



HAL
open science

Longueur Arborescente des Graphes Série-Parallèles

Thomas Dissaux, Guillaume Ducoffe, Nicolas Nisse, Simon Nivelles

► **To cite this version:**

Thomas Dissaux, Guillaume Ducoffe, Nicolas Nisse, Simon Nivelles. Longueur Arborescente des Graphes Série-Parallèles. ALGOTEL 2021 - 23èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications, Jun 2021, La Rochelle, France. hal-03217731v1

HAL Id: hal-03217731

<https://hal.science/hal-03217731v1>

Submitted on 5 May 2021 (v1), last revised 5 May 2021 (v2)

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 Arborescente des Graphes Série-Parallèles[†]

Thomas Dissaux¹, Guillaume Ducoffe², Nicolas Nisse¹ et Simon Nivelles³

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

² University of Bucharest & National Institute of Research and Development in Informatics, Romania

³ INSPÉ Paris, Sorbonne Université, France

La *longueur* d'une décomposition arborescente d'un graphe G est la plus grande distance entre deux sommets d'un sac de la décomposition. La *longueur arborescente* de G est la plus petite longueur d'une de ses décompositions arborescentes. Ce paramètre a été étudié pour ses applications algorithmiques dans des problèmes classiques comme le problème du voyageur de commerce ou le calcul de tables de routage compactes et également pour ses liens avec la largeur arborescente. Décider si la longueur arborescente d'un graphe quelconque est au plus 2 est NP-complet, et le meilleur algorithme d'approximation connu a un rapport d'approximation de 3 (il n'existe pas de c -approximation pour $c < \frac{3}{2}$). Le problème de calculer la longueur arborescente est facile dans certaines classes de graphes comme celle des graphes planaires extérieurs. Cependant, rien n'est connu sur la complexité de ce problème dans d'autres classes de graphes planaires. Dans cet article, nous étudions ce problème dans la classe des graphes série-parallèles (graphes SP). Nous montrons que le problème est non-trivial dans les graphes melons, une sous-classe très simple des graphes SP. Nous concevons une $\frac{3}{2}$ -approximation pour calculer la longueur arborescente (et une décomposition correspondante) dans les graphes SP. Notre résultat principal est que décider si la longueur arborescente d'un graphe SP est au plus 2 peut être résolu en temps polynomial. Ce résultat se base sur une caractérisation de ces graphes par des sous-graphes isométriques interdits.

Mots-clefs : décomposition arborescente, longueur arborescente, graphes série-parallèles, sous-graphes isométriques.

1 Introduction

Les *décompositions arborescentes* 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 arborescente d'un graphe $G = (V, E)$ est une paire $(T, \mathcal{X} = \{X_t \mid t \in V(T)\})$ telle que T est un arbre et \mathcal{X} est une famille de sous-ensembles (appelés sacs) de sommets dans G telle que :

- $\bigcup_{t \in V(T)} X_t = V(G)$;
- Pour toute arête $\{u, v\} \in E(G)$, il existe $t \in V(T)$ tel que $u, v \in X_t$;
- Pour tout sommet $v \in V(G)$, l'ensemble $\{t \in V(T) \mid v \in X_t\}$ induit un sous-arbre de T .

Une mesure classique d'une décomposition arborescente (T, \mathcal{X}) est sa *largeur* $w((T, \mathcal{X})) = \max_{t \in V(T)} |X_t| - 1$, correspondant à la taille d'un plus gros sac. La *largeur arborescente* $tw(G)$ de G est la plus petite largeur des décompositions arborescentes de G . Les décompositions arborescentes ont été extrêmement étudiées pour leurs applications algorithmiques. Par exemple, de nombreux problèmes NP-difficiles peuvent être résolus en temps linéaire dans la classe des graphes de largeur arborescente bornée [6]. Le calcul de la largeur arborescente d'un graphe est un problème NP-complet [1] mais FPT en la taille de la solution [3]. Il n'existe pas d'algorithme d'approximation de ratio constant pour ce problème en général. Par ailleurs, des algorithmes polynomiaux pour résoudre ce problème ont été proposés dans de nombreuses classes de graphes. En revanche, la complexité du problème dans les graphes planaires reste ouverte depuis plus de 30 ans. Le meilleur algorithme polynomial connu pour les graphes planaires est une $\frac{3}{2}$ -approximation [14].

[†]Les preuves omises dans cet article sont disponibles ici [8]

Longueur arborescente. D’autres mesures ont été proposées pour les décompositions arborescentes. En particulier, la *longueur* $length((T, \mathcal{X}))$ d’une décomposition (T, \mathcal{X}) d’un graphe $G = (V, E)$ est la plus grande distance (dans G) entre deux sommets d’un même sac. Précisément, étant donnés $u, v \in V$, $d_G(u, v)$ est le plus petit nombre d’arêtes d’un chemin entre u et v dans G , et $length((T, \mathcal{X})) = \max_{t \in V(T)} \max_{u, v \in \mathcal{X}_t} d_G(u, v)$. La *longueur arborescente* $tl(G)$ est la plus petite longueur d’une décomposition arborescente de G [9].

Comme la largeur arborescente, la longueur arborescente a de nombreuses applications algorithmiques. Par exemple, le problème du voyageur du commerce admet un FPTAS [12], les problèmes de couverture et de “packing” admettent des algorithmes d’approximation [4] et des schémas de routage compacts peuvent être calculés [11] dans les graphes de longueur arborescente bornée, le calcul de la dimension métrique est FPT quand la longueur arborescente est bornée [2], *etc.* En contraste avec la largeur arborescente, calculer la longueur arborescente est non seulement NP-complet mais n’est pas même FPT. Précisément, décider si la longueur arborescente d’un graphe est au plus 2 est NP-complet [13]. En revanche, il existe une $\frac{3}{2}$ -approximation pour ce problème [9] (mais pas de c -approximation générale pour $c < \frac{3}{2}$ si $P \neq NP$ [13]).

Les principales applications de la longueur arborescente (TSP, routage, *etc.*) suggèrent que le calcul de ce paramètre dans les graphes planaires est un problème important. Peu de résultats à ce sujet sont connus, le principal étant que la longueur arborescente du cycle C_n à n sommets est égale à $\lceil \frac{n}{3} \rceil$ [9]. Ajouté au fait que, pour tout sous-graphe *isométrique* H de G (un sous-graphe H de G est isométrique dans G si les distances dans H sont égales à celles dans G), $tl(H) \leq tl(G)$, cela permet de prouver que, pour tout graphe *planaire extérieur* G (qui admet un plongement planaire tel que tous les sommets sont sur la face extérieure), $tl(G) = \lceil \frac{is(G)}{3} \rceil$ où $is(G)$ est la taille d’un plus grand cycle isométrique de G [9]. Ainsi, dans cet article, nous initions l’étude de la longueur arborescente dans une autre sous-classe de graphes planaires très étudiée, à savoir les graphes série-parallèles [10].

Pour conclure cette introduction, une autre motivation pour étudier la longueur arborescente des graphes planaires vient de la relation entre longueur et largeur arborescentes dans une large classe de graphes (incluant les graphes planaires) [5]. Ainsi, obtenir de nouveaux résultats sur la longueur arborescente des graphes planaires pourrait permettre de mieux comprendre leur largeur arborescente et, donc, de progresser dans l’étude de la complexité de calculer la largeur arborescente des graphes planaires (voir, e.g., [7]).

Nos contributions. Nous étudions le calcul de la longueur arborescente dans la classe des graphes (planaires) série-parallèles (SP). Dans un premier temps, nous explicitons une formule close de la longueur arborescente des graphes *melons* (obtenus par composition parallèle de plusieurs chemins). Dans ces graphes, la longueur arborescente dépend à la fois de la longueur d’un plus grand cycle et de celle d’un plus grand cycle isométrique (qui peuvent être calculées en temps polynomial) et la formule non-triviale que nous obtenons laisse présager de la difficulté du problème. Nous présentons ensuite une $\frac{3}{2}$ -approximation pour le calcul de la longueur arborescente dans la classe des graphes SP. Notre résultat principal est un algorithme polynomial pour décider si $tl(G) \leq 2$ pour tout graphe SP G . Ce résultat repose sur une caractérisation des graphes SP de longueur arborescente au plus 2 par des sous-graphes isométriques interdits.

2 Graphes Série-Parallèles et Graphes Melons

Tout d’abord, restreignons les classes de graphes à considérer. Étant donné un graphe $G = (V, E)$, un ensemble $S \subset V$ est un *séparateur* si supprimer S de G augmente le nombre de composantes connexes. De plus, S est une *clique-séparatrice* si S induit un graphe complet dans G . Il est facile de prouver que le problème du calcul de la longueur arborescente peut se restreindre à la classe des graphes sans clique-séparatrices. Dans ce qui suit, nous supposons que les graphes considérés sont sans clique-séparatrices et, en particulier, sont 2-connexes, i.e., sans sommet (clique de taille 1) séparateurs.

La classe des graphes SP est définie récursivement comme suit [10]. Informellement, un graphe SP est soit une arête, soit est obtenu par la “composition en série” ou la “composition en parallèle” de deux graphes SP. Les graphes SP ont été très étudiés à la fois dans le contexte de la théorie des graphes et pour des applications pratiques, entre autres parce qu’ils modélisent des circuits électriques [10].

Certaines de nos preuves sont basées sur la définition suivante qui est équivalente dans le cas des graphes 2-connexes. Un graphe 2-connexe est un graphe SP si et seulement si il admet une *décomposition en oreilles imbriquées* [10]. Nous prouvons de plus qu’il est possible d’ajouter certaines conditions d’isométrie.

Précisément, un graphe SP 2-connexe $G = (V, E)$ admet une *décomposition isométrique en oreilles imbriquées*, c'est-à-dire une partition (E_0, \dots, E_p) de E telle que :

- E_0 induit un plus grand cycle isométrique dans G ;
- Pour tout $0 \leq i \leq p$, le sous-graphe G_i induit par $\bigcup_{j \leq i} V(E_j)$ est un graphe SP 2-connexe qui est un sous-graphe isométrique de G ;
- Pour tout $1 \leq i \leq p$, il existe $j_i < i$ (si plusieurs indices sont possibles, on prendra le plus petit) et deux sommets a_i et b_i (d'*attachement*) tels que $V(E_i) \cap V(G_{i-1}) = \{a_i, b_i\} \subseteq V(E_{j_i})$ et E_i induit un chemin entre a_i et b_i . On dit que l'oreille E_i est *attachée* à l'oreille E_{j_i} ;
- Si deux oreilles E_i et $E_{i'}$ sont attachées à une même oreille E_j ($j = j_i = j_{i'}$), elles ne se "croisent" pas. Précisément, soit le sous-chemin P de E_j entre a_i et b_i contient le sous-chemin Q de E_j entre $a_{i'}$ et $b_{i'}$, soit Q contient P , soit P et Q n'ont pas de sommets internes communs.

Pour conclure cette section préliminaire, nous étudions la classe la plus simple des graphes SP 2-connexes, à savoir les *melons*, qui sont obtenus par la composition parallèle de chemins. Bien que cette classe de graphes soit très simple, l'expression de leur longueur arborescente n'est, de façon surprenante, pas triviale.

Un graphe *melon* $G = (P_1, \dots, P_p)$ est obtenu à partir de $p \geq 2$ chemins P_1, \dots, P_p en identifiant une extrémité de chaque chemin en un sommet unique x , puis en identifiant l'autre extrémité de chaque chemin en un sommet y (i.e., si $p = 2$, alors le graphe est un cycle et sinon, x et y sont les seuls sommets de degré au moins 3). Soit $\ell_i = |P_i|$ le nombre d'arêtes de P_i pour tous $1 \leq i \leq p$ et, w.l.o.g., supposons que $\ell_1 \geq \dots \geq \ell_p > 0$. Notons que les cycles isométriques de G sont de la forme $P_p \cup P_i$ pour tout $i < p$. Un plus grand cycle isométrique est ainsi formé par P_1 et P_p , donc $is(G) = \ell_1 + \ell_p$. Pour finir, $P_1 \cup P_2$ est un plus grand cycle de G . Soit $lc(G) = \ell_1 + \ell_2 \geq is(G)$, i.e., $lc(G)$ est la taille d'un plus grand cycle dans G .

Théorème 1. *Pour tout graphe melon $G = (P_1, \dots, P_p)$, $tl(G) = \min\{\lceil \frac{lc(G)}{3} \rceil; \max\{\lceil \frac{is(G)}{3} \rceil; |P_p|\}\}$.*

Idée de la preuve : Le fait que $tl(G) \leq \min\{\lceil \frac{lc(G)}{3} \rceil; \max\{\lceil \frac{is(G)}{3} \rceil; |P_p|\}\}$ vient de constructions explicites de décompositions arborescentes (voir [8]). La borne inférieure est plus instructive. Tout d'abord, rappelons que la longueur arborescente d'un cycle est le tiers de sa longueur, et que la longueur arborescente est close par sous-graphes isométriques. Ainsi, $\lceil \frac{is(G)}{3} \rceil = tl(P_1 \cup P_p) \leq tl(G)$ puisque $P_1 \cup P_p$ est un plus grand cycle isométrique. Supposons que $|P_p| > \lceil \frac{is(G)}{3} \rceil$. Si x et y sont dans un même sac d'une décomposition (T, \mathcal{X}) , alors $length(T, \mathcal{X}) \geq |P_p|$. Sinon, si x et y n'appartiennent pas à un même sac de (T, \mathcal{X}) , alors, par les propriétés des décompositions arborescentes, un sac contenant x doit contenir un séparateur de x et y . Ce sac devra donc contenir x et au moins un sommet de chaque chemin P_i , $i \leq p$. Cela implique que ce sac doit contenir 3 sommets de chaque cycle de G . En particulier, ce sac contient 3 sommets de $P_1 \cup P_2$ (de taille $lc(G)$) et il est possible de montrer qu'au moins 2 d'entre eux sont à distance au moins $\lceil \frac{lc(G)}{3} \rceil$. \diamond

Notons que nous pouvons construire des exemples de graphes SP (pas melons) dont la longueur arborescente ne satisfait pas la formule du Théorème 1 [8].

3 $\frac{3}{2}$ -Approximation et cas de la longueur arborescente égale à 2

Cette section est dédiée à nos résultats principaux. Tout d'abord, nous décrivons une $\frac{3}{2}$ -approximation pour approcher la longueur arborescente de tout graphe SP. Notre résultat principal est une caractérisation (décidable en temps polynomial) sous forme de sous-graphes isométriques interdits des graphes SP de longueur arborescente au plus 2.

Théorème 2. *Pour tout graphe SP G , une décomposition arborescente de G de longueur au plus $\frac{3}{2} \cdot tl(G)$ peut être calculée en temps quadratique.*

Idée de la preuve : Comme dit plus haut, il est possible de se restreindre au cas des graphes SP 2-connexes. Ainsi, supposons que G est 2-connexe et soit (E_0, \dots, E_p) une décomposition isométrique en oreilles imbriquées de G . Notre algorithme repose sur deux observations : (1) $is(G) \leq 3 \cdot tl(G)$ (pour tout graphe G) et (2) chaque oreille E_i , $0 \leq i \leq p$, fait partie d'un cycle isométrique (en effet, soit P un plus court chemin entre a_i et b_i dans G_{i-1} , alors $P \cup E_i$ est un cycle isométrique de G). Par (1), $|V(E_0)| = is(G) \leq 3 \cdot tl(G)$ et la distance entre deux sommets de $V(E_0)$ est au plus $\frac{3}{2} \cdot tl(G)$. D'après (2) et le fait que G_{i-1} est isométrique, la

distance entre deux sommets de E_i est au plus $\frac{3}{2} \cdot t\ell(G)$. La décomposition construite récursivement comme suit a donc une longueur au plus $\frac{3}{2} \cdot t\ell(G)$. Commençons avec un sac contenant $V(E_0)$. Puis, pour $1 \leq i \leq p$, ajoutons un sac contenant $V(E_i)$ adjacent au sac contenant $V(E_{j_i})$. La complexité de l’algorithme vient donc uniquement du calcul de la décomposition isométrique en oreilles imbriquées de G . \diamond

Nous définissons la famille des graphes *Dumbo* comme celle de tout graphe obtenu d’un cycle $C_0 = (v_0, \dots, v_5)$ de taille 6 auquel on ajoute un chemin R de longueur au moins 3 entre v_0 et v_2 , et un chemin L de longueur au moins 3 entre v_3 et v_5 (R et L n’intersectent C_0 qu’en leurs extrémités et sont mutuellement sommet-disjoints). Notons qu’un graphe *Dumbo* est un graphe SP.

Théorème 3. *Soit G un graphe SP. Alors, $t\ell(G) \leq 2$ si et seulement si $is(G) \leq 6$ et G ne contient aucun graphe *Dumbo* comme sous-graphe isométrique. De plus, il existe un algorithme polynomial qui soit calcule une décomposition arborescente de longueur au plus 2 soit exhibe un sous-graphe isométrique interdit.*

Idée de la preuve : Notons que $t\ell(G) = 1$ si et seulement si G est un graphe cordal (sans cycles induits de plus de 3 sommets, ce qui peut être décidé en temps linéaire). Donc supposons que $t\ell(G) > 1$ et, encore une fois, que G est 2-connexe et donc admet une décomposition isométrique en oreilles imbriquées (E_0, \dots, E_p) .

Le “seulement si” vient du fait que la longueur arborescente est close par sous-graphe isométrique (en particulier $t\ell(G) \geq \lceil \frac{is(G)}{3} \rceil$) et du fait que nous prouvons que, pour tout graphe *Dumbo* D , $t\ell(D) > 2$.

Pour le “si”, nous commençons par calculer $is(G)$ (en temps polynomial). Dans le cas où $is(G) \leq 6$, nous concevons un algorithme récursif (sur le nombre d’oreilles) qui, en temps polynomial, soit calcule une décomposition arborescente de longueur 2, soit exhibe un graphe *Dumbo* comme sous-graphe isométrique. Informellement, supposons par induction sur i que, pour $0 \leq i < p$, notre algorithme a calculé une décomposition arborescente (T_i, X_i) de G_i de longueur 2 et telle que, pour toute oreille E_j , $j > i$, de longueur au moins 3 et dont les sommets d’attachement sont dans G_i , alors ces sommets d’attachement sont dans un même sac de (T_i, X_i) . Si E_{i+1} est de longueur 2, nous étendons “directement” la décomposition arborescente (T_i, X_i) à une décomposition (T_{i+1}, X_{i+1}) (avec les propriétés désirées) de G_{i+1} . Sinon, par une analyse de cas sur $|E_{i+1}|$ et $is(G)$, nous montrons que soit (T_i, X_i) peut être étendue à une décomposition (T_{i+1}, X_{i+1}) de G_{i+1} (avec les propriétés désirées), soit un graphe *Dumbo* peut être identifié. \diamond

Travaux futurs. Notre approche (en terme de sous-graphes isométriques interdits) semble difficile à généraliser pour des valeurs de longueur arborescente au moins 3. En effet pour les graphes SP de longueur arborescente 3, nous avons déjà identifié plus de 15 familles infinies de sous-graphes isométriques interdits. Toutes ces familles sont des variations des graphes *Dumbo* ou des graphes melons, mais nous ne savons pas comment les décrire de façon synthétique. La question de la complexité du calcul de la longueur arborescente des graphes SP reste ouverte et, *a fortiori*, dans les graphes planaires également.

Références

- [1] S. Arnborg, D. G. Corneil, and A. Proskurowski. Complexity of finding embeddings in a k-tree. *SIDMA*, 8(2) :277–284, 1987.
- [2] R. Belmonte, F. V. Fomin, P. A. Golovach, and M. S. Ramanujan. Metric dimension of bounded tree-length graphs. *SIDMA*, 31(2) :1217–1243, 2017.
- [3] H. L. Bodlaender and T. Kloks. Efficient and constructive algorithms for the pathwidth and treewidth of graphs. *J. Algorithms*, 21(2) :358–402, 1996.
- [4] Victor Chepoi. Packing and covering δ -hyperbolic spaces by balls. In *In APPROX-RANDOM 2007*, pages 59–73.
- [5] D. Coudert, G. Ducoffe, and N. Nisse. To Approximate Treewidth, Use Treelength! *SIDMA*, 30(3) :13, 2016. AlgoTel 2015.
- [6] B. Courcelle and M. Mosbah. Monadic second-order evaluations on tree-decomposable graphs. *TCS*, 109 :49–82, 1993.
- [7] Y. Dieng and C. Gavaille. On the tree-width of planar graphs. *Electron. Notes Discret. Math.*, 34 :593–596, 2009.
- [8] T. Dissaux, G. Ducoffe, N. Nisse, and S. Nivelles. Treelength of series-parallel graphs. hal.archives-ouvertes.fr/hal-03010346, accepté à LAGOS 2021.
- [9] Y. Dourisboure and C. Gavaille. Tree-decompositions with bags of small diameter. *Discrete Maths.*, 307(16) :2008–2029, 2007.
- [10] D. Eppstein. Parallel recognition of series-parallel graphs. *Inf. Comput.*, 98(1) :41–55, May 1992.
- [11] A. Kosowski, B. Li, N. Nisse, and K. Suchan. k-chordal graphs : From cops and robber to compact routing via treewidth. *Algorithmica*, 72(3) :758–777, 2015. AlgoTel 2012.
- [12] R. Krauthgamer and J. R. Lee. Algorithms on negatively curved spaces. In *FOCS*, pages 119–132. IEEE, 2006.
- [13] D. Lokshtanov. On the complexity of computing treelength. *Discrete Applied Mathematics*, 158 :820–827, 01 2010.
- [14] P. D. Seymour and R. Thomas. Call routing and the ratcatcher. *Comb.*, 14(2) :217–241, 1994.