

# THE COMPLEXITY OF PLANAR COUNTING PROBLEMS

ARTICLE : HARRY B. HUNT III, MADHAV V. MARATHE,  
VENKATESH RADHAKRISHNAN, RICHARD E. STEARNS  
RAPPORT : XAVIER PUJOL

## 1. INTRODUCTION

Le thème de cet article est l'étude de problèmes classiques "difficiles" à différents degrés, liés à 3SAT ou concernant des graphes, mais en se restreignant à des instances dites planaires. Pour les problèmes de graphes, la signification est claire. Pour les autres, on définit au cas par cas une association entre les instances du problème et des graphes, et on s'intéresse aux instances dont le graphe est planaire.

Il y a dans cet article un grand nombre de résultats. Voici son organisation générale (les différents problèmes sont définis au moment de leur étude dans le rapport) :

- Preuve de la #P-complétude et de la NP-complétude de PL-3SAT
- A partir du résultat précédent, preuve de la #P-complétude et de la NP-complétude de 1-VALID PL-3SAT, puis PL-EX3SAT, PL-1-EX3SAT, PL-1-EX3MONOSAT, et enfin X3C.
- #P-complétude des versions planaires de problèmes de graphes classiques.
- NP-complétude des problèmes d'ambiguïté et  $D^P$ -complétude des problèmes d'unicité associés à ces problèmes de graphes.
- En corollaire, inapproximabilité des instances planaires du problème de programmation linéaire en nombres entiers.

## 2. ETUDE DES VARIANTES DE 3SAT

### 2.1. Classe #P, #P-complétude.

**Définition.** On dit qu'un problème  $\Pi$  (modélisé formellement par une fonction  $\pi$  définie sur l'ensemble  $D_\Pi$  des instances) appartient à #P si et seulement s'il existe une machine de Turing  $M$  non déterministe fonctionnant en temps polynomial et telle que pour tout mot  $x \in D_\Pi$ ,  $\pi(x)$  soit égal aux nombres de calculs acceptants de  $M$  sur l'entrée  $x$ . Ces calculs acceptants sont appelés solutions du problème.

*Remarque.* On dit que  $\Pi$  est le problème de comptage associé au problème de décision  $D \in NP$  consistant à déterminer si  $x$  est accepté par  $M$ , c'est à dire s'il existe au moins une solution. On utilise la notation  $\Pi = \#D$ . Par exemple, 3SAT est le problème consistant à décider si une formule sous forme normale conjonctive avec des clauses de trois littéraux au plus possède une solution, et #3SAT est le problème consistant à déterminer combien d'assignements satisfont la formule.

**Définition.** Etant donnés deux problèmes  $\Pi$  et  $\Omega$  de #P modélisés par  $\pi$  et  $\omega$ , une réduction de  $\Pi$  à  $\Omega$  est une fonction  $f : D_\Pi \rightarrow D_\Omega$ . La réduction  $f$  est dite parcimonieuse si  $\forall x \in D_\Pi, \pi(x) = \omega(f(x))$ . Cela signifie qu'une instance de  $\Pi$  et l'instance de  $\Omega$  associée ont le même nombre de solutions.

La réduction  $f$  est dite faiblement parcimonieuse s'il existe une fonction  $g$  calculable en temps polynomial telle que  $\forall x \in D_{\Pi}, \pi(x) = g[x, \omega(f(x))]$ .

Dans la suite, une réduction parcimonieuse sera abrégée par p-réduction et une réduction faiblement parcimonieuse par fp-réduction.

**Définition.** On dit qu'un problème  $\Omega \in \#P$  est  $\#P$ -complet si tout problème  $\Pi \in \#P$  se fp-réduit à  $\Omega$ .

Pour montrer la  $\#P$ -complétude d'un problème  $\Omega$ , il suffit de montrer qu'un problème  $\Pi \in \#P$ -complet se fp-réduit à  $\Omega$ . Néanmoins, si le problème de décision associé à  $\Pi$  est aussi NP-complet, alors utiliser une p-réduction plutôt qu'une fp-réduction permet de montrer du même coup que le problème de décision associé à  $\Omega$  est NP-complet. Pour commencer, on se basera sur la  $\#P$ -complétude de  $\#3SAT$ .

*Remarque.* Dans l'article, la notion de fp-réduction est définie différemment. La condition est  $\forall x \in D_{\Pi}, \pi(x) = g(x)\omega(f(x))$ . Il me semble que c'est une erreur de frappe. En effet, il existe des problèmes  $\#P$ -complets auxquels sont associés des problèmes de décision solubles en temps polynomial (par exemple DNF, la satisfiabilité d'une formule sous forme normale disjonctive).  $3SAT$  pourrait être résolu en temps polynomial en utilisant la réduction de  $\#3SAT$  à l'un de ces problèmes. J'ai donc repris la définition donnée dans un autre article [3].

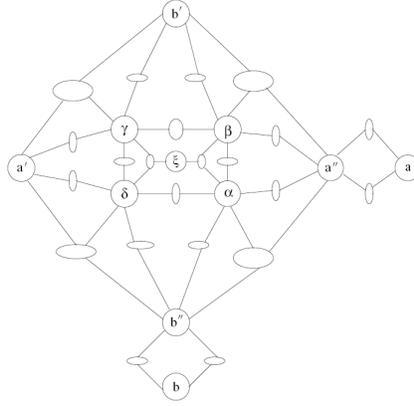
**2.2. Etude de PL-3SAT.** A partir d'une instance quelconque de  $3SAT$ , on construit le graphe biparti suivant : il y a un sommet pour chaque variable et un sommet pour chaque clause. Le sommet d'une clause est relié aux sommets des variables que la clause contient (le fait que les variables portent ou non une négation n'intervient pas dans la construction du graphe).

PL-3SAT est le problème de savoir si une instance de  $3SAT$  qui a un graphe planaire possède une solution.  $\#PL-3SAT$  est le problème de comptage associé. La NP-complétude de PL-3SAT a déjà été démontrée. L'apport de l'article est de remarquer que la réduction de  $3SAT$  à PL-3SAT proposée par un autre article [2] est en fait parcimonieuse, ce qui implique la  $\#P$ -complétude de  $\#PL-3SAT$ .

L'idée de la réduction est la suivante : on se donne une instance quelconque de  $3SAT$  et on souhaite construire une instance de PL-3SAT. On part de l'instance de  $3SAT$ , une formule  $f$ . Supposons que deux arêtes se croisent dans le graphe associé. Les deux arêtes relient deux variables  $a$  et  $b$  à deux clauses  $c_1$  et  $c_2$ . On introduit huit nouvelles variables  $a', b', a'', b'', \alpha, \beta, \gamma, \delta$ . Dans les clauses  $c_1$  et  $c_2$  de la formule  $f$ , les variables  $a$  et  $b$  sont remplacées par les variables  $a'$  et  $b'$ . On remplace la formule  $f$  par la formule

$$\begin{aligned} f' = & f \wedge (\overline{a''} \vee \overline{b''} \vee \alpha) \wedge (a'' \vee \overline{\alpha}) \wedge (b'' \vee \overline{\alpha}) \wedge (\overline{a''} + b' + \beta) \wedge (a'' \vee \overline{\beta}) \wedge (\overline{b'} \vee \overline{\beta}) \wedge \\ & (a' \vee b' \vee \gamma) \wedge (\overline{a'} \vee \overline{\gamma}) \wedge (\overline{b'} \vee \overline{\gamma}) \wedge (a' \vee \overline{b''} \vee \delta) \wedge (\overline{a'} \vee \overline{\delta}) \wedge (\alpha \vee \beta \vee \gamma \vee \delta) \wedge \\ & (\overline{\alpha} \vee \overline{\beta}) \wedge (\overline{\beta} \vee \overline{\gamma}) \wedge (\overline{\gamma} \vee \overline{\delta}) \wedge (\overline{\delta} \vee \overline{\alpha}) \wedge (a'' \vee \overline{\alpha}) \wedge (a \vee \overline{a''}) \wedge (b'' \vee \overline{b}) \wedge (b \vee \overline{b''}) \end{aligned}$$

Le graphe de cette formule est obtenu à partir du graphe de la formule précédente en remplaçant le croisement par le gadget suivant :



qui a la propriété d'être planaire. Dans ce gadget, les sommets non étiquetés correspondent aux clauses ajoutées à  $f$ . Les variables  $a'$  et  $b'$  sont reliées aux clauses  $c_1$  et  $c_2$ . La formule  $f'$  est telle qu'en définissant  $a$  et  $b$  on impose la valeur de toutes les autres variables ajoutées, en particulier  $a' = a$  et  $b' = b$ . Ceci justifie que la réduction est valide et parcimonieuse. Il suffit maintenant de répéter cette opération autant de fois que nécessaire pour obtenir une instance planaire.

**2.3. 1-VALID-PL-3SAT.** #1-VALID-PL-3SAT est le problème de comptage #3SAT restreint aux instances planaires et qui sont vraies lorsque toutes les variables sont à vrai. On va montrer que #1-VALID-PL-3SAT est #P-complet par fp-réduction à partir de #3SAT. Soit  $f$  une formule instance de 3SAT contenant les variables  $x_1, \dots, x_n$ . On considère la formule  $f' = (f \wedge \overline{x_{n+1}}) \vee (x_1 \wedge \dots \wedge x_n \wedge x_{n+1})$  où  $x_{n+1}$  est une nouvelle variable. Les assignements qui satisfont  $f'$  correspondent à ceux qui satisfont  $f$  avec  $x_{n+1} = faux$  ainsi que l'assignement de toutes les variables à vrai.  $f'$  a donc exactement une solution de plus que  $f$ . On peut mettre  $f'$  sous forme normale conjonctive en utilisant la distributivité de  $\vee$  par rapport à  $\wedge$ . Cela peut se faire en temps polynomial. Enfin, il faut passer d'une formule SAT à une formule 3SAT équivalente, ce qui peut se faire par une fp-réduction en temps polynomial puisque #3SAT est #P-complet donc #SAT s'y réduit. (L'article procède de façon un peu différente pour la réduction. Il utilise un théorème intermédiaire, et une méthode générale est mentionnée pour transformer  $f'$  en une formule 3SAT mais elle n'est pas vraiment détaillée. De toute façon c'est la forme particulière de  $f'$  qui justifie le fait que la réduction n'est pas exponentielle).

**2.4. Autres problèmes dérivés de 3SAT.** Dans cette partie, on étudie les versions planaires de variations de 3SAT. Plutôt que de faire des réductions de PL-3SAT aux versions planaires de ces problèmes, les auteurs ont fait des réductions des problèmes généraux mais en associant aux instances planaires du problème de départ des instances planaires du problème d'arrivée. Ainsi, les réductions sont valides à la fois pour les versions planaires et les versions générales (bien que les résultats soient déjà connus sur ces dernières). De plus, toutes les réductions sont des p-réductions, donc on prouve à la fois la #P-complétude des problèmes de comptage et la NP-complétude des problèmes de décision.

EX3SAT est le problème 3SAT restreint aux instances dont chaque clause contient exactement trois variables distinctes, #EX3SAT le problème de comptage associé. Dans un premier temps, on peut réduire 3SAT au problème où chaque clause contient au moins deux variables distinctes (si une clause contient une unique variable, on remplace simplement la variable par vrai dans toute la formule et on la retire, c'est bien une p-réduction). Dans

un second temps, il reste à éliminer les clauses à deux variables, ce qui se fait en créant une troisième variable pour la compléter et en ajoutant à la formule 8 autres variables et 14 clauses telles que :

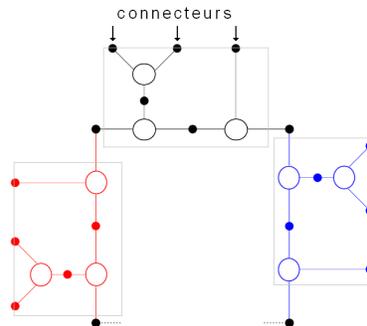
- Le seul assignement des 9 variables satisfaisant les 14 nouvelles clauses soit de toutes les mettre à faux (ainsi, la nouvelle formule a le même nombre de solutions que la précédente).
- Ce qu'on ajoute au graphe soit planaire.

La construction n'est pas détaillée ici. Le résultat est la #P-complétude de #PL-EX3SAT et la NP-complétude de PL-EX3SAT.

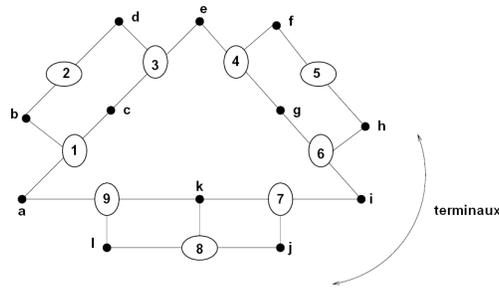
1-EX3SAT est le problème EX3SAT avec une contrainte supplémentaire pour qu'un assignement de variables soit solution : exactement un littéral sur les trois doit être à vrai dans chaque clause. Les auteurs font la démonstration par une réduction à partir de 3SAT, mais il me semble que partir de EX3SAT aurait évité l'examen du second cas (clause à deux variables). On part d'une instance de EX3SAT. Une clause  $a \vee b \vee c$  est remplacée par  $(a \vee u \vee v) \wedge (\bar{b} \vee u \vee w) \wedge (v \vee w \vee t) \wedge (\bar{c} \vee v \vee x)$  où  $t, u, v, w$  et  $x$  sont des nouvelles variables. On peut alors montrer que les deux formules ont le même nombre de solutions (avec EX3SAT pour la première et 1-EX3SAT pour la seconde) et que la construction est planaire d'où les résultats.

1-EX3MONOSAT est le problème 1-EX3SAT restreint aux instances qui ne contiennent aucune négation. La réduction se fait à partir de 1-EX3SAT. On prend une formule de 1-EX3SAT, pour chaque littéral de la forme  $\bar{z}$ , on introduit des nouvelles variables  $y, a, f, e, d$ , on remplace  $\bar{z}$  par  $y$  et on ajoute à la formule les quatre clauses  $(z \vee y \vee a) \wedge (a \vee d \vee e) \wedge (a \vee f \vee e) \wedge (d \vee f \vee e)$  (les trois dernières forcent  $a$  à faux, et la première force donc  $y$  à être la négation de  $z$ ).

X3C est le problème de "couverture exacte d'un ensemble par des ensembles de trois éléments". On se donne un ensemble  $E$  de cardinal  $3m$  et un ensemble  $P$  de parties de  $E$  à trois éléments (au moins  $m$ ). Le problème X3C est de déterminer si on peut trouver  $m$  parties dans  $P$  dont l'union est  $E$  en entier. Le graphe associé à une instance de X3C est le suivant : on crée un sommet pour chaque élément de  $E$  et chaque partie appartenant à  $P$ . Le sommet associé à une partie est lié aux sommets des éléments qu'elle contient. On réduit 1-EX3MONOSAT à X3C en raisonnant sur les graphes associés. On part du graphe d'une instance de 1-EX3MONOSAT. Chaque variable est remplacée par la construction suivante :



Le même motif - celui contenu dans un rectangle gris - est répété pour former un cycle de longueur égale au degré du sommet (c'est à dire la nombre de clauses dans lesquelles apparaît la variable). Chaque clause est remplacée par :



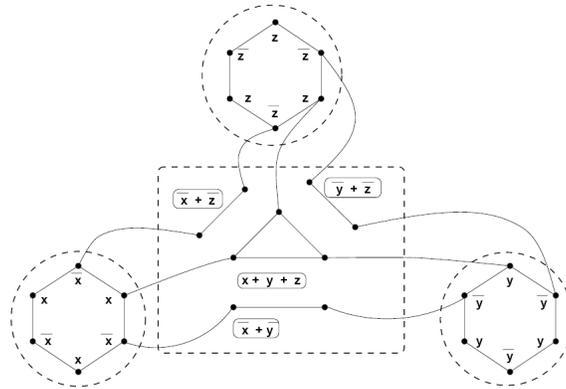
Lorsqu'il y a dans le graphe de départ une arête entre une variable et une clause, on fusionne trois connecteurs de la variable à trois terminaux de la clause. Dans le graphe obtenu, les points correspondent aux éléments de  $E$  du problème X3C, les cercles et ellipses représentent les parties de trois éléments.

Sans rentrer dans les détails, l'idée de la démonstration est de lier la valeur d'une variable dans 1-EX3MONOSAT au fait que les éléments issus de la fusion entre connecteurs et terminaux soient couverts par une partie appartenant au sous-graphe de la variable ou bien par une partie appartenant au sous-graphe de la clause.

Au final, on montre qu'on a une p-réduction de 1-EX3MONOSAT à X3C donc #X3C est #P-complet et X3C est NP-complet.

### 3. ETUDE DE PROBLÈMES DE GRAPHES

3.1. **Vertex cover.** Ce problème consiste à déterminer, étant donné un graphe et un entier  $K$ , s'il existe un ensemble de sommets de taille inférieure ou égale à  $K$  tel que toute arête soit adjacente à l'un des sommets de cet ensemble. On s'intéresse ici au problème de comptage associé. On montre qu'il est #P-complet par une réduction à partir de 1-EX3MONOSAT. Le principe est toujours le même. On part du graphe associé à une instance de 1-EX3MONOSAT.

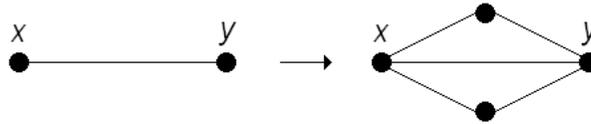


Les cycles alternant  $z$  et  $\bar{z}$  remplacent chaque variable  $z$  (ils sont de longueur égale au double du nombre de clause contenant  $z$ ). Les clauses  $x \vee y \vee z$  sont remplacées par la construction à l'intérieur du rectangle (expliquée dans l'article par une étape intermédiaire pour passer de 1-EX3MONOSAT à 3SAT). On pose  $K = 8m$  où  $m$  est le nombre de clauses de la formule de 1-EX3MONOSAT. On peut montrer facilement qu'une couverture par sommets contient au moins  $K$  sommets. La question est de savoir s'il en existe de taille  $K$ .

L'article renvoie à un autre article pour la preuve. L'idée pour mettre en correspondance les solutions des deux problèmes est de prendre dans la couverture tous les sommets étiquetés par  $x$  si  $x$  est vraie dans la solution de 1-EX3MONOSAT, et tous les sommets étiquetés par  $\bar{x}$  sinon. Cela laisse deux possibilités pour choisir les sommets dans chaque clause. Il y a ainsi  $2^m$  solutions de VERTEX COVER pour chaque solution de 1-EX3MONOSAT. Comme les problèmes sont de taille  $\Omega(m)$  et que  $2^m$  est de taille  $m$ , la réduction est bien une fp-réduction.

Dans l'article, les auteurs parlent du problème MINIMUM VERTEX COVER au lieu de VERTEX COVER. Le problème de comptage associé doit consister à compter les couvertures par sommets de taille minimale, ce qui ici revient au même puisque  $K$  est posé minimal. En conclusion, compter les solutions des instances planaires de l'un ou l'autre des problèmes est #P-complet.

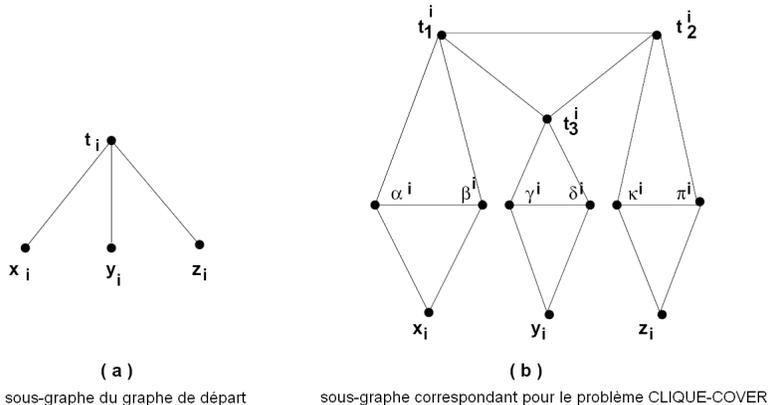
**3.2. Problèmes réduits à partir de VERTEX COVER.** DOMINATING SET est assez semblable à VERTEX COVER : il s'agit de déterminer, étant donné un graphe et un entier  $K \in \mathbb{N}$ , s'il existe un ensemble de sommets tel que chaque sommet est à une distance au plus 1 d'un élément de cet ensemble. La p-réduction planaire de VERTEX COVER à DOMINATING SET est très simple. On considère une instance de VERTEX COVER et on réalise sur chaque arête la transformation suivante :



Les solutions de VERTEX COVER sont en bijection directe avec celles de DOMINATING SET sur le nouveau graphe. DOMINATING SET restreint aux instances planaires est donc #P-complet.

Deux autres problèmes sont mentionnés mais leurs preuves ne sont pas détaillées.

**3.3. Problèmes réduits à partir de X3C.** Pour continuer la série, voici (MINIMUM) CLIQUE COVER : déterminer si on peut partitionner les sommets d'un graphe en moins de  $K$  sous-ensembles (ou un un nombre minimal de sous-ensembles) de sommets formant chacun une clique. On part du graphe de X3C et on effectue les transformations suivantes à partir du graphe de départ :



(a) sous-graphe du graphe de départ

(b) sous-graphe correspondant pour le problème CLIQUE-COVER

On obtient un graphe planaire si on part d'un graphe planaire, et les seules cliques de ce graphe sont de taille 3. La correspondance entre les solutions est la suivante : la partie  $t_i$  de trois éléments est incluse dans la couverture de l'ensemble si et seulement si la clique formée par  $t_1^i, t_2^i$  et  $t_3^i$  est incluse dans la partition des sommets du graphe. On a finalement un p-réduction de X3C à MINIMUM CLIQUE COVER.

PARTITION INTO CLAWS : déterminer si on peut partitionner les arêtes d'un graphe pour obtenir des sous-graphes de taille 4 isomorphes au graphe suivant (un "claw").



On réduit à partir de X3C. On prend une instance de X3C, on peut se réduire au cas où chaque élément apparaît dans 2 ou 3 parties. On considère dans le graphe associé le noeud correspondant à un élément. Il est de degré 2 ou 3. On ajoute au graphe une ou deux nouvelles arêtes partant de ce noeud et allant vers un ou deux nouveaux noeuds, pour que le noeud soit de degré 4 (si le graphe est planaire au départ, il le reste). A une solution de X3C correspond la partition de G en claws où chaque noeud d'un élément est le centre d'un claw. Etant de degré 4, une des arête partant du noeud ne fait pas partie du claw : elle est reliée au noeud du triplet couvrant l'élément.

Pour les problèmes suivants (BIPARTITE DOMINATING SET, PARTITION INTO TRIANGLES, MINIMUM HITTING SET), les preuves ne sont pas faites en détail. Les auteurs ajoutent finalement que d'après l'étude de 1-VALID-PL-3SAT, les problèmes restent #P-difficiles même si on se donne en entrée une instance et une solution du problème.

#### 4. PROBLÈMES D'AMBIGUÏTÉ ET D'UNICITÉ SUR LES GRAPHS

Le problème d'unicité associé à un problème de décision consiste à se demander si une instance du problème possède une unique solution. Le problème d'ambiguïté est le même mais sur une entrée composée d'une instance du problème et d'une solution (il reste à déterminer s'il y en a une autre).

**Définition.**  $D^P = \{L_1 - L_2 \mid L_1, L_2 \in NP\} = \{L_1 \cap (\Sigma^* \setminus L_2) \mid L_1, L_2 \in NP\}$

**Définition.** Une réduction temps polynomial probabiliste est une réduction par une machine de Turing probabiliste fonctionnant en temps polynomial (une machine de Turing probabiliste choisit à chaque étape une transition en fonction de son état et du caractère lu en suivant une distribution de probabilité fixée).

Les auteurs ont montré la  $D^P$ -complétude de UNIQUE-PL-3SAT selon une réduction en temps polynomial probabiliste en modifiant une preuve d'un autre article [1], mais sans donner les détails. En "utilisant des idées similaires" aux démonstrations de cet article, on peut montrer que AMBIGUOUS-PL-3SAT est NP-complet, AMBIGUOUS-1-VALID PL-3SAT est NP-complet et UNIQUE-1-VALID PL-3SAT est CoNP-complet. Les problèmes d'ambiguïté associés aux versions planaires de EX3SAT, 1-3SAT, 1-EX3SAT ou 1-EX3MONOSAT sont NP-complets et les problèmes sont  $D^P$ -complets pour la réduction en temps polynomial probabiliste.

Le dernier résultat de l'article est l'inapproximabilité du problème de programmation linéaire en nombres entiers en temps polynôme même pour les instances planaires (si  $P \neq NP$ ). Le principe de la démonstration est de montrer qu'obtenir une  $\varepsilon$ -approximation permet de résoudre 3SAT. On commence d'abord par se ramener à une formule qui a une solution en utilisant une technique similaire à celle de la preuve de 1-VALID PL-3SAT. Cette formule contient une variable de plus et il existe une solution où elle vaut 1 si et

seulement si la formule d'origine est satisfiable. On utilise les preuves de complétude des versions planaires pour se ramener à une formule planaire, puis à une instance de 1-EX3-PL-MONOSAT et enfin à un système de contraintes linéaires : pour une clause  $x \vee y \vee z$ , on pose la contrainte  $x + y + z \geq 1$  (il doit falloir ajouter également  $x + y + z \leq 1$  sinon en mettant toutes les variables à 1, on vérifie le système).

## 5. CONCLUSION

Cet article mentionne un très grand nombre de résultats, au point que beaucoup ne sont pas vraiment démontrés. Il suscite apparemment de l'intérêt puisque d'après une recherche rapide il a été cité 36 fois depuis sa parution en 1998. Il est assez difficile à synthétiser car il n'y a pas vraiment de résultat principal. Toutes les démonstrations sont basées les unes sur les autres et ont à peu près le même principe mais chaque problème demande une construction différente.

L'article est assez long, mais étant donné l'importance du contenu, les démonstrations sont souvent condensées et font référence à d'autres articles. La définition de la #P-complétude me pose problème mais à part cela je n'ai pas trouvé d'erreur qui mette en doute la validité d'un résultat.

## REFERENCES

- [1] L. G. Valiant and V. V. Vazirani, *NP is as easy as detecting unique solutions*, in Proc. 17th Annual ACM Symp. on Theory of Computing, 1985, pp. 458-463.
- [2] D. Lichtenstein, *Planar formulae and their uses*, SIAM J. Comput., 11 (1982), pp. 329-343.
- [3] N. Creignou and M. Hermann, *On #P-completeness of some counting problems*, research report 2144, INRIA, December 1993.