ces quatre clauses ssi $v(l) = \mathbf{true}$.
Si $C_i = l \vee l \vee l'$ avec $l' \neq l$, c.-à-d. si C_i contient un littéral en double, on remplace C_i par

$$(l \vee l' \vee a_i) \wedge (l \vee l' \vee \neg a_i)$$

et l'argument de correction est similaire.
Cette réduction est logspace et montre que distinct-3SAT est NP-difficile, donc NP-complet.

1C. Le problème NAE-3SAT (NAE pour "Not all equal"), consiste à décider, étant donnée une conjonction ϕ de m clauses de trois littéraux *distincts* (parmi k variables booléennes ou leur négation), s'il existe une valuation v qui dans chacune des clauses valide au moins un littéral *et* en invalide au moins un. (On dit alors que v satisfait la condition NAE pour ϕ).

10. Montrez que si une formule admet une valuation satisfaisant la condition NAE alors elle en admet au moins deux (sauf exception dans un cas limite à préciser).

Solution:
Si une valuation v valide certains littéraux et en invalide d'autres, la valuation "opposée" $\bar{v}$, définie par $\bar{v}(x) = \neg v(x)$, va donc invalider les littéraux qui étaient validés par v, et valider ceux qui étaient invalidés. Finalement, si v satisfait la condition NAE pour ϕ, $\bar{v}$ la satisfait aussi. Maintenant $v \neq \bar{v}$ sauf si le domaine de v est vide, possible quand ϕ contient 0 clause et n'utilise aucune variable.

11. Quelles inclusions sont vérifiées entre NAE-3SAT et les problèmes mentionnés plus haut ?

Solution:
On a NAE-3SAT $\subseteq$ distinct-3SAT (et donc $\subseteq$ 3SAT) car la définition de NAE-3SAT exige qu'une instance respecte la contrainte "pas de littéral répété dans une clause". Il n'y a pas d'inclusion avec 1-in-3SAT : p.ex. $(x \vee \neg x \vee y) \wedge (x' \vee \neg x' \vee y')$ est dans NAE-3SAT$\setminus$1-in-3SAT, tandis que $x \vee x \vee \neg x$ est dans 1-in-3SAT $\setminus$ NAE-3SAT.

12. Montrez que le problème NAE-3SAT est dans NP.

Solution:
(Solution omise.)

13. On note $N(x, y, z)$ le prédicat qui est vrai ssi une au moins des trois propositions est vraie, et une au moins est fausse. Démontrez que la formule $x \vee y \vee z$ est équivalente à la formule $\exists a : N(x, y, a) \wedge N(z, \neg a, \mathbf{false})$.

Solution:

Notons $\psi \equiv N(x, y, a) \wedge N(z, \neg a, \mathbf{false})$. Supposons $v(x \vee y \vee z) = \mathbf{true}$. Si $v(x \vee y) = \mathbf{true}$ alors en étendant alors v par $v(a) = \mathbf{false}$ on obtient $v(\psi) = \mathbf{true}$, donc $v(\exists a : \psi) = \mathbf{true}$. Sinon on a $v(x) = v(y) = \mathbf{false}$ et $v(z) = \mathbf{true}$, on étend par $v(a) = \mathbf{true}$ et ici aussi $v(\psi) = \mathbf{true} = v(\exists a : \psi)$.

Réciproquement, supposons $v(\exists a : \psi) = \mathbf{true}$. Il y a deux cas : soit $v(a) = \mathbf{false}$, impliquant $v(N(x, y, \mathbf{false})) = \mathbf{true}$ (cf. 1ère clause de ψ) d'où $v(x \vee y) = \mathbf{true}$, soit $v(a) = \mathbf{true}$, impliquant $v(N(z, \mathbf{false}, \mathbf{false})) = \mathbf{true}$ (cf. 2ème clause de ψ) d'où $v(z) = \mathbf{true}$. Dans tous les case $v(x \vee y \vee z) = \mathbf{true}$.

14. Montrez que $\mathsf{NAE\text{-}3SAT}$ est NP-complet.

Solution:

Avec la question précédente, on pourrait réduire $\mathsf{3SAT}$ à $\mathsf{NAE\text{-}3SAT}$ à condition de savoir définir la constante **false** qui n'est pas un littéral autorisé dans nos 3-clauses. Mais, comme montré dans la question 10, on ne peut pas forcer $x = \mathbf{false}$ dans une instance de $\mathsf{NAE\text{-}3SAT}$.

On va donc utiliser une nouvelle variable et profiter de la dualité. Formellement, notre réduction r associe à une conjonction $\phi = \bigwedge_i C_i = \bigwedge_i \bigvee_{j=1}^{3} l_{i,j}$, la formule

$$r(\phi) \stackrel{\text{def}}{=} \bigwedge_i D_i = \bigwedge_i \big[(l_{i,1} \vee l_{i,2} \vee a_i) \wedge (l_{i,3} \vee \neg a_i \vee b)\big]$$

où b et les a_i sont des nouvelles variables.

(1) Montrons que si v satisfait la condition NAE sur $r(\phi)$ alors ϕ est satisfaisable :

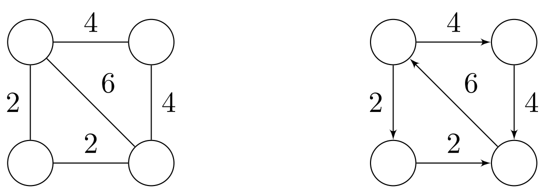
— Si $v(b) = \mathbf{false}$, alors on peut utiliser la question 13 et déduire, pour chaque i, que $v(C_i) = \mathbf{true}$ puisque v satisfait la condition NAE sur D_i. Donc $v(\phi) = \mathbf{true}$.

— Si $v(b) = \mathbf{true}$ alors $\bar{v}$ satisfait la condition NAE sur chaque $D'_i \stackrel{\text{def}}{=} (\neg l_{i,1} \vee \neg l_{i,2} \vee \neg a_i) \wedge (\neg l_{i,3} \vee a_i \vee \mathbf{false})$, donc, toujours par la question 13, $\bar{v}\big(\bigwedge_i \bigvee_{j=1}^{3} \neg l_{i,j}\big) = \mathbf{true}$ et donc $v\big(\bigwedge_i \bigvee_{j=1}^{3} l_{i,j}\big) = \mathbf{true}$ comme pour la question 10. Dans les deux cas, ϕ est satisfaisable.

(2) Réciproquement, si ϕ est satisfaisable, disons par une valuation v, on peut étendre v avec $v(b) = \mathbf{false}$ et $v(a_i)$ choisi comme à la question 13 en fonction des $v(l_{i,j})$, de façon à ce que v vérifie la condition NAE sur $r(\phi)$. Ceci achève de montrer la correction de la réduction qui est clairement logspace. On déduit que $\mathsf{NAE\text{-}3SAT}$ est NP-dur, et alors NP-complet.

Partie 2

On considère un graphe non orienté $G = (V, E)$ où chaque arête e est dotée d'un débit (un entier naturel). Le problème de *commutation de flots* consiste à affecter une orientation à chaque arête de G telle que la somme des débits entrant en chaque sommet soit égale à la somme des débits sortant du sommet. Nous avons présenté ci-dessous, un tel graphe et une solution possible.



15. Dans ce problème on va s'intéresser au problème de décision associé : « existe-t-il une orientation des arêtes telle que ... » sans forcément calculer une solution. Montrez que le problème de commutation de flots est dans NP.

Solution:
On devine une orientation et on vérifie les sommes des débits à chaque sommet.

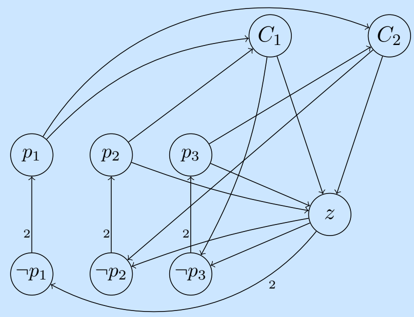
Soit $\phi = \bigwedge_{j=1}^{m} C_j$ une conjonction de m clauses de trois littéraux distincts. Les variables booléennes de ϕ sont notées $p_1, \ldots, p_n$, et on note $C_j = l_{j,1} \vee l_{j,2} \vee l_{j,3}$. On note α_i (resp. β_i) le nombre d'occurrences de p_i (resp. $\neg p_i$) dans les clauses. Sans perte de généralité on suppose que $\alpha_i + \beta_i > 0$ pour tout $i = 1, \ldots, n$.

On construit le graphe G_ϕ suivant :
— l'ensemble des sommets V est donné par $V = \{p_i, \neg p_i \mid 1 \leq i \leq n\} \cup \{C_j \mid 1 \leq j \leq m\} \cup \{z\}$;
— l'ensemble des arêtes E est défini par $E = L \cup C \cup D$, où :
— $L = \{\{p_i, \neg p_i\} \mid 1 \leq i \leq n\}$. Le débit de l'arête $\{p_i, \neg p_i\}$ est $\alpha_i + \beta_i$.
— $C = \{\{C_j, l_{j,a}\} \mid 1 \leq j \leq m \wedge 1 \leq a \leq 3\}$. Le débit de ces arêtes est 1.
— $D = \{\{z, C_j\} \mid 1 \leq j \leq m\} \cup \{\{z, p_i\} \mid 1 \leq i \leq n \wedge \beta_i > 0\} \cup \{\{z, \neg p_i\} \mid 1 \leq i \leq n \wedge \alpha_i > 0\}$. Le débit de l'arête $\{z, C_j\}$ est 1. Le débit de l'arête $\{z, p_i\}$ est β_i. Le débit de l'arête $\{z, \neg p_i\}$ est α_i.

16. Construisez G_ϕ pour $\phi = \{p_1 \vee p_2 \vee \neg p_3, p_1 \vee \neg p_2 \vee p_3\}$. Montrez que ce graphe admet une commutation de flots.

Solution:

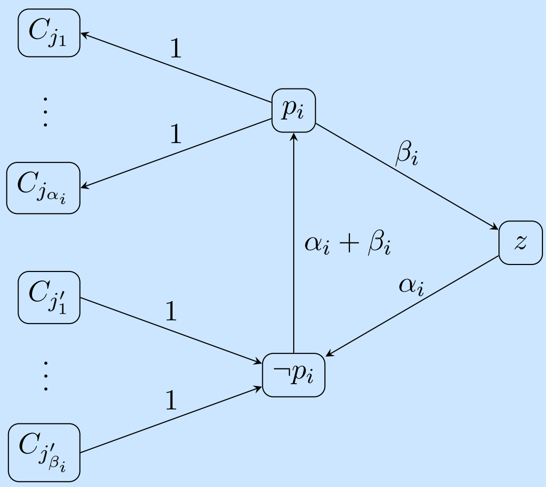
On donne à la fois le graphe et son orientation solution.
Pour alléger la figure on n'indique pas les poids valant 1.
Notons que ϕ donne lieu à $\alpha_1 = 2$, $\beta_1 = 0$, et $\alpha_2 = \beta_2 = 1 = \alpha_3 = \beta_3$.



17. Montrez que s'il existe une valuation v satisfaisant la condition NAE pour ϕ alors G_ϕ admet une commutation de flots. On vérifiera en particulier la contrainte de flots sur les sommets C_j et sur le sommet z.

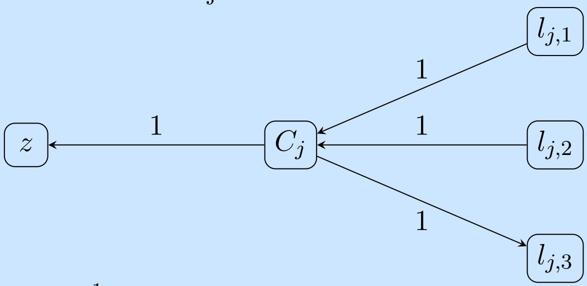
Solution:
Soit v une NAE-valuation de ϕ. On construit une commutation de flots pour G_ϕ comme suit :

— Pour tout $1 \leq i \leq n$, si $v(p_i) = \textbf{true}$ alors posons $C_{j_1}, \ldots, C_{j_{\alpha_i}}$ les clauses où le littéral p_i apparaît, et $C_{j'_1}, \ldots, C_{j'_{\beta_i}}$ les clauses où le littéral $\neg p_i$ apparaît.
Alors on pose (et on vérifie que la différence des débits entrant et sortant vaut 0 pour p_i et $\neg p_i$) :



— On traite symétriquement le $v(p_i) = \textbf{false}$.

— v est une NAE-valuation de ϕ, donc pour tout $1 \leq j \leq m$, v valide 2 littéraux ou un seul dans chaque clause. Supposons que pour C_j on ait $v(l_{j,1}) = v(l_{j,2}) = \textbf{true}$ et $v(l_{j,3}) = \textbf{false}$ (les autres cas sont similaires).
On oriente les arêtes de C_j comme suit, en vérifiant bien que notre orientation est compatible avec les orientations déjà données.



On a donc choisi que $C_j \xrightarrow{1} z$ ssi C_j a deux littéraux qui s'évaluent positivement.

— Toutes les arêtes sont orientées, il reste simplement à vérifier que z est équilibré. Posons $I^+ = \{i \mid v(p_i) = \textbf{true}\}$ et $I^- = \{i \mid v(p_i) = \textbf{false}\}$. On pose aussi $J^\top$ l'ensemble des $1 \leq j \leq m$ tel que C_j a deux littéraux qui s'évaluent positivement, et $J^\perp$ l'ensemble des $1 \leq j \leq m$ tel que C_j a deux littéraux qui s'évaluent négativement ($\{J^\top, J^\perp\}$ partitionnent J car v est une NAE-valuation).
En z, p_i et $\neg p_i$ contribue a une différence de débit $\beta_i - \alpha_i$ si $i \in I^+$ et $\alpha_i - \beta_i$ si $i \in I^-$. De même, C_j contribue a une différence de 1 si $j \in J^\top$ et -1 si $j \in J^\perp$. La différence des débits en z vaut alors :

$$\sum_{i \in I^+} (\beta_i - \alpha_i) + \sum_{i \in I^-} (\alpha_i - \beta_i) + |J^\top| - |J^\perp|$$

On fait maintenant des additions de booléens en posant $\textbf{true} = 1$ et $\textbf{false} = -1$:

$$\begin{aligned}
|J^\top| - |J^\perp| &= \sum_{j \in J^\top} (1 + 1 - 1) + \sum_{j \in J^\perp} (1 - 1 - 1) \\
&= \sum_{j \in J^\top} v(l_{j,1}) + v(l_{j,2}) + v(l_{j,3}) + \sum_{j \in J^\perp} v(l_{j,1}) + v(l_{j,2}) + v(l_{j,3}) \\
&= \sum_{j \in J} v(l_{j,1}) + v(l_{j,2}) + v(l_{j,3}) \\
&= \sum_{i \in I^+} \alpha_i - \sum_{i \in I^+} \beta_i - \sum_{i \in I^-} \alpha_i + \sum_{i \in I^-} \beta_i = -\Big[\sum_{i \in I^+} (\beta_i - \alpha_i) + \sum_{i \in I^-} (\alpha_i - \beta_i) \Big] .
\end{aligned}$$

Donc z est bien équilibré.

18. Toujours dans G_ϕ, on considère l'ensemble des arêtes adjacentes aux deux sommets p_i et $\neg p_i$. Montrez qu'il n'y a que deux orientations possibles de cet ensemble d'arêtes qui puissent apparaître dans une solution au problème de commutation de flots.

Solution:
p_i a une arête de poids $\alpha_i + \beta_i$ (l'arête $\{p_i, \neg p_i\}$), et la somme des poids des autres arêtes adjacentes à p_i vaut $\alpha_i + \beta_i$.
Par conséquent, si p_i est équilibré alors l'orientation de $\{p_i, \neg p_i\}$ est opposé à l'orientation de toutes les autres arêtes. On a donc bien seulement deux orientations possibles.

19. Montrez que si G_ϕ admet une commutation de flots alors ϕ est une instance positive du problème NAE-3SAT. Indiquez la valuation appropriée. Déduisez-en que le problème de commutation de flots est NP-complet.

Solution:

Supposons une commutation de flots pour G_ϕ et posons $v(p_i) = \textbf{true}$ ssi $\neg p_i \overset{\alpha_i+\beta_i}{\rightarrow} p_i$. Alors, d'après la question précédente, il n'y a qu'une orientation possible pour les arêtes adjacentes à p_i et $\neg p_i$, et ce pour tout $1 \leq i \leq n$. De plus cette orientation est celle de notre réponse à la question 17.

Supposons que v n'est pas une NAE-valuation. Soit j tel que $v(l_{j,1}) = v(l_{j,2}) = v(l_{j,3}) =$ **false** ou $v(l_{j,1}) = v(l_{j,2}) = v(l_{j,3}) =$ **true**. On traite le cas **true**, l'autre cas est identique. On a alors $l_{j,1} \overset{1}{\rightarrow} C_j$, $l_{j,2} \overset{1}{\rightarrow} C_j$ et $l_{j,3} \overset{1}{\rightarrow} C_j$. Cependant, C_j n'a que quatre arêtes adjacentes. Il n'y a donc aucun moyen d'équilibrer C_j. Absurde. Donc si G_ϕ admet une commutation de flots, ϕ est une instance positive de NAE-3SAT.

Réciproquement, si ϕ est une instance positive, G_ϕ admet une commutation (question 17). On a donc une réduction $\phi \mapsto G_\phi$ de 3SAT au problème de commutation de flots, déjà montré dans NP (question 15). On conclut en vérifiant que la réduction est logspace.

20. Montrez que le problème de commutation de flots est dans PTIME si les débits sont tous égaux à 1.

Solution:

Tout d'abord, remarquons que si $G_1, \ldots, G_n$ est la décomposition en composantes connexes de G alors G admet une commutation de flots ssi pour tout i, G_i admet une commutation de flots. Calculer les composantes connexes de G est faisable en temps polynomial, donc il suffit de donner un algorithme pour G connexe avec des débits de valeur 1.

Il est facile de vérifier qu'une orientation d'un graphe G connexe est une commutation de flots ssi elle vérifie les conditions d'Euler vues en cours (quand $s = t$). Donc G admet une commutation de flots ssi G admet un cycle Eulérien, ce qui se vérifie en temps polynomial.

21. Le problème de commutation de flots reste-t-il NP-complet si les débits sont exprimés en représentation unaire ? Justifiez votre réponse.

Solution:

Oui. Cette version reste NP-difficile : la réduction de la question 19 utilise des valeurs $\leq 3|\phi|$ pour les débits, donc il est possible de les écrire en base un dans une réduction logspace. (Et l'appartenance à NP reste valide a fortiori.)