

Méthode Combinatoire pour le Calcul du Pfaffien

Romain Demangeon

L'article [4] présente un algorithme combinatoire pour calculer le Pfaffien d'une matrice anti symétrique. Les lemmes et les théorèmes sont nombreux, et utilisent des notions et des résultats non rappelés dans l'article.

Le Pfaffien est un opérateur directement lié aux couplages parfaits du graphe associé à une matrice antisymétrique.

Cet article prouve que le Pfaffien est dans NC en donnant un algorithme qui le calcule. Il montre aussi que le Pfaffien est $GapL$ complet.

Les auteurs s'intéressent ensuite au calcul du nombre de couplage parfait pour un graphe planaire.

On présente dans une première partie les notions de base nécessaires à la compréhension des concepts développés. On présentera ensuite le cheminement de l'article. On expliquera l'algorithme principal puis on s'intéressera au cas des graphes planaires.

1 Notions Préliminaires

1.1 Classes de problèmes

1.1.1 NC

Une suite de circuits $(C_n)_n$ est polynomiale uniforme s'il existe une machine de Turing travaillant en espace logarithmique qui produit C_n sur l'entrée 1^n . Un problème est dans $TUPS(f, g)$ s'il est calculé par une suite polynomiale uniforme telle que C_n est de taille $O(g(n))$ et de profondeur $O(f(n))$.

La classe NC est l'union des classes $TUPS(f, g)$ avec f puissance de \log et g polynôme.

La classe NC^i est l'union des classes $TUPS(\log^i(n), g)$ avec g polynôme.

RNC est la classe randomisée de NC qui sont les problèmes de décisions résolus par une machine NC tels que si la réponse est "oui", la moitié des chemins de calculs au moins sont acceptants, si la réponse est "non", ils sont tous rejetants.

1.1.2 $GapL$ et $\sharp L$

L est la classe déterministe en espace logarithmique.

$\sharp L$ est la classe de dénombrement pour L , c'est à dire la classe des fonctions qui comptent le nombre de chemins acceptant dans les machines de NL .

$GapL$ est la classe des fonctions qui comptent le nombre de chemins acceptants moins le nombre de chemins refusants d'une machine de NL . C'est donc la clôture par soustractions de $\sharp L$.

On peut aussi voir les fonctions de $\sharp L$ comme des fonctions qui calculent des poids de chemins entre une racine s et un but t sur un graphe pondéré acyclique. De la même manière, on peut voir les fonctions de $GapL$ comme des fonctions qui calculent la différence des poids des chemins entre la racine s et un but t_+ et des poids des chemins entre la racine s et un but t_- .

$GapL$ peut aussi être vue comme la classe des fonctions de L réduisibles à un calcul de déterminant entier.

1.2 Théorie des Graphes

Rappel: Un graphe *dirigé* est un graphe non-orienté auquel on a donné une orientation à chaque arête (c'est une notion différente de celle de graphe *orienté*).

1.2.1 Matrices d'adjacence

Un graphe pondéré orienté avec boucles $G = (V, E, w)$ peut être représenté par une matrice $M_G = (m_{ij})_{i,j}$. m_{ij} code le poids de l'arête $w(\langle v_i, v_j \rangle)$. M_G est appelée *Matrice d'adjacence de G*.

Si le graphe est non-orienté, dirigé et sans boucles, on peut prendre une matrice M_G antisymétrique pour le représenter. Dans ce cas, le coefficient M_{Gij} de la matrice vaut $w(\langle i, j \rangle)$ si $\langle i, j \rangle \in E$, $-w(\langle i, j \rangle)$ si $\langle j, i \rangle \in E$, 0 sinon. La matrice M_G est dite *Matrice de Tutte* du graphe G .

Réciproquement, à une matrice antisymétrique D on peut associer un unique graphe dirigé G tels que les arêtes soient orientées en avant: i.e. du plus petit sommet vers le plus grand, on appelle cette orientation: *orientation canonique* de G .

1.2.2 Planarité

Un graphe $G = (V, E)$ est dit planaire lorsqu'on peut représenter les sommets de ce graphe comme des points du plan réel $v_i = (x_i, y_i)$ et les arêtes $\langle v_i, v_j \rangle$ comme des lignes entre ces points (de longueur finie et infiniment dérivables) qui ne se croisent jamais.

Le théorème de Kuratowski nous apprend qu'un graphe est planaire si et seulement si il ne contient aucun sous-graphe homéomorphe à K_5 ou $K_{3,3}$.

$K_5 = (\{1, 2, 3, 4, 5\}, \{\langle 1, 2 \rangle, \langle 1, 3 \rangle, \langle 1, 4 \rangle, \langle 1, 5 \rangle, \langle 2, 3 \rangle, \langle 2, 4 \rangle, \langle 2, 5 \rangle, \langle 3, 4 \rangle, \langle 3, 5 \rangle, \langle 4, 5 \rangle, \})$ (la clique à 5 éléments)

$K_{3,3} = (\{1, 2, 3, 4, 5, 6\}, \{\langle 1, 4 \rangle, \langle 1, 5 \rangle, \langle 1, 6 \rangle, \langle 2, 4 \rangle, \langle 2, 5 \rangle, \langle 2, 6 \rangle, \langle 3, 4 \rangle, \langle 3, 5 \rangle, \langle 3, 6 \rangle, \})$ ("les 3 maisons et les 3 bornes")

1.2.3 Couplage

Le problème du couplage parfait est un problème classique d'algorithmique.

Un couplage d'un graphe pondéré $G = (V, E)$ est un sous-ensemble C de E dont les arêtes n'ont pas d'extrémités communes. Formellement $\forall (\langle v_i, v_j \rangle, \langle v_k, v_l \rangle) \in C^2, \text{Card}\{v_i, v_j, v_k, v_l\} = 4$.

Un couplage $C = (\langle v_i, v_j \rangle)_{i,j}$ est dit *parfait* si chaque sommet est extrémité d'une arête de C . Formellement $\forall v_i \in V, \exists v_j \in V, (\langle v_i, v_j \rangle \in E) \vee (\langle v_j, v_i \rangle \in E)$.

Le poids d'un couplage parfait est le produit des poids de ses arêtes.

On ne sait pas si le problème du calcul d'un couplage parfait est dans NC mais il est dans RNC .

Le problème de compter le nombre de couplages parfaits d'un graphe est dans $\#P$.

1.3 Algèbre

1.3.1 Matrices Antisymétriques

Une matrice carrée $A = (a_{ij})_{i,j}$ est antisymétrique si $A^T = -A$.

Cela implique que $\forall i, a_{ii} = 0$ et que $\forall (i, j), a_{ij} = -a_{ji}$. Une matrice antisymétrique $n \times n$ est donc définie par $\frac{n*(n-1)}{2}$ coefficients.

1.3.2 Permutations

S_n désigne ici l'ensemble des permutations de $\{1, \dots, n\}$, c'est à dire les bijections de $\{1, \dots, n\}$ dans $\{1, \dots, n\}$.

Un cycle $C(n_1, n_2, \dots, n_m)$, les n_i étant tous différents, est une permutation telle que $\forall k \notin (n_i)_i, C(k) = k$ et $\forall i \in \{1, \dots, m\}, C(n_i) = n_{i+1} \pmod{m}$. On appelle *support* de C l'ensemble $(n_i)_i$.

Une permutation se décompose de manière unique en produit de cycles de support disjoints.

Une permutation peut se décomposer en produit de transpositions.

Le signe d'une permutation $sgn(\sigma) \in \{-1, +1\}$ est la parité du nombre de transposition dans une décomposition en transposition de σ .

On peut considérer une permutation comme un n -uplet $(\sigma(1), \dots, \sigma(n))$. On les représentera souvent par la suite comme un $n/2$ -uplet de couples $((\sigma(1), \sigma(2)), \dots, (\sigma(n-1), \sigma(n)))$

1.3.3 Déterminant, permanent

On rappelle les définitions du permanent et du déterminant:

$$\text{per}(A) = \sum_{\sigma \in S_n} \prod_i a_{i, \sigma(i)}$$

$$\det(A) = \sum_{\sigma \in S_n} sgn(\sigma)$$

Le déterminant est dans NC

1.4 Pfaffiens

1.4.1 Définition

Le Pfaffien est un opérateur défini pour une matrice antisymétrique $D = (d_{ij})_{ij}$ $n * n$ et un graphe dirigé G .

Le poids d'une permutation $\sigma \in S_n$ est défini par:

$$wt(\sigma) = \prod_{i=1}^{n/2} d_{\sigma(2i-1)\sigma(2i)}$$

Si \mathcal{M} est un couplage parfait de G son terme Pfaffien $p(\mathcal{M})$ est défini comme $sgn(\sigma).wt(\sigma)$, où σ est n'importe quelle permutation telle que $\mathcal{M} = \{< \sigma(1), \sigma(2) >, < \sigma(3), \sigma(4) >, \dots, < \sigma(n-1), \sigma(n) >\}$. Le terme Pfaffien est bien défini car les permutations (il y en a $2^{n/2} \cdot (n/2)!$) qui conviennent ont toutes le même signe.

On associe à un couplage parfait \mathcal{M} une permutation canonique $\sigma_{\mathcal{M}}$, qui est l'unique permutation où chaque arête va du plus petit sommet vers le plus grand et où les arêtes sont classées par ordre croissant de plus petit sommet. Formellement, $\forall i < j, \sigma(2i-1) < \sigma(2j-1)$ et $\forall i, \sigma(2i-1) < \sigma(2i)$.

Par exemple, si $G_0 = (V_0, E_0)$ avec $V_0 = \{1, 2, 3, 4\}$, et $E_0 = \{< 1, 2 >, < 2, 3 >, < 4, 3 >, < 1, 4 >\}$, $\mathcal{M} = \{< 1, 2 >, < 3, 4 >\}$ constitue un couplage parfait. Id_4 et la transpositions (34)(21) conviennent comme permutations associées. Elles ont le même signe +1.

Le Pfaffien de la matrice D (ou de manière équivalente le Pfaffien du graphe dirigé pondéré G) est défini par:

$$Pf(D) = \sum_{\mathcal{M}} p(\mathcal{M}).$$

(La somme parcourt tous les couplages parfaits \mathcal{M} , on peut aussi parcourir tous les couplages parfaits possibles (ie toutes les partitions de $\{1, \dots, n\}$ en paires) car le terme Pfaffien associé à un couplage parfait possible qui n'est pas un couplage parfait du graphe est nul).

1.4.2 Exemple

Considérons G_0 , pondéré comme à la figure 1.

La matrice associée D est:

$$D = \begin{pmatrix} 0 & 2 & 0 & 1 \\ -2 & 0 & 1 & 0 \\ 0 & -1 & 0 & -2 \\ -1 & 0 & 2 & 0 \end{pmatrix}$$

Il y a 3 couplages parfaits possibles pour un graphe à 4 éléments. Leurs permutations canoniques sont $[(1, 2), (3, 4)]$, $[(1, 3), (2, 4)]$ et $[(1, 4), (2, 3)]$ de signes respectifs +1, -1 et +1.

Ainsi le Pfaffien de D (ou de G_0) vaut $2 * -2 - 0 * 0 + 1 * 1 = -3$.

1.4.3 Complexité

Notons que le Pfaffien d'une matrice $n * n$ est nul si n est impair (il n'y a pas de couplages parfait).

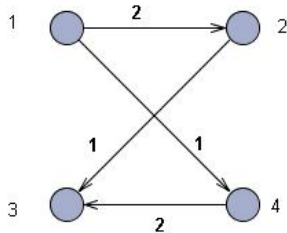


Figure 1: Le Graphe G_0 .

Si n est pair, on a $\det(D) = Pf(D)^2$. Cette égalité montre que calculer le Pfaffien au signe près est dans NC . [4] donne pour la première fois un algorithme NC pour calculer le Pfaffien avec son signe.

L'article [1] assure que le Pfaffien peut être calculé (avec son signe) en temps polynomial en expliquant, par contre, que les techniques qui permettent de calculer le déterminant dans NC ne se généralisent pas au Pfaffien.

Calculer le signe du Pfaffien n'est pas négligeable. Par exemple, l'algorithme présenté dans [3] pour chercher un couplage parfait sur un graphe biparti de genre borné utilise une combinaison linéaire de Pfaffiens.

Le Pfaffien peut, sous un certain angle, être vu comme "plus général" que le déterminant. Pour une matrice carrée A de taille $n * n$ avec $n \equiv 0(\text{mod}4) \vee n \equiv 1(\text{mod}4)$, on a:

$$\det(A) = Pf\left(\begin{pmatrix} 0 & A \\ -A^t & 0 \end{pmatrix}\right)$$

1.5 Programmation Dynamique

La programmation dynamique est une technique de programmation utilisable pour résoudre des problèmes qui se décomposent en sous-problèmes de telle façon que la sous-solution optimale d'un sous-problème est calculée à partir des sous-sous-solutions optimales des sous-sous-problèmes.

Si le nombre de sous-problèmes est petit (par exemple polynomial en la taille de l'entrée), une approche récursive conduit souvent à un temps de calcul surpolynomial, parce que certains sous-problèmes sont résolus plusieurs fois.

La programmation dynamique consiste en un calcul progressif des sous-solutions de tous les sous-problèmes en stockant les résultats pour les rendre directement accessibles pendant la résolution de sur-problèmes.

Les problèmes de la *Plus Longue Sous Suite Commune* et de l'*Arbre de Recherche Optimal* sont des exemples classiques de problèmes qui se prêtent à la programmation dynamique.

2 Notions et Théorèmes de l'article [4]

2.1 Les *clow*

Une *clow* est un chemin dans un graphe qui part d'un certain sommet (la tête), visite des sommets d'indice strictement plus grand un nombre quelconque de fois et revient au sommet de départ.

Soit $G = (V, E)$ un graphe $|V| = n$. Une *clow* (closed walk) est une suite d'arêtes $C = \langle e_1, e_2, \dots, e_m \rangle$ telle que $e_i = \langle v_i, v_{i+1} \rangle$, $e_m = \langle v_m, v_1 \rangle$, $\forall j, v_1 \neq v_j$ et $v_1 = \min\{v_1, \dots, v_m\}$.

v_1 est la tête de la *clow* $h(C)$, m sa longueur (notée aussi $|C|$). Le poids de la *clow* $wt(C)$ est $\prod_{i=1}^m wt(e_i)$.

Une suite de *clow* est un uple de *clow* (C_1, \dots, C_k) telle que $h(C_1) < \dots < h(C_k)$ et $\sum_{i=1}^k |C_i| = n$.

La dernière condition permet de s'assurer qu'on ne travaillera pas avec des *clows* de taille arbitraire.

2.2 Superposition de deux couplages parfaits

Soit F_1 et F_2 deux couplages parfaits de G . Leur superposition $F_1 \cup F_2$ est le graphe obtenu en incluant toutes les chemins fermés d'arêtes prises alternativement dans F_1 et F_2 .

On choisit un premier sommet v_1 , puis on prend l'unique arête de F_1 incidente à v_1 , son autre extrémité est v_2 . On prend ensuite l'unique arête de F_2 incidente à v_2 , etc. Quand on referme la boucle (c'est forcément en v_1), on choisit un autre sommet pas encore traité jusqu'à ce que tout le graphe soit traité. On dispose alors de chemin fermé dans $F_1 \cup F_2$, tous de longueur paire. Il y a 2 manières de parcourir chaque cycle, correspondant à 2 orientations du cycle. On appelle ces deux sens horaires et anti-horaires, le sens horaire étant celui où le premier sommet est le plus petit du cycle et où la première arête est dans F_1 .

Exemple: sur G_0 , si $F_1 = [(1, 2), (3, 4)]$ et $F_2 = [(1, 4), (2, 3)]$

Alors $F_1 \cup F_2$ contient un unique cycle dont l'orientation horaire est $\langle 1, 2 \rangle, \langle 2, 3 \rangle, \langle 3, 4 \rangle, \langle 4, 1 \rangle$

Si on dirige le graphe G , un cycle de $F_1 \cup F_2$, pris dans une certaine orientation, traverse des arêtes dans le même sens que leur orientation, et d'autres en sens inverse. On dit qu'une arête e est *correctement orientée* pour un cycle C d'une superposition de couplages (avec son orientation), si C traverse e selon son orientation.

Un cycle a une orientation paire selon G , si le nombre d'arêtes de C correctement orientées est pair. Dans le cas contraire, il a une orientation impaire.

La parité d'une orientation d'un cycle ne dépend pas de son sens de parcours. (Puisque le cycle contient un nombre pair d'arêtes).

Sur l'exemple précédent, en prenant les orientations de la figure 1, on avait un unique cycle $\langle 1, 2 \rangle, \langle 2, 3 \rangle, \langle 3, 4 \rangle, \langle 4, 1 \rangle$ dans le sens horaire, dont 2 arêtes sont correctement orientées: $\langle 1, 2 \rangle$ et $\langle 2, 3 \rangle$, son orientation est donc paire. Si on considère le sens antihoraire du cycle: $\langle 1, 4 \rangle, \langle 4, 3 \rangle, \langle 3, 2 \rangle, \langle 2, 1 \rangle$, 2 arêtes sont correctement orientées: $\langle 1, 4 \rangle$ et $\langle 4, 3 \rangle$.

2.3 Théorèmes sur le signe du Pfaffien

Lemme 1 *Le premier lemme démontré dans [4] assure que pour 2 couplages parfaits d'un même graphe G : $p(F_1).p(F_2) = (-1)^k wt(F_1).wt(F_2)$ où k est le nombre de cycles d'orientations paires de $F_1 \cup F_2$*

La preuve consiste à regarder ce qui se passe sur les signes des permutations associées aux deux couplages

Ce lemme donne une première idée pour calculer le Pfaffien avec son signe: calculer le signe de chaque terme relativement à une orientation canonique, c'est à dire en réorientant chaque arête du sommet le plus petit vers le sommet le plus grand.

La deuxième idée est de calculé chaque terme Pfaffien relativement à un couplage parfait de référence (qui peut être de poids nul): le couplage parfait \mathcal{I} associé à la permutation identité.

On déduit du lemme le corollaire suivant:

Corollaire 1 *$sgn(\sigma) = (-1)^k$ où k est le nombre de cycles d'orientation paire dans $\mathcal{M} \cup \mathcal{I}$ selon l'orientation canonique du graphe.*

Lemme 2 *Soit \mathcal{M} est un couplage parfait possible (une partition en paire de $\{1 \dots n\}$) et σ_m sa permutation canonique.*

Soit l le nombre de cycles d'orientation paire de $\mathcal{M} \cup \mathcal{I}$ et \mathcal{C} la couverture en cycle obtenue en prenant chaque cycle de $\mathcal{M} \cup \mathcal{I}$ dans le sens horaire.

Alors le signe du terme Pfaffien de \mathcal{M} est:

$$sgn(\mathcal{M}) = (-1)^{|\{ \langle i, j \rangle : \langle i, j \rangle \in \mathcal{C}, i < j \}| + l}$$

Corollaire 2 *$sgn(\sigma_{\mathcal{M}}) = (-1)^{|M_2| + |M_3| + l}$ où les ensembles M_2 et M_3 sont définis par:*

$$M_2 = \{ \langle i, 2j \rangle; \langle i, 2j \rangle \in \mathcal{M} \cap \mathcal{C}, i < 2j \}$$

$$M_3 = \{ \langle i, 2j - 1 \rangle; \langle i, 2j - 1 \rangle \in \mathcal{M} \cap \mathcal{C}, i > 2j \}$$

La preuve se base, encore une fois, sur l'étude du rapport entre le changement ou non de signe des permutations canoniques et la parité arêtes orientées correctement dans la superposition. Le choix d'une orientation canonique nous permet de calculer quand une arête fait changer le signe des permutations associées.

2.4 Les p-arêtes et les pclows

Une paire d'arêtes $E = (e_1, e_2)$ est une p-arête si pour $i \in [1, n]$, soit:

$e_1 = \langle i, 2j \rangle$ pour un certain j et $e_2 = \langle 2j, 2j - 1 \rangle$ ou

$e_1 = \langle i, 2j - 1 \rangle$ pour un certain j et $e_2 = \langle 2j - 1, 2j \rangle$

Une pclow est une clow avec sa suite ordonnée d'arêtes pouvant s'écrire $P = \langle E_1, E_2, \dots, E_n \rangle$ où les E_i sont des p-arêtes. Sa longueur $|P|$ est $2m$.

Une suite de pclow est à la pclow ce que la suite de clow est à la clow.

Le signe d'une suite de pclow est la parité du nombre de clow d'orientation paire par rapport à G redirigé.

Le poids d'une p-arête $E = \langle e_1, e_2 \rangle$ est le poids de e_1 prise dans le sens "petit sommet \rightarrow grand sommet". (e_2 a une contribution 1). Le poids d'une pclow est le produit des poids des p-arêtes qui la compose. Le poids d'une suite de pclows est le produit des poids des pclows.

Si une suite de pclow représente un couplage parfait \mathcal{M} (c'est à dire que les cycles de la suite sont ceux de la superposition de \mathcal{M} et de \mathcal{S}), alors son poids est $wt(\mathcal{M})$ et son signe est $sgn(\mathcal{M})$.

On peut généraliser les lemmes sur les superpositions pour avoir:

Lemme 3 Si \mathcal{P} est une suite de pclow contenant l pclows, alors:

$$sgn(\mathcal{P}) = (-1)^{|\{\langle i,j \rangle : \langle i,j \rangle \in \mathcal{P}, i < j\}|}$$

2.4.1 Théorème Principal

Les résultats précédents amènent à la caractérisation du Pfaffien suivante:

Théorème 1 Si D est une matrice anti-symétrique, son Pfaffien est donné par:

$$Pf(D) = \sum_{\mathcal{W}} sgn(\mathcal{W}).wt(\mathcal{W}) \text{ où } \mathcal{W} \text{ parcourt les suites de pclows}$$

La preuve consiste en la construction d'une involution sur les suites de pclow. Celles qui ne correspondent pas à des superpositions de couplages parfaits avec le couplage de référence sont envoyées sur des suites de pclow qui ne correspondent pas non plus à une telle superposition et qui sont de signe opposé. Les points fixes de l'involution sont les suites de pclow qui correspondent à des superpositions de couplages parfaits avec le couplage de référence, c'est à dire des couvertures en cycles.

La première partie consiste à "éliminer" les suites de pclow dans lesquelles il y a une pclow qui a un sous-cycle de longueur impaire. Ceci est réalisé en les envoyant sur des pclow inverses, i.e. la pclow obtenue en inversant les orientations du cycle de longueur impaire. Cette manipulation inverse juste le signe de la suite de pclow. La contribution de telles suites de pclow sera donc nulle dans la somme.

La deuxième partie consiste à éliminer les pclow qui ne sont pas des couvertures par cycles. On considère pour une telle suite de pclow P_1, P_2, \dots, P_k le plus petit i tel que $P_i + 1, \dots, P_k$ soient des cycles disjoints. 2 cas apparaissent lorsqu'on parcourt P_i :

- Soit on arrive à un sommet v tel que v touche P_j pour $j > i$. On appelle w le partenaire de v dans \mathcal{S} . Dans P_i , w est le successeur de v (on est arrivé à v en premier). Deux sous-cas se présente:
 - Dans P_j , on va de v vers w . Dans ce cas, on envoie cette suite de pclow sur la suite où P_i et P_j sont collées.
 - Dans P_j , on va de w vers v . Dans ce cas, on envoie cette suite de pclow sur la suite où P_i et l'orientation inverse de P_j sont collées.
- Soit on arrive à v qui complete un cycle simple de P dans P_i . w est défini comme précédemment. Dans ce cas on envoie cette suite de pclow sur celle où P et $P_i - P$ sont 2 pclows différentes.

Il ne nous reste que des suites de pclow correspondant à des superpositions de \mathcal{S} et de couplages parfaits, ce qui démontre le théorème.

3 Algorithmme

Pour déduire un algorithme des résultats précédents, on va construire un graphe H_D contenant 3 sommets spécifiques s, t_+, t_- et montrer que calculer le Pfaffien de D revient à calculer la fonction Gap-path de ce graphe (la différence des poids des chemins de s à t_+ et des poids des chemins de s à t_-).

H_D a pour ensemble de sommets $\{s, t_+, t_-\} \cup \{[p, h, u, i], p \in \{0, 1\}, h, u \in [1, n], i \in [0, n-1]\}$. Un chemin de s à (p, h, u, i) indique qu'une suite de pclow est construite le long de ce chemin. p est la parité de cette suite, h la tete de la pclow, u le sommet en cours et i le nombre d'arêtes traitées. Un chemin de s à t_+ (resp. à t_-) représente une suite de pclow de signe positif (resp. négatif).

H_D a n couches du type $[*, *, *, i]$ et les arêtes entre les couches $2j - 1$ et $2j$ sont construites à l'avance et ne dépendent pas de D .

Les arêtes de H_D sont:

1. $\langle s, [0, h, h, 0] \rangle$ pour $h = 2i - 1$ où $i \in [1, n/2]$ de poids 1.
2. $\langle [p, h, u, 2i], [1 - p, h, v, 2i + 1] \rangle$ avec $v > h, v > u, i \in [0, n/2 - 1]$ de poids d_{uv} .
3. $\langle [p, h, u, 2i], [p, h, v, 2i + 1] \rangle$ avec $v > h, v < u, i \in [0, n/2 - 1]$ de poids d_{vu} .
4. $\langle [p, h, 2j - 1, 2i - 1], [1 - p, h, 2j, 2i] \rangle$ si $2j - 1 > h, 2i < n$ de poids 1.
5. $\langle [p, h, 2j, 2i - 1], [p, h, 2j - 1, 2i] \rangle$ si $2j - 1 > h, 2i < n$ de poids 1.
6. $\langle [p, h, h + 1, 2i - 1], [1 - p, h', h', 2i] \rangle$ si $h' > h, h'$ est impair, $2i < n$ de poids 1.
7. $\langle [0, h, h + 1, n - 1], t_- \rangle$ et $\langle [1, h, h + 1, n - 1], t_+ \rangle$ si $h = 2i - 1$ de poids 1.

Les arêtes de poids 1 sont des arêtes de fonctionnement, celles de poids d_{uv} sont les arêtes qui représente la matrice et sur lesquelles est fait le calcul.

1. On ajoute les arêtes partant de la source jusqu'au départ de chaque pclow.
2. On ajoute une modélisation des arêtes uv de G , $v > u$ avec leur poids correspondant, qui changent le signe de la suite
3. On ajoute une modélisation des arêtes uv de G , $u > v$ avec leur poids correspondant, qui ne changent pas le signe de la suite.
4. On ajoute les parties triviales de la superposition: les arêtes $\langle 2j - 1, 2j \rangle$.
5. Idem
6. On ajoute les arêtes qui permettent de changer de pclow.
7. On ajoute les arêtes finales qui permettent de rejoindre t_- ou t_+ selon la parité de l'orientation, que l'on a "gardée en mémoire".

Il y a bien bijection entre les chemins de s à t_+ (resp. t_-) et les suites de pclow de G d'orientation paire (resp. impaire).

La preuve qu'une suite de pclow a une unique image dans les chemins se fait par induction sur la suite de pclow. On montre que si apres avoir parcouru la pclow P_i on est en $\langle p, h_{i+1}, h_{i+1}, m_i \rangle$ on sera en $\langle p, h_{i+2}, h_{i+2}, m_{i+1} \rangle$ apres la traversée de P_{i+1} (avec m_i le nombre d'arêtes de P_1, \dots, P_i).

Le théorème de correction est:

Théorème 2 Pour une matrice D antisymétrique donnée, le graphe construit ci dessus donne:

$$Pf(D) = \sum_{\rho: s \rightsquigarrow t_+} wt(\rho) - \sum_{\eta: s \rightsquigarrow t_-} wt(\eta)$$

En utilisant la programmation dynamique, $Pf(D)$ peut être évalué en temps polynomial: $O(n^4)$ opérations sur l'anneau auquel appartiennent les éléments de D .

Un algorithme parallèle pour calculer une fonction Gap-path peut être utilisé pour paralléliser le calcul du Pfaffien. Si D a des coefficients entiers, alors l'algorithme parallèle est dans GapL et on a:

Théorème 3 *Calculer le Pfaffien d'une matrice antisymétrique d'entiers est GapL-complet sous projections uniformes, même si on restreint les coefficients de la matrice à $\{-1, 0, 1\}$.*

La complétude vient du fait que pour une fonction f de Gap-Path, on peut considérer son graphe de calcul comme celui d'un Pfaffien d'une matrice où les coefficients sont déduits des poids des arêtes.

et finalement:

Théorème 4 *Le Pfaffien d'une matrice $n \times n$ antisymétrique sur un anneau commutatif peut être calculé par un circuit arithmétique de $O(n^4)$ portes et de taille $O(\log(n))$*

Il faut bien noter que ce calcul n'utilise pas de division, ce qui est numériquement intéressant.

4 Orientation Pfaffienne et Couplages Parfaits

4.1 Objectifs

Une *orientation pfaffienne* d'un graphe dirigé G est une orientation telle que tous les couplages parfaits ont le même signe.

Si on dispose d'une orientation Pfaffienne de G , alors en prenant D la matrice de Tutte de G telle que chaque arête a un poids 1, alors chaque couplage parfait va avoir un terme Pfaffien de +1 et la valeur du Pfaffien de D sera le nombre de couplages parfaits de G .

On va montrer comment obtenir une orientation Pfaffienne d'un graphe planaire.

Une orientation de G est dite *admissible* si tous les cycles simples résultants de superpositions de 2 couplages parfaits sont d'orientation paire.

Lemme 4 *admissible \equiv Pfaffienne*

4.2 Algorithme

On a d'abord:

Lemme 5 *Soit G un graphe planaire donné avec un plongement de G dans le plan et un indexage des faces tel que chaque face a une arête non-commune avec les faces d'indice inférieur.*

Alors trouver une orientation admissible pour G se réduit en logspace au problème de l'arbre de parité.

[2] donne une méthode pour trouver une orientation admissible:

1. Prendre une face, et orienter toutes les arêtes arbitrairement sauf une.
2. Pour la dernière arête prendre une orientation telle que le cycle qui entoure la face soit de parité impaire traversé en sens horaire.
3. S'il reste une face non traitée, revenir à l'étape 1

En traitant les faces selon leur indice croissant, on s'assure d'avoir toujours une arête à orienter.

On appelle *arête critique* d'une face C_i l'arête dont on va devoir choisir l'orientation.

Il y a 3 types d'arêtes qui déterminent l'orientation de l'arête critique de C_i :

- les arêtes non critiques des faces d'indice plus petit.

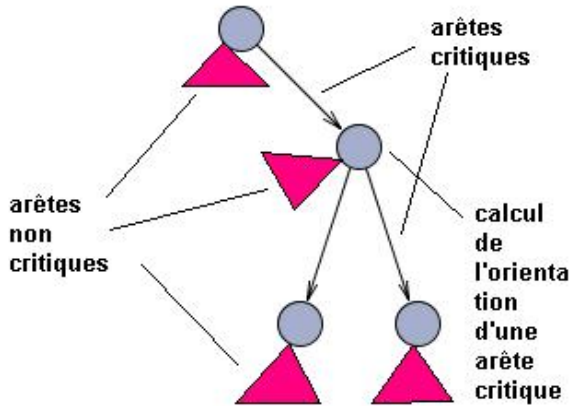


Figure 2: Structure de calcul des orientations.

- les arêtes critiques des faces d'indice plus petit.
- les arêtes de C_i que l'on vient d'orienter arbitrairement.

Ainsi lorsqu'on parcourt une face pour calculer l'orientation d'une arête critique, on rencontre des arêtes critiques de cycles précédents, dont l'orientation est le résultat d'un calcul similaire. On peut ainsi représenter le calcul comme un arbre de parité, où chaque noeud est un calcul et chaque arête critique un lien vers un noeud fils. Comme le graphe est planaire, une arête n'est commune qu'à 2 faces, et cette structure est bien un arbre (cf figure 2).

Lemme 6 *Calculer un arbre de parité peut se faire de manière logspace.*

Ce qui nous conduit au théorème:

Théorème 5 *Soit G un graphe planaire donné avec un plongement de G dans le plan et un indilage des faces tel que chaque face a une arête non-commune avec les faces d'indice inférieur.*

Alors compter le nombre de couplages parfaits de G est dans GapL.

Notons qu'on ne sait pas s'il est possible d'obtenir un tel indilage dans GapL, mais c'est faisable dans NL.

5 Contexte, Critiques, Conclusions

5.1 Résumé

L'article [4] présente un algorithme dans $NC \cap \text{GapL}$ pour calculer le Pfaffien d'une matrice anti-symétrique. On sait déjà calculer le Pfaffien au signe pres en utilisant le déterminant. Les auteurs donnent ici un algorithme qui calcule le Pfaffien directement.

Dans une première partie la définition du Pfaffien est rappelée, son lien avec les couplages parfaits d'un graphe et les orientations des cycles.

Ensuite des outils comme les pflows et les superpositions de couplages parfaits sont introduits. Des théorèmes sur les signes des termes Pfaffiens relativement à un couplage de référence sont ensuite démontrés jusqu'au théorème qui caractérise le Pfaffien comme somme sur toutes les suites de pflows.

Un algorithme est déduit de cette caractérisation, permettant de prouver les résultats de complexité annoncés sur le Pfaffien.

Une dernière partie explique comment calculer le nombre de couplages parfaits d'un graphe planaire, en trouvant une orientation Pfaffienne.

5.2 Contexte

Les références listées dans cet article sont nombreuses, car, comme on l'a dit précédemment, cet article réutilise beaucoup de résultats. On trouve des références sur les classes de complexité, d'autres sur l'algorithmique de graphe comme [2]. Une référence importante est [1] car c'est cet article qui donne un calcul du Pfaffien au signe près en utilisant le déterminant.

Une recherche sur *Google Scholar* montre que les articles traitant du Pfaffiens sont nombreux et certains datent des années 60.

L'article [4] a été cité 6 fois selon *Google Scholar*. C'est un article de 2004 et beaucoup d'autres articles traitant de résultats sur le Pfaffien existent déjà. Une citation notable vient de [3], un article qui présente un algorithme pour trouver un couplage parfait d'un graphe biparti de genre borné.

M. Mahajan a publié beaucoup d'articles sur les couplages et la classe de complexité NC . Un de ses articles récents est intitulé *On the bipartite unique perfect matching problem* qui n'utilise pas le Pfaffien.

5.3 Critiques

Cet article, quoique difficile d'accès, est assez compréhensible une fois que toutes les notions et les résultats préliminaires sont assimilés. Il est suffisamment didactique dans sa structure pour assurer une compréhension séquentielle des résultats. Toutefois les termes similaires traitant de l'orientation ou du signe d'un cycle, d'un sous-cycle, d'une pflow, d'une suite de pflow, d'un terme Pfaffien ou d'une permutation induisent parfois en erreur.

Les preuves principales, comme celle de la caractérisation du Pfaffien comme somme sur l'ensemble de pflows, sont détaillées et convaincantes. On regrette que d'autres, comme celle de la GapL-complétude soient moins prolixes.

Les résultats démontrés sont motivés par l'absence d'algorithme qui calcule directement le Pfaffien avec son signe, ce qui fait que l'article apparaît comme tout à fait pertinent.

References

- [1] F Maffioli G Galbiati. On the computation of pfaffians. *Discrete Applied Mathematics*, 1994.
- [2] P.W. Kastelyn. Graph theory and crystal physics. *Graph Theory and Theoretical Physics*, 1967.
- [3] K. Varadhan Meena Mahajan. A new nc-algorithm for finding a perfect matching in planar and bounded genus graph. *Proceedings of the 32nd Annual ACM Symposium on Theory of Computing*, 2000.
- [4] V. Vinay Meena Mahajan, P.R. Subramanya. The combinatorial approach yields an nc algorithm for computing pfaffians. *Discrete Applied Mathematics*, 2004.
- [5] Thomas Cormen Charles Leiserson Ronald Rivest Clifford Stein. Introduction to algorithms. *Massachusetts Institute of Technology*, 2001.