



HAL
open science

Longueur linéaire des graphes planaires extérieurs

Thomas Dissaux, Nicolas Nisse

► **To cite this version:**

Thomas Dissaux, Nicolas Nisse. Longueur linéaire des graphes planaires extérieurs. AlgoTel 2022 - 24èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications, May 2022, Saint-Rémy-Lès-Chevreuse, France. hal-03655647

HAL Id: hal-03655647

<https://hal.science/hal-03655647>

Submitted on 29 Apr 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Longueur linéaire des graphes planaires extérieurs [†]

Thomas Dissaux¹ and Nicolas Nisse¹

¹ Université Côte d’Azur, Inria, CNRS, I3S, France

Une *décomposition linéaire* (*path-decomposition*) d’un graphe $G = (V, E)$ est une représentation de G comme une séquence de sous-ensembles (appelés *sacs*) de V vérifiant des propriétés de connexité. La *longueur* d’une décomposition linéaire est le plus grand diamètre d’un de ses sacs et la *longueur linéaire* de G est la plus petite longueur d’une de ses décompositions linéaires. Ce paramètre a été étudié pour ses applications algorithmiques dans des problèmes métriques classiques (e.g., plus court chemin d’excentricité minimum, plongement de graphes dans des espaces métriques avec faible distorsion des distances...). Il est connu que décider si la longueur linéaire d’un graphe quelconque est au plus 2 est NP-complet, et le meilleur algorithme d’approximation connu a un rapport d’approximation de 2 (il n’existe pas de c -approximation pour $c < \frac{3}{2}$). Dans cet article, nous étudions ce problème dans des sous-classes des graphes planaires. Nous prouvons que la longueur linéaire d’un cycle de n sommets est $\lfloor \frac{n}{2} \rfloor$ et que celle des arbres peut être déterminée en temps linéaire. Notre résultat principal est un algorithme polynomial donnant une $+1$ -approximation de la longueur linéaire des graphes planaires extérieurs 2-connexes. Cet algorithme repose sur le fait que, dans cette classe de graphes, nous prouvons qu’il existe des décompositions (presque) optimales (de longueur minimum à un près) avec des propriétés structurelles spécifiques.

Mots-clefs : décomposition linéaire, longueur linéaire, graphes planaires extérieurs, dual.

1 Introduction

Les décompositions linéaires (*path-decompositions*) de graphes sont très étudiées depuis leur introduction dans le cadre de la théorie des mineurs de graphes de Robertson et Seymour et pour leurs nombreuses applications algorithmiques. Une *décomposition linéaire* d’un graphe $G = (V, E)$ est une séquence (X_1, \dots, X_p) de sous ensembles (appelés *sacs*) de V telle que (1) $\bigcup_{i \leq p} X_i = V$, (2) pour toute arête $\{u, v\} \in E$, il existe $1 \leq i \leq p$ tel que $u, v \in X_i$, et (3) pour tout $1 \leq i \leq z \leq j \leq p$, $X_i \cap X_j \subseteq X_z$. Ces contraintes impliquent la propriété fondamentale (notamment dans les preuves) selon laquelle, pour tout $1 \leq i < p$, $S = X_i \cap X_{i+1}$ sépare $A = \bigcup_{j \leq i} X_j \setminus S$ et $B = \bigcup_{j \geq i} X_j \setminus S$ (i.e., tout chemin de A à B passe par S).

La mesure la plus connue d’une décomposition linéaire est sa *largeur* (correspondant à la taille maximum de ses sacs). La *largeur linéaire* (*pathwidth*) d’un graphe G est la largeur minimum d’une de ses décompositions. Typiquement, le fameux théorème de Courcelle implique que de nombreux problèmes NP-complets en général peuvent être résolus en temps polynomial dans les graphes de largeur linéaire bornée.

Nous nous intéressons à une mesure des décompositions qui, bien que moins étudiée, a aussi de nombreuses applications algorithmiques. Pour tous $u, v \in V$, soit $d_G(u, v)$ (ou $d(u, v)$ s’il n’y a pas d’ambiguïté) la distance entre u et v dans G . La *longueur* $\ell(D)$ d’une décomposition $D = (X_1, \dots, X_p)$ de G est le *diamètre* maximum de ses sacs, i.e., $\ell(D) = \max_{i \leq p} \max_{u, v \in X_i} d_G(u, v)$. La *longueur linéaire* $pl(G)$ (*path-length*) de G est la longueur minimum de ses décompositions linéaires [DG07]. En particulier, cette mesure permet de capturer des propriétés métriques des graphes. Par exemple, le problème de plongement avec faible distorsion de graphes dans des espaces métriques peut être approché lorsque la longueur linéaire est bornée [DKL17], ce qui a des applications en vision par ordinateur, chimie et biologie numérique, protocoles distribués...(voir les références dans [DLN20]). De plus, la longueur linéaire étant une borne

[†]Ce travail est partiellement financé par les projets UCA JEDI (ANR-15-IDEX-01) et EUR DS4H (ANR-17-EURE-004) Investments in the Future. Par manque de place, les preuves sont uniquement esquissées. Les preuves complètes sont disponibles ici.

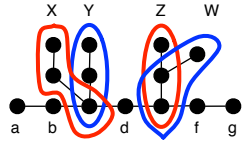


FIGURE 1: Exemple d'un arbre T (avec $pl(T) = 2$) dont une décomposition optimale décrite dans la preuve du Théorème 1 est $(\{ab\}, \{bc\}, X, Y, \{cd\}, \{de\}, Z, W, \{ef\}, \{fg\})$.

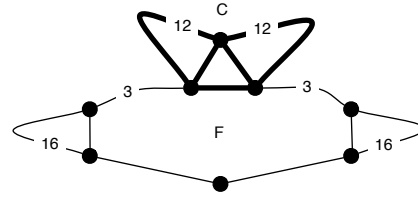


FIGURE 2: Exemple d'un graphe G dont aucune décomposition optimale ($pl(G) = 10$) ne vérifie la Propriété 2 pour la composante C (en gras). Une ligne avec une étiquette x représente un chemin de x arêtes.

supérieure de la longueur arborescente, des problèmes comme celui du voyageur de commerce ou de la dimension métrique peuvent être résolus (ou approchés) efficacement dans les graphes de longueur linéaire bornée [KL06, BFGR17, DDNN21].

Étant donné un graphe quelconque G , décider si $pl(G) \leq 2$ est NP-complet et il n'existe pas de c -approximation pour le calcul de $pl(G)$ pour tout $c < 3/2$ (si $P \neq NP$) [DLN20]. Il existe une 2-approximation pour ce problème [DKL17] mais la complexité dans le cas des graphes planaires n'est pas connue. Dans cet article, nous nous attaquons à ce problème en commençant par les *graphes planaires extérieurs* (qui peuvent être plongés dans le plan, sans croisement d'arêtes et tel que tout sommet est sur la face extérieure). Notons que cette classe de graphes a aussi été étudiée dans le cas de la longueur arborescente [DG07] et de la largeur linéaire. Pour le calcul de la largeur linéaire des graphes planaires extérieurs, le meilleur algorithme exact connu a une complexité au moins $O(|V|^{11})$ et il existe une 2-approximation [BF02, CHS07].

Contributions. Dans cet article, nous donnons d'abord un algorithme linéaire permettant de calculer la longueur linéaire des arbres et prouvons que $pl(C_n) = \lfloor \frac{n}{2} \rfloor$ pour tout cycle C_n de n sommets. Notre résultat principal est un algorithme qui calcule, en temps $O(n^3 \cdot pl(G)^2)$, une décomposition linéaire de longueur au plus $pl(G) + 1$ de tout graphe planaire extérieur 2-connexe G de n sommets.

Préliminaires. Un *chemin diamétral* de $G = (V, E)$ est un plus court chemin de longueur maximum ($\max_{x, y \in V} d(x, y)$) dans G . La distance $d(u, S)$ de $u \in V$ à $S \subseteq V$ vaut $\min_{v \in S} d(u, v)$. Dans la suite, "décomposition" signifiera toujours décomposition linéaire. La propriété suivante est très utile dans nos preuves. Un sous-graphe H d'un graphe G est *isométrique* si, pour tout $u, v \in V(H)$, $d_H(u, v) = d_G(u, v)$ (les distances sont préservées dans H). Pour tout sous-graphe H isométrique de G , $pl(H) \leq pl(G)$ [DG07].

2 Arbres, Cycles et Graphes planaires extérieurs

Nous commençons par l'étude de la longueur linéaire des arbres. Ce résultat relativement simple nous permettra de donner l'intuition de notre contribution principale sur les graphes planaires extérieurs.

Théorème 1 *Soit $T = (V, E)$ un arbre d'au moins 2 sommets. Si T est un chemin, alors $pl(T) = 1$, sinon, soit P un chemin diamétral de T et $z \in V$ tel que $d(z, V(P)) = \max_{v \in V} d(v, V(P)) = k > 0$. Alors, $pl(T) = k$.*

Esquisse de preuve : Supposons que T n'est pas un chemin. Alors T contient une *k-araignée* (graphe obtenu d'une étoile à trois feuilles dont chaque arête a été subdivisée $k - 1$ fois) comme sous graphe isométrique et donc $pl(T) \geq k$ [DG07]. Pour la borne supérieure, nous définissons une décomposition D de T de longueur k . Intuitivement, elle "suit" P en "ajoutant" les "branches" séquentiellement dans l'ordre où elles sont rencontrées. Formellement, soient $P = (v_0, \dots, v_r)$, $e_i = \{v_{i-1}, v_i\}$ et T_i la composante de $T \setminus \{e_i, e_{i+1}\}$ contenant v_i ($0 \leq i < r$). Le premier sac de D est $\{v_0, v_1\}$ puis, séquentiellement (de $i = 1$ à $r - 1$), on "ajoute" les sacs contenant chacun le chemin de v_i à une feuille de T_i (pour toutes les feuilles de T_i , ordonnées dans un ordre DFS arbitraire de T_i) puis le sac $\{v_i, v_{i+1}\}$ (voir une illustration en Figure 1). Clairement, P et une telle décomposition (et $pl(T)$) peuvent être calculés en temps linéaire. \diamond

Nous nous tournons maintenant vers les graphes planaires extérieurs 2-connexes. Pour les plus simples d'entre eux, les cycles, la longueur linéaire peut être calculée en temps constant.

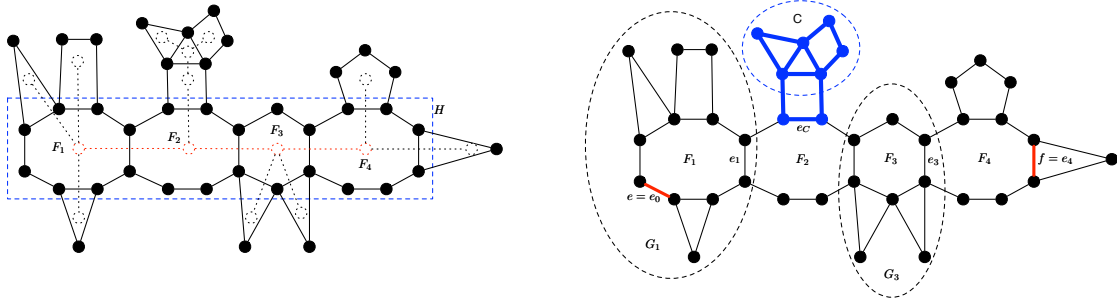


FIGURE 3: Notations du Théorème 3 sur un exemple. À gauche, un graphe G planaire extérieur 2-connexe (en traits pleins), son dual faible G^* (pointillés) avec le chemin $P^* = (F_1, \dots, F_4)$ (rouge) associé aux arêtes e et f et le “chemin de faces” H correspondant (en bleu). À droite, sont représentés les arêtes $e = e_0, \dots, e_4 = f$, les faces F_1, \dots, F_4 et les sous-graphes G_1 et G_3 . Le sous-graphe \bar{C} associé à la composante C de $G \setminus H$ est représenté en bleu et en gras.

Théorème 2 Soit $C_n = (V, E)$ un cycle de $n \geq 3$ sommets. Alors, $pl(C_n) = \lfloor \frac{n}{2} \rfloor$.

Esquisse de preuve : La décomposition avec l’unique sac $V(C_n)$ a longueur $\ell = \lfloor \frac{n}{2} \rfloor \geq pl(C_n)$. Pour prouver la borne inférieure, soient $u, v \in V$ diamétralement opposés et P et P' les deux u - v -chemins dans C_n . Soit une décomposition $D = (X_1, \dots, X_p)$ de C_n . S’il existe un sac de D contenant u et v , alors $\ell(D) \geq \ell$. Sinon, soient $1 \leq i < j \leq p$ avec $u \in X_i$ et $v \in X_j$ et $j - i$ minimum. Pour tout $i < k \leq j$, $X_k \cap X_{k-1}$ sépare u et v et doit donc contenir un sommet de P et un sommet de P' . Nous montrons qu’il existe $i < k < j$ tel que X_k contient deux sommets diamétralement opposés et donc la longueur de D est au moins ℓ . \diamond

Considérons un graphe $G = (V, E)$ planaire extérieur 2-connexe. Le dual faible G^* de G est le graphe dont les sommets sont les faces internes (bornées) de G (notons que, par 2-connexité, G a un unique plongement planaire extérieur) et deux sommets de G^* sont adjacents si les faces correspondantes partagent une arête dans G . Il est bien connu que G^* est un arbre (voir Fig. 3). En suivant l’idée de preuve du Théorème 1, étant donné un chemin P^* de G^* , une décomposition (pas forcément optimale) de l’arbre G^* peut être définie en “suivant” P^* et en intégrant les “branches” (de $G^* \setminus P^*$) dans l’ordre où elles sont rencontrées.

Nous “transposons” ce type de décomposition au graphe planaire extérieur 2-connexe G . Intuitivement, P^* correspond à un “chemin de faces” H (le sous-graphe de G induit par les sommets des faces de P^*) dans G et les “branches” de $G^* \setminus P^*$ correspondent dans G aux composantes connexes de $G \setminus H$. Nous montrons qu’il existe une décomposition linéaire de G “presque” optimale (de longueur au plus $pl(G) + 1$) qui “suit” H (pour un certain chemin de G^*) de la même façon. Nous montrons également qu’une telle décomposition peut être calculée en temps polynomial.

Théorème 3 Soit $G = (V, E)$ un graphe planaire extérieur 2-connexe de n sommets. Une décomposition linéaire de G de longueur au plus $pl(G) + 1$ peut être calculée en temps $O(n^3 pl(G)^2)$.

Esquisse de preuve : Notre algorithme calcule en fait $O(n^2)$ décompositions, une par paire d’arêtes de G (puisque G est planaire, $|E| = O(n)$), puis renvoie l’une de ces décompositions de longueur minimum. Précisément, étant fixées deux arêtes $e, f \in E$ (possiblement $e = f$), nous calculons une certaine (voir ci-dessous) décomposition de G dont le premier sac contient e et le dernier sac contient f .

Si $e = f$, nous concevons un algorithme glouton qui calcule, en temps $O(n)$, une meilleure (de plus petite longueur) décomposition de G parmi celles dont le premier et le dernier sacs (donc tous les sacs d’après la troisième propriété des décompositions linéaires) contiennent e .

Le cas $e \neq f$ est (beaucoup) plus technique et nécessite des notations (illustrées sur la Figure 3). Soit $P^* = (F_1, \dots, F_r)$ le plus court chemin de G^* (le dual faible de G) avec $e \in V(F_1)$ et $f \in V(F_r)$ (possiblement $F_1 = F_r$). Rappelons que P^* correspond à un sous-graphe (un “chemin de faces”) H de G . Pour $1 \leq i < r$, soit e_i l’arête partagée par les faces F_i et F_{i+1} ($e = e_0$ et $f = e_r$). Pour toute composante C de $G \setminus H$, soit e_C l’arête de H telle que $N(C) = \{u \in V \setminus C \mid \exists v \in C, \{u, v\} \in E\} = e_C$ et soit \bar{C} le sous-graphe de G induit par C et e_C . De plus, soit D_C une décomposition optimale de \bar{C} parmi celles où e_C appartient à chacun des

sacs. Finalement, pour tout $1 \leq i \leq r$, soit G_i le sous-graphe de G induit par les sommets de la face F_i et ceux de chaque composante C de $G \setminus H$ telle que $e_C \in E(F_i)$ (l'ensemble de ces composantes est noté C_i). Nous donnons un algorithme qui calcule une décomposition D^{ef} de longueur minimum parmi celles qui vérifient les 3 propriétés suivantes :

1. **“face de H par face de H , de gauche à droite (de 1 à r)”** : $D^{ef} = D_1 \odot \dots \odot D_r$ (\odot est pour la concaténation de séquences) avec, pour $1 \leq i \leq r$, D_i est une décomposition de G_i (on décompose séquentiellement G_1 puis G_2 , ... puis G_r) avec e_{i-1} dans son premier sac et e_i dans son dernier sac ;
2. **pour chaque face de H , “composante par composante” de $G \setminus H$** : pour tout $1 \leq i \leq r$, les composantes de C_i sont “ajoutées” une à une dans $D_i = (X_1^i, \dots, X_{r_i}^i)$. Formellement, pour tout $C \in C_i$, il existe $1 \leq a_C \leq b_C \leq r_i$ tels que (1) la restriction de $(X_{a_C}^i, \dots, X_{b_C}^i)$ à \bar{C} est exactement D_C et (2) pour tout $a_C \leq j \leq j' \leq b_C$, $X_j^i \setminus V(\bar{C}) = X_{j'}^i \setminus V(\bar{C}) \subseteq V(F_i)$. Informellement, une fois qu'un sommet d'une composante $C \in C_i$ est “ajouté” à D_i , alors on ne peut pas “ajouter” un sommet d'une autre composante tant que tous les sommets de C n'ont pas été “ajoutés” à D_i ;
3. **pour chaque face de H , de “gauche à droite”** : pour tout $1 \leq i \leq r$, l'ordre dans lequel les composantes de C_i sont “ajoutées” à D_i est contraint de la manière suivante (intuitivement, “de gauche à droite”). Précisément, soient A_i et B_i les deux chemins disjoints de e_i à e_{i+1} dans F_i . Pour toutes $C, C' \in C_i$ telles que $e_C, e_{C'} \in E(A_i)$ (resp., $e_C, e_{C'} \in E(B_i)$) avec e_C plus proche de e_{i-1} que $e_{C'}$ dans A_i (resp., dans B_i), alors C doit “arriver avant” C' dans D_i (i.e., $b_C < a_{C'}$).

Les deux arêtes e et f ($e \neq f$) fixées, notre algorithme calcule d'abord les décompositions D_C pour chaque composante C de $G \setminus H$, ce qui peut être calculé en temps global $O(n)$ par l'algorithme glouton mentionné précédemment. Le point délicat est, pour chaque $1 \leq i \leq r$, de calculer un ordre (Propriété 3) optimal (permettant de minimiser la longueur). Nous définissons un algorithme par programmation dynamique qui calcule un tel ordre, et donc D_i , en temps $O(|F_i|^2) = O(pl(G)^2)$ (toute face F est un cycle isométrique et donc $|F| = O(pl(G))$). Donc, notre algorithme calcule D^{ef} en temps $O(n + r \cdot pl(G)^2) = O(n \cdot pl(G)^2)$.

Ainsi, les $O(n^2)$ décompositions désirées sont calculées en temps total $O(n^3 pl(G)^2)$. La clé de la preuve est de montrer que l'une d'entre elle a longueur au plus $pl(G) + 1$. Pour cela, nous considérons une décomposition optimale D ($\ell(D) = pl(G)$) de G et montrons que, s'il n'existe pas une arête présente dans chacun de ses sacs, alors il est possible de la transformer en une décomposition de G satisfaisant les trois propriétés (pour deux arêtes différentes $e, f \in E$ déterminées en fonction de D) et de longueur au plus $pl(G) + 1$. \diamond

Perspectives. La prochaine étape sera naturellement de concevoir un algorithme polynomial exact (s'il en existe un) pour calculer la longueur linéaire des graphes planaires extérieurs. Notons que la dégradation de la longueur (+1) dans notre algorithme vient de la propriété 2 des décompositions que nous considérons. L'exemple de la Figure 2 montre que l'augmentation de la longueur ne peut pas être évitée si l'on conserve cette propriété. Notons également que la condition de 2-connexité que nous imposons ici ne semble pas être rédhibitoire (mais simplifie certains aspects de rédaction des preuves). Une autre question est de savoir si notre algorithme pour les arbres peut être adapté au cas des graphes cordaux. Enfin, déterminer la complexité du calcul de la longueur linéaire (ou arborescente) des graphes planaires reste ouvert.

Références

- [BF02] H. L. Bodlaender and F. V. Fomin. Approximation of pathwidth of outerplanar graphs. *J. Alg.*, 43(2) :190–200, 2002.
- [BFR17] R. Belmonte, F. V. Fomin, P. A. Golovach, and M. S. Ramanujan. Metric dimension of bounded tree-length graphs. *SIAM J. Discret. Math.*, 31(2) :1217–1243, 2017.
- [CHS07] D. Coudert, F. Huc, and J.-S. Sereni. Pathwidth of outerplanar graphs. *J. Graph Theory*, 55(1) :27–41, 2007.
- [DDNN21] T. Dissaux, G. Ducoffe, N. Nisse, and S. Nivelle. Treelength of series-parallel graphs. In *XI Latin and American Algorithms, Graphs and Optimization Symposium (LAGOS)*, volume 195, pages 30–38, 2021. (et dans **AlgoTel 2021**).
- [DG07] Y. Dourisboure and C. Gavaille. Tree-decompositions with bags of small diameter. *Discret. Math.*, 307(16) :2008–2029, 2007.
- [DKL17] F. F. Dragan, E. Köhler, and A. Leitert. Line-distortion, bandwidth and path-length of a graph. *Algorithmica*, 77(3) :686–713, 2017.
- [DLN20] G. Ducoffe, S. Legay, and N. Nisse. On the complexity of computing treewidth. *Algorithmica*, 82(6) :1574–1600, 2020.
- [KL06] R. Krauthgamer and J. R. Lee. Algorithms on negatively curved spaces. In *47th Annual IEEE Symp. on Foundations of Computer Science (FOCS 2006)*, pages 119–132, 2006.