Décomposition de graphes en plus courts chemins et en cycles de faible excentricité

Léo Planche

UNIVERSITÉ SORBONNE-PARIS-CITÉ

École Doctorale de Sciences Mathématiques de Paris-Centre

Thèse

pour obtenir le titre de

Docteur en sciences

Spécialité : Mathématiques informatique

Présentée et soutenue par

Léo PLANCHE

Décomposition de graphes en plus courts chemins et en cycles de faible excentricité

Thèse dirigée par Etienne BIRMELÉ et
Fabien de MONTGOLFIER

préparée au MAP5 et à l'IRIF

soutenue publiquement le 23 novembre 2018

devant le jury composé de :

Rapporteurs :	Johanne COHEN	-	Université Paris-Sud
	Cyril GAVOILLE	-	Université de Bordeaux
Directeur :	Étienne BIRMELÉ	-	Université Paris Descartes
Co-Directeur :	Fabien de MONTGOLFIER	-	Université Paris Diderot
Examinateurs :	Michel HABIB	-	Université Paris Diderot
	Frédéric HAVET	-	Université Nice-Sophia-Antipolis
	Stéphane VIALETTE	-	Université Paris-Est Marne-la-Vallée

Remerciements

Mes remerciements sont, tout d'abord, dirigés envers Étienne Birmelé et Fabien de Montgolfier pour avoir accepté de diriger cette thèse. Je leur suis reconnaissant pour leurs précieux conseils qu'ils m'ont prodigués ainsi que pour leur soutien, leur disponibilité, et leur bienveillance tout au long de ces années.

J'adresse mes sincères remerciements à Johanne Cohen et à Cyril Gavoille qui ont accepté de s'intéresser à cette thèse en tant que rapporteurs et part leur présence dans le jury de soutenance.

Je remercie également vivement Michel Habib, Frédéric Havet et Stéphane Vialette pour leur présence et participation à mon jury de soutenance.

Mes remerciements envers Michel Habib sont multiples puisque je le remercie également pour son apport dans la collaboration avec l'institut de Biologie Paris-Seine, ses travaux, ainsi que les discussions que nous avons pu avoir sur différents sujets de cette thèse. Je remercie l'équipe « Adaptation, Intégration, Réticulation et Évolution » de l'Institut de Biologie Paris Seine, en particulier Eric Bapteste, Philippe Lopez, Chloé Vigliotti et Guillaume Bernard pour avoir partagé avec nous leurs données, et pour les nombreuses discussions que nous avons pu avoir sur ce sujet. Lors de notre études des graphes de reads, Laurent Viennot fut d'une aide précieuse et a clairement contribué à augmenter l'intérêt de notre travail, en particulier, par son travail et son apport de connaissances quant aux représentations compactes de distances, détaillées dans le chapitre 5.

Il me fut très agréable de travailler au MAP5 pendant ces années, et pour cette raison je remercie l'ensemble de ses membres d'y avoir contribué, en particulier Fabienne Comte pour sa fonction en tant que directrice du laboratoire et toute l'équipe administrative menée par Marie-Hélène Gbaguidi. Je remercie chaleureusement l'équipe des doctorants pour les discutions autour de cafés, rendant chaque journée de travail au MAP5 plus douce.

Mes remerciements vont également au laboratoire IRIF pour son excellent accueil. Mes fonctions en tant que moniteur à l'Université Paris Diderot furent rendues très agréables par la compétence et bienveillance de l'ensemble des membres de l'IRIF et de l'UFR d'informatique.

Enfin je souhaite exprimer ma gratitude aux personnes parmi mes amis et ma famille qui m'ont soutenu pendant ces trois années de thèse.

Table des matières

Chapitre 1

Introduction

Sommaire

1.1 Contexte et motivation de la thèse

L'impulsion première qui a donné naissance aux travaux de recherche présentés dans cette thèse provient d'une collaboration entre l'Institut de Biologie Paris-Seine à Jussieu, plus précisément l'équipe "Adaptation, Intégration, Réticulation et Evolution" et différents chercheurs du laboratoire IRIF. Cette collaboration a pour objectif la compréhension et le traitement d'un jeu de données résultant d'une expérience réalisée dans les années 1970 par [Nevo 1972]. Deux espèces de lézards vivant sur deux îles distinctes, toutes deux au large de la Croatie, furent mélangées afin d'étudier la compétitivité entre elles. Précisément, 10 lézards insectivores de l'espèce *Podarcis sicula* de l'île de Pod Kŏpiste furent introduits sur l'île de Pod Mrčaru. Symétriquement, 10 lézards *Podarcis melisellensis* de l'île de Pod Mrčaru furent introduits sur l'île de Pod Kopište. 35 ans plus tard, une équipe de scientifiques est revenue sur les îles. Ils ont observé que les *Podarcis melisellensis* avaient disparu et que les *Podarcis sicula* sur l'île de Pod Mrčaru étaient devenus omnivores [Herrel 2008]. Une des questions qui se pose alors est l'influence de l'environnement sur le microbiome intestinal des lézards. Peut-on en analysant le microbiome des lézards *Podarcis sicula* déplacés, comprendre leur évolution due au nouvel environnement ? Est-il possible de distinguer les lézards déplacés de ceux des lézards restés sur leur île originelle par la simple étude de leur microbiome intestinal ? Pour répondre à cette question, l'ADN à l'intérieur de l'intestin des lézards fut prélevé et séquencé. Il est alors obtenu

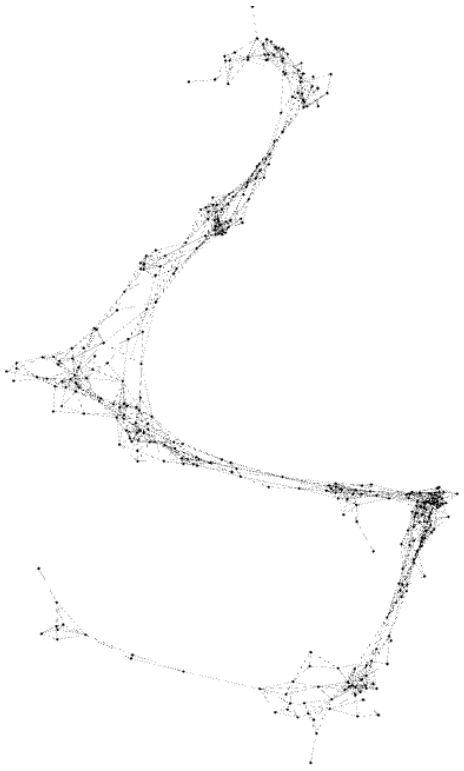

FIGURE 1.1 – Un graphe de read, calculé à partir de l'ADN d'un lézard. Nous appellerons par la suite cette forme de graphe "laminaire".

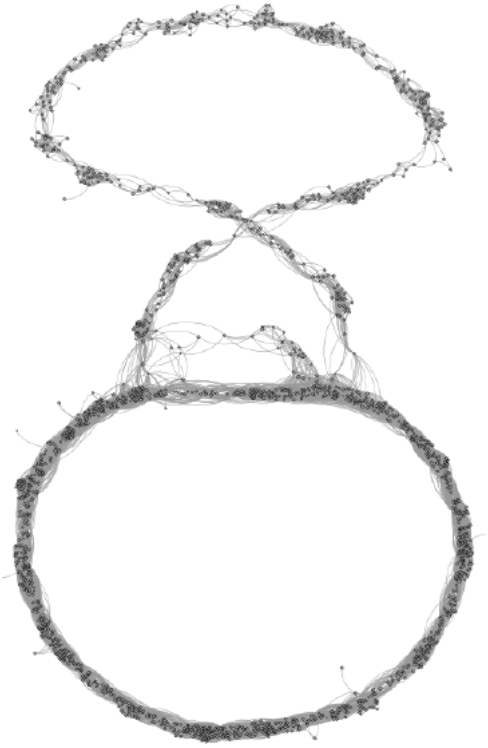

FIGURE 1.2 – Un graphe de read, calculé à partir de l'ADN d'un lézard. Nous appellerons par la suite cette forme de graphe "hub-laminaire".

pour chaque lézard un ensemble de graphes où chaque sommet correspond à un bout d'ADN, nommé *read* et où deux sommets sont reliés s'il y a une similarité suffisante entre leur séquence d'ADN. En visualisant ces graphes on peut noter une structure bien particulière. À savoir, qu'ils sont composés d'un ou plusieurs chemins tels que tous les sommets du graphe en sont proches. Les figures 1.1 et 1.2 en donnent des exemples.

Afin de comprendre les raisons potentielles de la structure bien particulière de ces graphes, présentons, sans rentrer en profondeur, le processus du séquençage d'ADN. Ce dernier consiste à déterminer l'ordre d'enchaînement des nucléotides (A,T,G,C) pour un fragment d'ADN donné. Plusieurs méthodes de séquençage existent, dans tous les cas les fragments d'ADN sont d'abord découpés en séquences de reads de faible taille avant d'être reconstitués. Dans le cas de nos données, ces reads sont de longueur aux alentours de 300.

La séquence d'ADN peut être reconstituée soit en utilisant un génome de référence et en associant à chaque read sa place dans le génome, soit par une méthode *de novo* où l'assemblage se fait sans génome de référence mais uniquement par l'étude de similarité des séquences. C'est cette seconde méthode qui a été utilisée pour générer

les données que nous traitons. En effet, l'ADN étudié provient du microbiome intestinal de lézards et contient donc l'ADN de nombreuses espèces (méta-génomique). Il n'est donc pas possible de le séquencer en le comparant à un génome de référence.

Les méthodes d'assemblages *de novo* se basent généralement sur la recherche de plus long chemin dans des graphes d'overlap ou de de Bruijn [Zerbino 2008]. Un graphe de de Bruijn est un graphe orienté qui permet de représenter les chevauchements de longueur $n-1$ entre tous les mots de longueur n sur un alphabet donné. L'application des graphes de de Bruijn à l'assemblage de read n'est pas directe et nécessite une définition plus souple des graphes de de Bruijn. Premièrement, pour que deux séquences soient adjacentes dans un graphe de de Bruijn il faut qu'elles diffèrent d'au maximum 2 lettres. Plus précisément, un arc d'un mot m à m' est présent si et seulement si $m = ax$ et $m' = xb$ avec a, b deux lettres et x un mot de taille $|m|-1$. Dans le cas des reads de taille 300, cela signifie un chevauchement de taille 299, inatteignable en pratique. Deuxièmement, les reads peuvent contenir des erreurs et une correspondance parfaite de séquence ne peut être espérée. Diverses méthodes existent pour résoudre ces problèmes [Compeau 2011]. Entre autres, deux séquences sont reliées si elles correspondent sur au moins $x\%$ de leur taille. De plus, un taux d'erreur est autorisé, la correspondance n'est pas nécessairement exacte. Dans notre cas, deux reads sont reliés s'ils ont 80% de leur séquence en commun avec 10% d'erreur autorisée sur la correspondance.

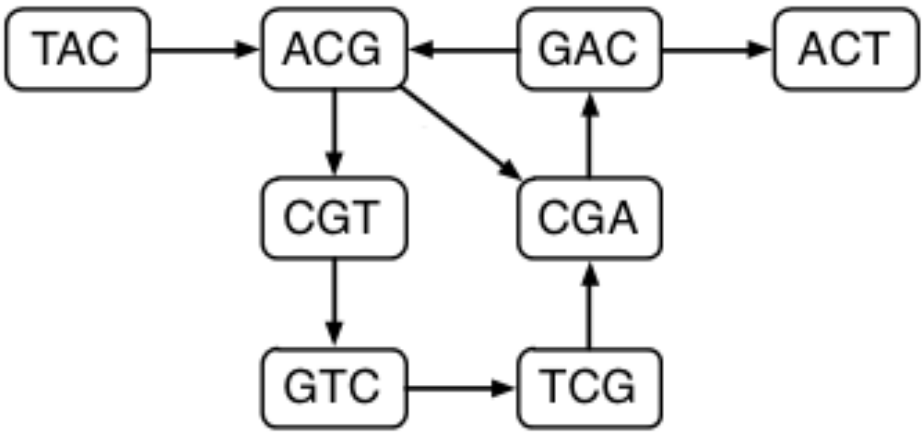

FIGURE 1.3 – Exemple d'un graphe de de Bruijn sur des mots de longueur 3.

Ces graphes sont utiles pour les biologistes car ils permettent une mesure de la diversité génétique d'un environnement par l'analyse de la diversité de l'ensemble de reads correspondant. Quand un ensemble de reads couvre un partie continue du génome, ils s'assemblent en un graphe centré autour d'un chemin (figure 1.1). Nous appellerons ce type de graphes *laminaires*. Ceci définit alors une suite de gènes que les biologistes peuvent ensuite étudier pour mesurer la diversité génétique de la communauté. Cependant au cours de l'évolution, des séquences d'ADN peuvent se déplacer (transposons) ou se dupliquer. Ainsi, une même séquence d'ADN peut se

retrouver dans différents contextes génomiques. Le graphe de similarité résultant est alors un ensemble de laminaires reliés au niveau de ces répétitions (figure 1.2). Reconstruire des séquences génomiques continues et ordonnées (contigs) dans ce dernier cas est une tâche particulièrement difficile, nécessitant de nouvelles méthodes.

Notre jeu de données contient des prélèvements d'ADN effectués sur 24 lézards, 12 pour chacune des deux îles. Pour chaque lézard, le graphe obtenu est de très grande taille, plusieurs millions de sommets, et est divisé en de nombreuses composantes connexes. Pour chaque graphe, ces composantes connexes sont au nombre d'environ 500000 et sont de tailles très variables allant de deux sommets à quelques centaines de milliers. Le nombre important de composantes connexes s'explique principalement par un sous-échantillonnage des données.

L'objectif initial de notre collaboration sur les graphes de read était d'essayer de distinguer, sur une base topologique, ceux provenant des lézards de l'île de Pod Mrčaru de ceux provenant de l'île de Pod Kopište. Déterminer les caractères topologiques qui différencient les lézards ayant changé de régime alimentaire permettrait en effet de pointer des pistes d'études potentielles aux chercheurs en biologie. Nous cherchons donc à déterminer quels graphes possèdent des informations importantes, et où ces dernières sont localisées à l'intérieur de chacun.

Des centaines de milliers de graphes devant être comparés, des méthodes automatisées sont indispensables, et une réduction des graphes de plusieurs centaines de sommets à des structures plus légères est souhaitable pour obtenir ses temps de calcul acceptables et gagner en interprétabilité. L'approche retenue est alors de réduire chaque graphe en un "squelette" rendant au mieux compte de sa topologie globale. Il est à noter que cette approche avait été initiée par Michel Habib et Finn Völkel dans [Völkel 2016], dans le cas des graphes laminaires où le squelette prend la forme d'un chemin.

La suite de cette introduction présente plusieurs méthodes permettant de discriminer les graphes les uns des autres ou de les résumer efficacement. Nous discuterons de leur pertinence quant à l'étude des graphes de reads, avant de proposer un nouveau modèle.

1.2 Réduction de graphes

Les graphes de nos données semblent être caractérisés par un (figure 1.1) ou plusieurs (figure 1.2) chemins tels que chaque sommet en soit proche. Pour cette raison nous nous intéressons particulièrement à la réduction et caractérisation de graphes en chemins.

1.2.1 Largeur d'arbre et de chemin

La décomposition d'un graphe en arbre (*tree decomposition*) ou en chemin (*path decomposition*) est définie par [Robertson 1983, Robertson 1984]. Ces décompositions permettent intuitivement de calculer l'éloignement d'un graphe à un arbre ou chemin. Cet éloignement est appelé respectivement *largeur d'arbre* (*treewidth*)

et *largeur de chemin* (*pathwidth*). Une décomposition en arbre (resp. chemin) d'un graphe G est un arbre (resp. chemin) dont les sommets sont des sous-ensembles (nommés *bags*) de sommet de G. De plus,

— chaque sommet de G est compris dans au moins un des bags.
— Pour toute arête (v, w) de G, il existe au moins un sous-ensemble contenant u et v.
— Pour tout sommet v de G, l'ensemble des bags contenant v forment un sous-arbre (resp. sous-chemin).

En utilisant la programmation dynamique, de nombreux problèmes NP-complets peuvent être résolus en temps polynomial quand la largeur d'arbre ou chemin est bornée [Bodlaender 2008]. Cependant le calcul de la largeur d'arbre et chemin est un problème NP-complet [Arnborg 1987]. A k fixé il est possible de déterminer en temps linéaire si un graphe a une largeur d'arbre/chemin de k [Bodlaender 1996].

La *longueur d'arbre* (*treelength*) d'un graphe est la distance maximale entre deux sommets d'un même bag, minimisée parmi toutes les décompositions en arbre du graphe [Dourisboure 2004]. Un graphe possédant un plus court chemin k-dominant (cf. section suivante) a une *longueur de chemin* d'au plus $4k+1$. Réciproquement, un graphe de longueur de chemin de k a un plus court chemin k-dominant [Leitert 2017].

Notons qu'une clique de taille n a une largeur de chemin de $n-1$ et un plus court chemin 1-dominant. A l'inverse, un cycle de taille n a une largeur de chemin de 2 et ne possède aucun plus court chemin k-dominant avec k inférieur à $\frac{n}{4}$. Ces deux notions qui peuvent sembler liées au premier abord ne le sont donc finalement pas directement.

Si les décompositions en arbre ou en chemin permettent, à largeur fixée, de maîtriser la complexité de nombreux algorithmes, elles imposent la forme de la structure dans laquelle le graphe est résumé. Elle ne permettent donc pas de déterminer un squelette tel que celui qui apparaît clairement sur le graphe de la Figure 1.2. De plus les grands cycles se retrouvent plaqués sur une structure d'arbre ou de chemin alors que nous voudrions les conserver. Et ces décompositions ne disent quasiment rien de la topologie du graphe ou de ses distances. Il apparait par conséquent plus adapté à notre problème d'étudier les squelettes sous la forme d'un nombre réduit de chemins tels que tout sommet est proche de l'un d'eux.

1.2.2 Problèmes de domination

Il existe plusieurs études de caractérisation d'un graphe par un chemin, envisagées sous la forme d'un problème de domination. Les graphes contenant un chemin diamétral dominant le graphe sont étudiés dans [Deogun 1995]. Un ensemble est dit dominant si chaque sommet ou arête du graphe en est voisin. Dans ce cas précis, tout sommet est à distance au plus 1 du chemin diamétral. Nous parlerons de k-domination d'un ensemble quand tout sommet en est à distance au plus k.

La recherche et existence de couples de sommets dans un graphe, tels que chaque chemin entre ces sommets domine le graphe, est étudiée dans [Deogun 2002]. [Bacso 2007] s'intéresse aux graphes tels que chaque sous-graphe induit contient des

plus courts chemins dominants. Les graphes tels que des chemins dominants soient présents dans tout sous-graphe induit sont étudiés dans [Bacso 2007].

Le chemin dominant peut de plus être choisi comme étant diamétral si le graphe est AT-free [Corneil 1997]. Ces graphes, introduits par Lekkerkerker et Boland [Lekkerkerker 1962], ne contiennent pas d'astéroïde triple (AT), c'est-à-dire de triplet de sommets telle que chacune des paires est reliée par un chemin qui évite le voisinage du troisième sommet. Il est à noter que cette classe de graphes contient de nombreuses autres classes telles que les graphes d'intervalles, de permutations, ou de cocomparatibilité. Des algorithmes en temps linéaire de recherche de chemin dominant ou de paire de sommets dominants ont été développés pour les graphes AT-free [Corneil 1995, Corneil 1999].

Si ces propriétés de chemins dominants sont particulièrement intéressantes au regard de nos données issues de la biologie, l'analogie a cependant ses limites. En effet nos graphes ne possèdent pas en général de chemin diamétral 1-dominant comme les graphes AT-free. Beaucoup d'entre eux ont cependant une k-domination diamétrale, avec k supérieur à 1 mais faible. Il faut donc, pour établir des squelettes, étendre les résultats précédents de 1-domination à des résultats de k-domination, ce qui a été initié par [Völkel 2016] . De plus, les graphes contenant plusieurs laminaires s'intersectant (figure 1.2) ne possèdent aucun chemin diamétral k-dominant, avec k faible, et la recherche du squelette reviendra à chercher plusieurs chemins dont l'union k-domine le graphe.

1.3 Approximation des distances

1.3.1 Plongement

Une autre approche visant à résumer un graphe en un chemin est celle du plongement de distorsion minimale d'un graphe dans une ligne. Le problème de plongement d'espaces métriques a été très largement étudié et ce depuis [Assouad 1979]. Un plongement consiste à définir une fonction d'un espace métrique à un autre, généralement en préservant au mieux les distances. Étant donnés (X, d_x), (Y, d_Y) deux espaces métriques, une fonction injective f de X dans Y est appelée *plongement*. Un plongement est *non-contractif* si pour tous u, v de X, $d_X(u, v) \leq d_Y(f(u), f(v))$. La distorsion du plongement est le plus petit λ tel que pour tout u, v de X, $d_Y(f(u), f(v)) \leq \lambda d_X(u, v)$. On cherchera dans la plupart des cas un plongement non-contractif minimisant λ et on parle alors de *plongement de distorsion minimale*. Ce problème possède de nombreuses applications en informatique [B. Tenenbaum 2000], comme en biologie ou chimie [Indyk 2001, Indyk 2004]. En classification, réduire la taille des données en diminuant leur dimension est un problème particulièrement récurrent. L'objectif est alors de trouver des structures de dimension faible caractérisant les données de très grande dimension. Nous cherchons dans notre cas à réduire la dimension des graphes de reads pour caractériser efficacement chaque famille de lézards.

Dans le cas des graphes, un des plongements les plus étudiés est celui dans

les arbres [Fakcharoenphol 2004]. Il n'existe a priori aucune borne garantissant une distorsion faible. Un cycle de taille n ne possède aucun plongement dans un arbre de distorsion meilleure que $\Omega(n)$. Le critère de la distorsion reflétant le pire cas peut être trop discriminant et on préférera parfois étudier la *distorsion moyenne* d'un plongement. Le nom est assez transparent, il s'agit de la moyenne de l'écart multiplicatif des distances sur tout couple $u \neq v$ dans X.

Les graphes laminaires (figure 1.1) étant centrés autour d'un plus court chemin, on s'intéresse particulièrement au plongement dans une ligne. Ce dernier consiste en la recherche d'une fonction bijective f associant à chaque sommet d'un graphe $G(V, E)$ un point d'une ligne ℓ tel que $d_G(x, y) \leq |f(x) - f(y)| \leq \lambda d_G(x, y)$ pour tous sommets x, y de V et minimisant λ. Le plongement de distorsion minimale dans une ligne est un problème NP-complet mais il existe des algorithmes polynomiaux à paramètre fixé (FPT), la distorsion étant le paramètre fixé [Fellows 2013]. Des algorithmes en temps exponentiel et des approximations polynomiales sont proposées dans [Badoiu 2005b]. Plus précisément, leurs algorithmes atteignent une distorsion en $O(\lambda^2)$ pour les graphes généraux et une distorsion de $O(\lambda^{\frac{3}{2}})$ pour les arbres. Dans un autre article [Badoiu 2005a] montrent que le problème est difficile à approximer avec un facteur $O(n^{\frac{1}{12}})$ même dans le cas des arbres. Des algorithmes exponentiels calculant le plongement dans une ligne de distorsion minimale sont présentés dans [Cygan 2012, Fellows 2009, Fomin 2011]. Précisément, [Fomin 2011] montrent qu'un plongement de distorsion minimale peut être calculé en $5^{n+o(n)}$. [Fellows 2009] propose un algorithme en temps $O(n\lambda^4(2\lambda+1)^{2\lambda})$ qui pour un graphe G et un entier λ, calcule un plongement de distorsion au plus λ ou conclut qu'aucun plongement de ce type n'existe. Ainsi, le plongement de distorsion minimale dans une ligne est bien un problème FPT. Plus récemment, [Cygan 2012] ont amélioré l'algorithme de [Fomin 2011] en temps $5^{n+o(n)}$ pour passer en temps $O(4.383^n)$ en échange d'une complexité spatiale plus élevée. Le problème de plongement dans une ligne a été étudié pour des classes de graphes spécifiques par [Heggernes 2010, Heggernes 2008]. En particulier, ils donnent un algorithme en temps polynomial dans les graphes bipartis de permutation et dans les graphes de seuil [Heggernes 2008]. De plus, [Heggernes 2010] montrent que le problème reste NP-difficile pour les graphes bipartis, co-bipartis, triangulés, de co-comparabilité ou AT-free. Ils proposent de plus une approximation polynomiale pour certaines classes de graphes.

1.3.2 Labels de distance

La question du plongement de distorsion minimum d'un graphe dans une ligne est un cas particulier de celle de la représentation compacte des distances dans un graphe [Thorup 2005, Peleg 2000]. Des oracles d'approximation de distances, c'est à dire des représentations compactes approximant les distances, sont étudiés dans [Thorup 2005]. Une approche particulière, introduite par [Peleg 2000] consiste à assigner un label à chaque sommet du graphe de façon à ce que la distance entre deux sommets puisse être estimée par leurs labels. Plusieurs résultats existent quant au

rapport entre la taille des labels et la qualité de l'approximation. Une représentation exacte des distances utilisant les labels est étudiée par [Gavoille 2004] et nécessite des labels de taille $\Omega(n)$ bits pour les graphes généraux. Des approximations à facteur constant avec des tailles de labels sous-linéaires sont étudiées par [Thorup 2005]. Des approximations à facteur additif dans le cas des graphes hyperboliques sont étudiées dans [Gavoille 2005].

1.4 Classification de graphes et plongement dans $\mathbb{R}^n$

1.4.1 Étude des sous-graphes

Il est possible de caractériser les graphes par l'étude de leurs sous-graphes. L'intérêt de cette méthode est multiple mais on peut citer deux applications importantes, le clustering de graphes et la résolution de problèmes difficiles.

Certains problèmes se simplifient quand le graphe considéré ne contient pas certains sous-graphes. Par exemple, une classe célèbre de graphes est celle des graphes sans triangle. Un graphe est sans triangle s'il ne contient pas trois sommets reliés deux à deux par une arête. Ces graphes possèdent de nombreuses propriétés, entre autres, tout graphe planaire sans triangle est 3-colorable [Grotzsch 1958]. La reconnaissance des graphes sans triangle s'effectue en temps $O(m^{1.141})$ [Alon 1997]. Un autre exemple est les graphes sans P_4 (ou cographes) pour lesquels de nombreux problèmes NP-complets deviennent Polynomiaux. Un graphe est sans P_4 si il ne contient pas le chemin sur quatre sommets comme sous-graphe induit. La reconnaissance de ces graphes peut s'effectuer en temps linéaire [Corneil 1985, Habib 2005]. Entre autres, les problèmes de coloration, de recherche de cycle hamiltonien, de coupe maximum, clique maximale et d'autres deviennent polynomiaux (et souvent linéaires) dans le cas des cographes.

1.4.2 Classification de graphes

L'objectif initial de cette thèse est de distinguer les lézards selon leur île par l'étude de leurs graphes de reads associés, ce qui est un problème de classification supervisée. Notons que le terme "clustering de graphes" fait généralement référence à un problème différent, celui de la classification des sommets d'un graphe. Une différence notable entre ces deux problèmes est que s'il existe une distance bien définie entre les sommets d'un graphe, la notion de distance entre deux graphes est plus complexe. Or les méthodes de clustering usuelles nécessitent généralement l'existence d'une distance définie sur l'ensemble des objets à classer. Dans le cas des graphes, on peut en imaginer plusieurs, par exemple la distance d'édition. Cette distance entre deux graphes correspond au nombre d'opérations élémentaires pour passer de l'un à l'autre [Gao 2010, Zhao 2012]. Cependant, quelque soit la définition choisie, son calcul est difficile puisque savoir si deux graphes sont isomorphes, c'est à dire à distance nulle, est déjà un problème difficile dont aucun algorithme de résolution en temps polynomial n'est connu. Pour cette raison, la grande majorité

des algorithmes de clustering de graphes n'utilisent pas la distance exacte entre les graphes mais calculent un plongement dans $\mathbb{R}^n$ avant de les classer.

On peut citer plusieurs méthodes de plongements, la plus courante consistant à classer les graphes suivant la fréquence d'apparition de certains sous-graphes, nommés graphlets [Bandyopadhyay 2006, Deshpande 2005, Huan 2004, Stoica 2009]. [Papadopoulos 1999] utilise l'histogramme des degrés pour classer les graphes, cette méthode trouvant sa justification par le fait que la distance entre les histogrammes de deux graphes est au moins égale à sa distance d'édition. Toute sorte de plongement peut en réalité être pensée, il suffit de chercher des éléments pouvant caractériser les graphes étudiés puis de plonger ces valeurs dans un vecteur de $\mathbb{R}^n$. Ainsi [Li 2012] propose une méthode de clustering utilisant un vecteur contenant des valeurs aussi diverses que le degré moyen, le nombre de sommets, l'excentricité moyenne ou encore la trace de la matrice d'adjacence de chaque graphe.

Ces problèmes de clustering sont en lien direct avec notre application puisque nous cherchons à différencier les graphes les uns des autres : Soit pour les classer comme "Intéressants/Inintéressants", soit pour les classer suivant la famille de lézard auxquels ils appartiennent. Malheureusement l'approche par étude des sous-graphes peut difficilement fournir des résultats satisfaisants dans notre cas. En effet, les graphes de reads nécessitent une caractérisation plus macroscopique, c'est au niveau de leur topologie globale qu'ils peuvent se caractériser. Rien ne semble les distinguer les uns des autres au niveau des sous-graphes de faibles tailles. Notons de plus que la plupart des études de sous-graphes dans le cadre du clustering se font sur des graphes étiquetés. Par exemple dans le cadre des molécules, chaque sommet possède pour label l'atome lui correspondant, ainsi chaque sous-graphe offre plus d'informations que dans le cas général.

1.5 Contributions de cette thèse

Avant d'aborder notre contribution, présentons dans un premier temps les travaux précédemment initiés par la collaboration entre l'Institut de Biologie de Paris-Seine et l'IRIF.

Pour modéliser les graphes de reads, [Völkel 2016] ont défini la classe de graphes dont tous les sommets sont à distance faible d'un chemin diamétral (figure 1.1). Ils ont ainsi proposé la notion de laminarité :

Définition 1 (Laminarité)**.** *Un graphe est k-laminaire s'il possède un chemin diamétral d'excentricité k : tout sommet est à distance au plus k d'un sommet du chemin diamétral.*

Le problème laminaire consiste en la recherche de la laminarité k minimale.

L'excentricité d'un ensemble de sommets est sa distance au sommet dont il est le plus éloigné. Cette définition permet de résumer efficacement les graphes de reads en un unique chemin diamétral et une certaine largeur k, rendant leur analyse et leur lisibilité plus aisées. La recherche du chemin diamétral d'excentricité minimale est

un problème NP-complet, mais on peut décider de la k-laminarité d'un graphe en temps polynomial $O(n^{2k+1})$ [Völkel 2016]. L'étude de la k-laminarité a initialement été pensée pour l'analyse des graphes de reads mais elle peut également être vue comme une extension des problèmes de domination à des problèmes de k-domination.

C'est principalement pour obtenir de nouveaux algorithmes d'approximation du plongement de graphes que [Dragan 2017] ont défini le problème MESP. Ce dernier consiste en la recherche d'un plus court chemin d'excentricité minimale dans un graphe. On peut donc le voir comme une extension du problème laminaire qui se limite à la recherche du chemin diamétral d'excentricité minimale, un chemin diamétral étant par définition un plus court chemin.

Définition 2 (Minimum Eccentricity Shortest Path (MESP))**.** *Etant donné un graphe G, un chemin P d'excentricité minimale est tel que, pour tout plus court chemin Q, $ecc(P) \leq ecc(Q)$.*

Le problème MESP consiste en la recherche d'un plus court chemin d'excentricité minimale.

Le problème MESP est NP-complet mais permet une approximation du problème de plongement de distorsion minimale. Divers algorithmes d'approximation sont proposés par [Dragan 2017], une 8-approximation en temps linéaire, une 3-approximation en temps $O(nm)$ et une 2-approximation en temps $O(n^3)$. De plus, le calcul d'un chemin d'excentricité k peut s'effectuer en temps $O(n^{2k+2})$.

—

Nous développerons au chapitre 2 de nouveaux algorithmes d'approximation du problème MESP. Nous montrerons que l'algorithme de 8-approximation proposé par [Dragan 2017] est en fait une 5-approximation, puis nous développerons une 3-approximation en temps linéaire. Dans un second temps, nous montrerons précisément le lien entre les problèmes MESP et laminaires. Ces résultats ont été présentés à COCOA 2016 [Birmelé 2016].

Afin de préparer le terrain pour la caractérisation des graphes de reads nous définirons dans le chapitre 3 le problème du cycle isométrique d'excentricité minimale. Nous montrerons que ce problème est NP-complet et proposerons des 3-approximations polynomiales.

Nous proposerons une nouvelle décomposition de graphe nommée hub-laminaire dans le chapitre 4. Nous présenterons un algorithme en temps linéaire calculant cette décomposition, sous réserve d'existence. Cette décomposition offre une modélisation des graphes de read et permet d'en extraire les caractéristiques intéressantes en terme de squelette. Nous confronterons l'algorithme à nos données biologiques ainsi qu'à des graphes générés aléatoirement pour en tester les performances empiriques. Ces résultats ont été présentés à ISAAC 2017 [Birmelé 2017].

Pour finir, nous montrerons au chapitre 5 que la décomposition hub-laminaire permet une représentation compacte des distances avec une distorsion additive bornée. Nous montrerons également le lien entre le problème de plongement dans un cercle de distorsion minimale et le problème du cycle isométrique d'excentricité minimale.

Notations et définitions

Nous considérons des graphes finis, non orientés, sans valuations et connexes. L'ensemble des sommets et l'ensemble des arêtes d'un graphe G sont notés respectivement $V(G)$ et $E(G)$. Sauf contre indication et s'il n'y a pas d'ambiguïté quant au graphe concerné, n désigne le nombre de sommets de G et m le nombre d'arêtes.

Un *chemin* P est une séquence de sommets telle que deux sommets consécutifs sont reliés par une arête. Nous traiterons uniquement de chemins simples, c'est à dire tels que chaque sommet n'apparait au plus qu'une fois dans la séquence. Les *extrémités* d'un chemin désignent le premier et le dernier sommet de la séquence. Afin de simplifier les notations, P pourra également désigner l'ensemble des sommets apparaissant dans la séquence, ou encore l'ensemble des arêtes reliant les sommets de la séquence. La *longueur* d'un chemin est le nombre d'arêtes dans sa séquence. Un chemin est un *plus court chemin* s'il n'existe aucun chemin de longueur plus faible et possédant les deux mêmes extrémités. Pour tout sommet u et v de P, nous désignons par P_{uv} le *sous-chemin* de P ayant u et v pour extrémités.

Nous notons $d_G(u, v)$ la *distance* entre deux sommets, c'est à dire la longueur d'un plus court chemin les reliant. S'il n'y a pas d'ambiguïté quant au graphe considéré, nous omettons l'annotation $_G$ et écrivons simplement $d(u, v)$. On désigne par $B(u, r) = \{v \in V(G) \mid d(u, v) \leq r\}$ la *boule* de centre u et de rayon r. Soit U un ensemble de sommets, on note $B(U, r) = \cup_{u \in U} B(u, r)$. Soit U et W deux ensembles de sommets, on dit que U *k-domine* W si tout sommet de W est à distance au plus k d'un sommet de U, c'est à dire $W \subseteq B(U, k)$. Un ensemble U est d'*excentricité* k, si k est le plus petit entier tel que $B(U, k) = V(G)$ et on note $ecc(U) = k$.

Un *cycle* C d'un graphe G est un chemin tel que les deux extrémités sont identiques. Un cycle est un *cycle isométrique* s'il préserve les distances dans le graphe, c'est-à-dire que pour u et v sommets de C, $d_G(u, v) = d_C(u, v)$. En d'autres termes, pour toute paire de sommets du cycle, l'un des deux chemins les reliant dans le cycle est un plus court chemin. Un *cycle simple* est un cycle où chaque sommet n'apparait qu'une fois, hormis les deux extrémités qui sont elles toujours identiques.

Définissons ici l'algorithme de parcours en largeur — noté *BFS*, pour *Breadth First Search* — d'un graphe qui sera utilisé dans plusieurs chapitres. Le traitement de l'ensemble des sommets d'un graphe connexe par un algorithme BFS à partir d'une racine r se déroule comme suit :

1. Créer une liste et y ajouter r.
2. Retirer le premier sommet de la liste et le traiter.
3. Ajouter à la liste tous les voisins non déjà traités.
4. Si la file n'est pas vide, recommencer à l'étape 2.

Chapitre 2

Chemins de faible excentricité

Sommaire

2.1 Introduction

La structure laminaire de certains graphes de read implique assez naturellement une caractérisation des graphes par un de leur chemin de faible excentricité. C'est ce que proposent [Völkel 2016] en définissant le problème de laminarité d'un graphe comme la recherche de son chemin diamétral de plus faible excentricité. Le diamètre d'un graphe étant la plus grande distance séparant deux sommets du graphe, un chemin diamétral est un plus court chemin de taille maximale.

Définition 3 (Laminarité)**.** *Un graphe G est*

— *l-laminaire si G a un diamètre d'excentricité au plus l.*

— *s-fortement laminaire si tout diamètre à une excentricité d'au plus s.*

$l(G)$ et $s(G)$ sont les valeurs minimales de l et s tel que G soit l-laminaire et s-fortement laminaire.

Parallèlement [Dragan 2017] ont défini le problème du plus court chemin de plus faible excentricité, notamment pour son lien avec le problème du plongement de graphe dans une ligne [Yan 2007].

Définition 4 (Minimum Eccentricity Shortest Path (MESP))**.** *Soit un graphe G. Un plus court chemin P d'excentricité minimale est tel que pour tout plus court chemin Q, $ecc(P) \leq ecc(Q)$.*

On note $k(G)$ l'excentricité d'un chemin de plus faible excentricité de G.

Le problème MESP consiste en la recherche d'un plus court chemin d'excentricité minimale.

Le problème est NP-complet [Dragan 2017] et nécessite des algorithmes d'approximation en temps polynomial. Un algorithme est une α-approximation du problème MESP si quelque soit le graphe en entrée, le chemin en sortie est d'excentricité au plus $\alpha k(G)$.

Une 2-approximation en temps $O(n^3)$, une 3-approximation en $O(nm)$ et une 8-approximation linéaire ont toutes trois été développées par [Dragan 2017]. Cette dernière consiste en une simple procédure de double-BFS qui se déroule comme suit :

1. Choisir arbitrairement un sommet r
2. Calculer un BFS démarrant sur r et finissant sur un sommet x. x est alors à distance maximale de r.
3. Calculer un BFS démarrant sur x et finissant sur y.

Le résultat de l'algorithme est un plus court chemin entre x et y. Il s'agit entre autre d'une 2-approximation du diamètre, très efficace en pratique [Corneil 2003]. Cette procédure a été développée pour la première fois par [Handler 1973].

Dans ce chapitre, nous montrons qu'un double-BFS est une 5-approximation du problème MESP et que la borne est atteinte. Nous présenterons ensuite un nouvel algorithme de 3-approximation du MESP en temps linéaire. Enfin dans une dernière partie nous comparerons les problèmes MESP et Laminaire et établirons des bornes les liants. Ces résultats ont été publiés dans [Birmelé 2016].

2.2 Approximation en temps linéaire du MESP

2.2.1 Résultat préliminaire

Afin de prouver nos algorithmes d'approximation, un résultat liminaire que nous utiliserons régulièrement est nécessaire. Pour un chemin quelconque dans un graphe, ce lemme permet de déterminer un ensemble de sommets k-dominé par le chemin.

Lemme 1. *Soit G un graphe, Q un chemin quelconque et $P = v_0, v_1, ..., v_l$ un plus court chemin d'excentricité k. Soient i et j $(i \leq j)$, tels que v_i et v_j soient à distance au plus k de Q.*

Alors, tout sommet de $P_{v_{i-k}v_{j+k}}$ est à distance au plus $2k$ de Q. Par conséquent, tout sommet de G à distance au plus k de $P_{v_{i-k}v_{j+k}}$ est à distance au plus $3k$ de Q.

Ce lemme peut au premier regard sembler similaire au suivant :

Lemme 2 (Dragan *et al.* [Dragan 2017])**.** *Si G possède un plus court chemin de s à t d'excentricité au plus k alors tout chemin Q contenant s et tel que $d(s,t) \leq max_{v \in Q} d(s,v)$ est d'excentricité au plus $3k$.*

Cependant, alors que le k du lemme 2 est spécifique à un couple de sommets (s,t), le k du lemme 1 est global. A l'inverse, le lemme 2 donne une borne sur l'excentricité du chemin étudié en rapport au graphe alors que le lemme 1 offre une borne sur l'excentricité en rapport à un sous-graphe.

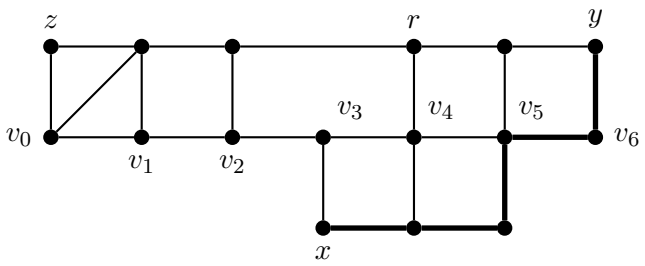

FIGURE 2.1 – La borne montrée au théorème 1 est atteinte. En effet, $v_0, v_1, ..., v_6$ est un plus court chemin d'excentricité 1. Le sommet z est à distance 5 du plus court chemin (arêtes en gras) entre les sommets x et y calculés par un double-BFS commencant sur r.

Démonstration du lemme 1. La seconde affirmation du lemme découle directement de la première.

Par l'absurde, supposons qu'il existe un sommet w dans de $P_{v_{i-k}v_{j+k}}$ qui est à distance plus de $2k$ de Q. Notons q_i (resp. q_j) un sommet de Q à distance au plus k de v_i (resp. v_j). Pour tout sommet $q_x \in Q$, définissons v_x comme un sommet de P tel que $d(q_x, v_x) \leq k$.

Il est clair que w n'appartient pas à $P_{v_{i-k}v_i}$ ou $P_{v_jv_{j+k}}$, sinon il serait à distance au plus $2k$ de q_i ou q_j, supposons le donc dans $P_{v_iv_j}$.

Comme v_i, w et v_j sont dans cet ordre dans P, il deux sommets consécutifs q_a et q_b dans $Q_{q_iq_j}$ et tel que v_a, w et v_b sont dans cet ordre dans P.

Mais w est à distance plus de $2k$ de q_a et q_b, donc $P_{v_av_b}$ est de taille au moins $2k+2$. Comme $d(v_a, v_b) \leq d(v_a, q_a) + d(q_a, q_b) + d(q_b, x_b) \leq 2k+1$, cela contredit le fait que P est un plus court chemin.

□

De plus, la borne est atteinte comme le montre la figure 2.3.

2.2.2 Un double BFS est une 5-approximation

Soit G un graphe et $Q = x, ..., y$ le résultat d'un double-BFS sur G, commençant sur un sommet r arbitrairement choisi, atteignant x puis y. Soit $P = v_0, v_1, ..., v_t$ un des plus courts chemins de G d'excentricité minimale k. Nous montrons dans cette section que Q est d'excentricité au plus $5k$.

Théorème 1. *Un double-BFS est une 5-approximation linéaire du problème MESP.*

Démonstration. Montrons que Q est un chemin d'excentricité au plus $5k$.

Soit i (resp. j) tel que v_i (resp. v_j) est à distance au plus k de r (resp. x). Les inégalités suivantes sont vérifiées :

$$d(r, x) \geq d(r, v_t) \geq d(v_i, v_t) - d(r, v_i) \geq d(v_i, v_t) - k \tag{2.1}$$

$$d(r,x) \leq d(r,v_i) + d(v_i,v_j) + d(v_j,x) \leq d(v_i,v_j) + 2k \tag{2.2}$$

En combinant ces inégalités :

$$d(v_i,v_t) - 3k \leq d(v_i,v_j) \tag{2.3}$$

De manière symétrique :

$$d(v_i,v_0) - 3k \leq d(v_i,v_j) \tag{2.4}$$

Ainsi v_j est à distance au plus $3k$ de v_0 ou v_t. Sans perte de généralité, supposons v_j à distance au plus $3k$ de v_0.

Soit l tel que v_l est à distance au plus k de y. On distingue deux cas :

(i) $l \leq j$:

Alors y est à distance au plus $5k$ de x. Comme y est un sommet à distance maximale de x, x $5k$-domine le graphe. Le lemme est vérifié.

(ii) $l > j$:

En appliquant à (x,y) les inégalités 2.3 établies en début de preuve :

$$d(v_j,v_t) - 3k \leq d(v_j,v_l) \tag{2.5}$$

Comme l est supérieur à j, il suit :

$$d(v_l,v_t) \leq 3k \tag{2.6}$$

La figure 2.2 montre la configuration du graphe dans ce dernier cas. Les sommets à distance au plus k d'un sommet v_s tel que $s \leq j$ (resp. $s \geq l$) sont à distance au plus $5k$ de x (resp. y).

On sait par le lemme 1 que tout sommet à distance au plus k d'un sommet v_s tel que s soit entre j et l est à distance au plus $3k$ de tout chemin entre x et y. Le lemme est donc vérifié.

□

La figure 2.1 montre que la borne est atteinte.

2.2.3 Un algorithme de 3-approximation en temps linéaire

Nous montrons dans cette partie qu'en calculant quelques BFS supplémentaires on peut obtenir une 3-approximation du problème MESP, toujours en temps linéaire.

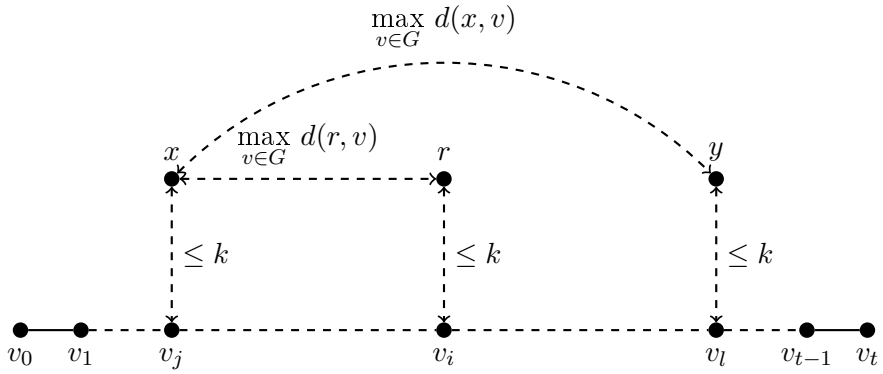

FIGURE 2.2 – Notations utilisées dans la preuve du lemme 1

Soit *meilleurChemin* et *meilleureExc* des variables globales utilisées pour contenir le meilleur chemin calculé et son excentricité. *meilleurChemin* n'est pas initialisé et *meilleureExc* l'est avec la valeur $|V(G)|$.

```
1 Algorithme3k
  Entrée : G graphe, x,y sommets de G, etape entier
2 Calculer un plus court chemin Q entre x et y;
3 Calculer z un sommet à distance maximale de Q;
4 Si d(Q,z) < meilleureExc alors
5 |   meilleurChemin ← Q;
6 |   meilleureExc ← d(Q,z);
7 Si etape < 8 alors
8 |   Algorithme3k(G,x,z,etape + 1);
9 |   Algorithme3k(G,y,z,etape + 1);
```

Algorithme 1: Pseudo-code de la fonction *Algorithme3k*

Théorème 2. *Une 3-approximation du problème MESP peut être calculée en temps linéaire, en considérant x et y deux sommets opposés retournés par un double-BFS et en calculant Algorithme3k(G,x,y,0).*

Démonstration. Soit G un graphe, x et y deux sommets de G, Q_{xy} un plus court chemin entre eux. Soit $i^{x,y}_{min}$ (resp. $i^{x,y}_{max}$) le plus petit (resp. plus grand) entier tel que $v_{i^{x,y}_{min}}$ (resp. $v_{i^{x,y}_{max}}$) est à distance au plus k de x ou y. Alors par le lemme 1,

$$\text{pour tout } j \text{ tel que } i^{x,y}_{min} - k \leq j \leq i^{x,y}_{max} + k, \quad d(Q_{xy}, v_j) \leq 2k \tag{2.7}$$

Ainsi, si $i^{x,y}_{min} \leq k$ et $i^{x,y}_{max} \geq t - k$, tout sommet de P est à distance au plus $2k$ de Q_{xy}. P étant d'excentricité k, Q_{xy} est alors d'excentricité au plus $3k$.

Algorithme3k utilise cette implication pour trouver une paire de sommets (x, y) telle que Q_{xy} est d'excentricité au plus $3k$. En effet, à chaque appel récursif de l'algorithme, une de ces propriétés est vérifiée :

1. Le sommet z sélectionné à la ligne 3 est à distance inférieure ou égale à $3k$ de Q_{xy}. Dans ce cas, *meilleurChemin* va être actualisé en Q_{xy} à moins qu'il contienne déjà un chemin d'excentricité égale ou plus faible. Dans tous les cas, le résultat de l'algorithme est un chemin d'excentricité au plus $3k$.
2. Le sommet z est à distance plus de $3k$ de Q_{xy}. Soit i_z le sommet tel quel v_{i_z} est à distance au plus k de z. Alors suivant l'équation (2.7),

$$i_z \leq i^{x,y}_{min} - k \text{ ou } i_z \geq i^{x,y}_{max} + k \tag{2.8}$$

   (a) Supposons que $i_z \geq i^{x,y}_{max} + k$. Alors dans le cas où $d(v_{i^{x,y}_{min}}, x) = k$, nous avons $i^{x,z}_{min} \leq i^{x,y}_{min}$ et $i^{x,z}_{max} \geq i^{x,y}_{max} + k$. Dans le cas $d(v_{i^{x,y}_{min}}, y) = k$, nous avons $i^{y,z}_{min} \leq i^{x,y}_{min}$ et $i^{y,z}_{max} \geq i^{x,y}_{max} + k$.

   (b) Si $i_z \leq i^{x,y}_{min} - k$, un raisonnement symétrique montre que nous avons $i^{x,z}_{min} \leq i^{x,y}_{min} - k$ et $i^{x,z}_{max} \geq i^{x,y}_{max}$ ou $i^{y,z}_{min} \leq i^{x,y}_{min} - k$ et $i^{y,z}_{max} \geq i^{x,y}_{max}$.

Ainsi, soit *meilleurChemin* contient déjà un chemin d'excentricité au plus $3k$, soit un des deux nouveaux appels récursifs s'effectue avec un couple (x', y') tel que l'intervalle $[i^{x',y'}_{min}, i^{x',y'}_{max}]$ contient $[i^{x,y}_{min} - k, i^{x,y}_{max}]$ ou $[i^{x,y}_{min}, i^{x,y}_{max} + k]$.

Considérons maintenant la paire (x_0, y_0) obtenue par le double-BFS étudié au Théorème 1. Il découle des cas (i) et (ii) de la preuve du Théorème 1 que,

$$i^{s,l}_{min} \leq 5k \text{ et } i^{s,l}_{max} \geq t - 5k \tag{2.9}$$

A chaque appel récursif, si aucun chemin d'excentricité au plus $3k$ n'a déjà été découvert, un des nouveaux appels augmente la taille de l'intervalle $[i^{x,y}_{min}, i^{x,y}_{max}]$ d'au moins k tout en contenant l'intervalle précédent. Les appels récursifs étant effectués jusqu'à $\textit{étape} = 8$, il suit que soit un chemin d'excentricité au plus $3k$ a été découvert, soit un des chemins Q_{xy} explorés correspond à un agrandissement de taille au moins $8k$ de l'intervalle $[i^{s,l}_{min}, i^{s,l}_{max}]$.

Dans ce cas, l'équation (2.9) implique que le couple (x, y) respecte $i^{x,y}_{min} \leq k$ et $i^{x,y}_{max} \geq t - k$. Ainsi tout sommet de P est à distance au plus $2k$ de Q_{xy} et Q_{xy} est d'excentricité au plus $3k$.

□

Démonstration de la complexité. L'algorithme calcule deux arbres BFS aux lignes 2 et 3, ce qui prend un temps $\mathcal{O}(n + m)$. Le reste des opérations s'effectuent en temps linéaire.

La largeur de la récursivité est de deux et sa profondeur de 8. L'algorithme est ainsi appellé 255 fois. Ainsi la complexité globale de l'algorithme est $\mathcal{O}(n + m)$.

□

Démonstration du fait que la borne de l'approximation est atteinte. La figure 2.3 montre un graphe pour lequel l'algorithme renvoie un chemin d'excentricité $3k(G)$ (voir légende). □

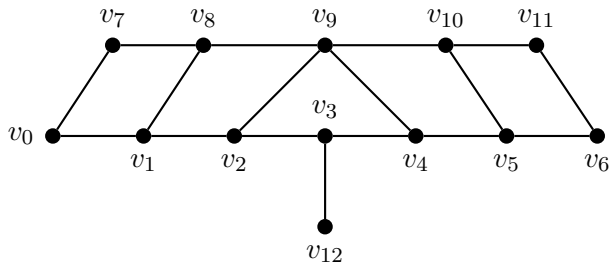

FIGURE 2.3 – La borne de l'algorithme 2 est atteinte. L'algorithme peut en effet boucler entre les paires de sommets suivantes : $(v_0, v_6), (v_0, v_{12}), (v_6, v_{12}), (v_0, v_{11}), (v_{11}, v_{12}), (v_6, v_7), (v_7, v_{12}), (v_{11}, v_7)$. A chaque étape l'algorithme peut choisir un chemin d'excentricité 3 (passant par v_8 v_9 et v_{10} quand v_{12} n'est pas un sommet en bout de chemin) alors que $v_0..v_3..v_6$ est d'excentricité 1. On peut vérifier que ce résultat reste valide en transformant chaque arrête en chemin simple de longeur k. Ainsi la borne est atteinte pour tout entier k supérieur à 0.

2.3 Liens entre les problèmes MESP et Laminaire

Dans cette partie nous étudions le lien entre le problème MESP et la notion de laminarité introduite par [Völkel 2016]. Ce lien a déjà été étudié dans le cas de la domination. Le problème de l'existence d'un plus court chemin d'excentricité 1 et le problème 1-laminaire sont montrés équivalents dans [Deogun 1995]. Ils affirment que l'on peut construire à partir de tout plus court chemin dominant, un chemin diamétral dominant. Cependant, nous montrons que ce résultat est incorrect en exhibant un graphe possédant un plus court chemin dominant mais aucun diamètre dominant.

Théorème 3. *Pour tout graphe G,*

$$k(G) \leq l(G) \leq 4k(G) - 2$$
$$k(G) \leq s(G) \leq 4k(G)$$

De plus, il existe trois suites de graphes $(G_k)_{k\geq 1}$, $(H_k)_{k\geq 1}$ et $(J_k)_{k\geq 1}$ tels que, pour tout k,

- $k(G_k) = l(G_k) = s(G_k) = k$;
- $k(H_k) = k$ *et* $l(H_k) = 4k - 2$;
- $k(J_k) = k$ *et* $s(J_k) = 4k$;

Les bornes données par les inégalités sont donc atteintes.

Lemme 3. *Pour tout graphe G,*

$$k(G) \leq l(G) \leq s(G)$$

Démonstration. Tout diamètre étant par définition un plus court chemin, ces inégalités sont immédiates. L'excentricité d'un diamètre est toujours au minimum $k(G)$. □

Lemme 4. *Pour tout graphe G,*

$$s(G) \leq 4k(G)$$

Démonstration. Soit $D = x_0, x_1, ..., x_s$ un diamètre de G et $P = v_0, v_1, ..., v_t$ un plus court chemin d'excentricité k. Nous allons montrer que D est d'excentricité au plus $4k$. Soit z un sommet de G quelconque. Soit v_i un sommet de P à distance au plus k de z. Distinguons trois cas :

- Cas 1 : il existe des sommets v_a, v_b dans P tels que $a \leq i \leq n$ à distance au plus k de D. Par le lemme 1, z est à distance au plus $3k$ de D.
- Cas 2 : il n'existe pas de sommet v_a dans P tel que $a \leq i$ et $d(v_a, D) \leq k$.
- Cas 3 : il n'existe pas de sommet v_a dans P tel que $i \leq a$ et $d(v_a, D) \leq k$.

Sans perte de généralité, étudions le cas 2 (illustré par la figure 2.4) qui est un cas symétrique du 3. Soit l (resp. m) tel que v_l (resp. v_m) est à distance au plus k de x_0 (resp. x_s), supposons $l \leq m$:

$$d(v_l, v_m) \geq d(x_0, x_s) - 2k \tag{2.10}$$

D étant un diamètre,

$$d(x_0, x_s) \geq d(v_0, v_t) \tag{2.11}$$

En combinant ces inégalités,

$$d(v_l, v_m) \geq d(v_0, v_t) - 2k \tag{2.12}$$

$$d(v_l, v_m) \geq d(v_0, v_i) + d(v_i, v_l) + d(v_l, v_m) + d(v_m, v_t) - 2k \tag{2.13}$$

$$2k \geq d(v_i, v_l) \tag{2.14}$$

Il suit que z est à distance au plus $4k$ de x_0. □

Lemme 5. *Pour tout graphe G,*

$$l(G) \leq 4k(G) - 2$$

Démonstration. Soit $D = x_0, x_1, ..., x_s$ un diamètre de G et $P = v_0, v_1 ..., v_t$ un plus court chemin d'excentricité k. Nous allons montrer que D est d'excentricité au plus $4k-2$ ou que G contient un diamètre D' d'excentricité au plus $3k$. Si P est un diamètre nous avons le résultat. Supposons P de taille au plus $|D| - 1$.

La figure 2.4 illustre alors les notations utilisées par la suite.

Soit z un sommet quelconque de G et v_i un sommet de P à distance au plus k de z. Distinguons les trois mêmes cas que dans la preuve précédente. Du premier cas suit $d(z, D) \leq 3k$. Le second et troisième cas étant symétriques, supposons sans perte de généralité qu'il n'existe pas de sommet v_a de P à distance k de D tel que $a \leq i$.

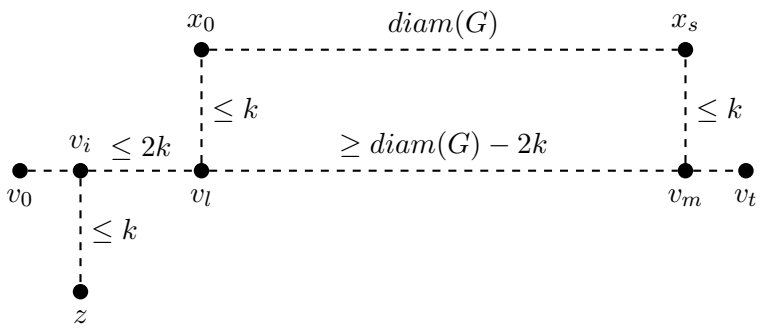

FIGURE 2.4 – Notations utilisées dans la preuve du Lemme 5

Soit v_l (resp. v_m) un sommet de P à distance au plus k de x_0 (resp. x_s). Alors,

$$d(v_l, v_m) \geq |D| - 2k. \tag{2.15}$$

Distinguons deux sous-cas :

• Cas 2.1 : $d(v_l, v_m) > |D| - 2k$,

$$d(v_i, v_l) \leq d(v_0, v_t) - d(v_l, v_m) \leq (|D| - 1) - (|D| - 2k + 1) \leq 2k - 2 \tag{2.16}$$

Il suit que z est à distance au plus $4k - 2$ de D.

• Cas 2.2 : $d(v_l, v_m) = |D| - 2k$

Dans ce cas, le chemin $D' = x_0, ..., v_l, v_{l+1}, ..., v_m, ..., x_s$ est un diamètre. Supposons $l \leq m$, l'équation 2.14 de la précédente preuve montre :

$$d(v_i, v_l) \leq 2k \tag{2.17}$$

et de manière symétrique,

$$d(v_m, v_t) \leq 2k \tag{2.18}$$

Il suit que tout sommet v de G à distance au plus k d'un sommet v_a tel que $a \leq l$ (resp. $a \geq m$) est à distance au plus $3k$ de v_l (resp. v_m). Donc à distance au plus $3k$ de D'. $v_l, v_{l+1}, ..v_m$ étant un sous-chemin de D', tout sommet v de G à distance au plus k d'un sommet v_a avec a entre l et m est à distance au plus k de D'. Finalement, tout sommet de G est à distance au plus $3k$ de D'.

□

Démonstration du fait que les bornes sont atteintes. Soit le graphe G_k réduit à un chemin P de taille $4k$ tel qu'un second chemin de taille k est attaché au milieu. P est alors simultanément le seul diamètre et un MESP. Il k-domine G_k mais ne le $(k-1)$-domine pas. Ainsi les inégalités $k(G) \leq l(G)$ et $k(G) \leq s(G)$ sont atteintes.

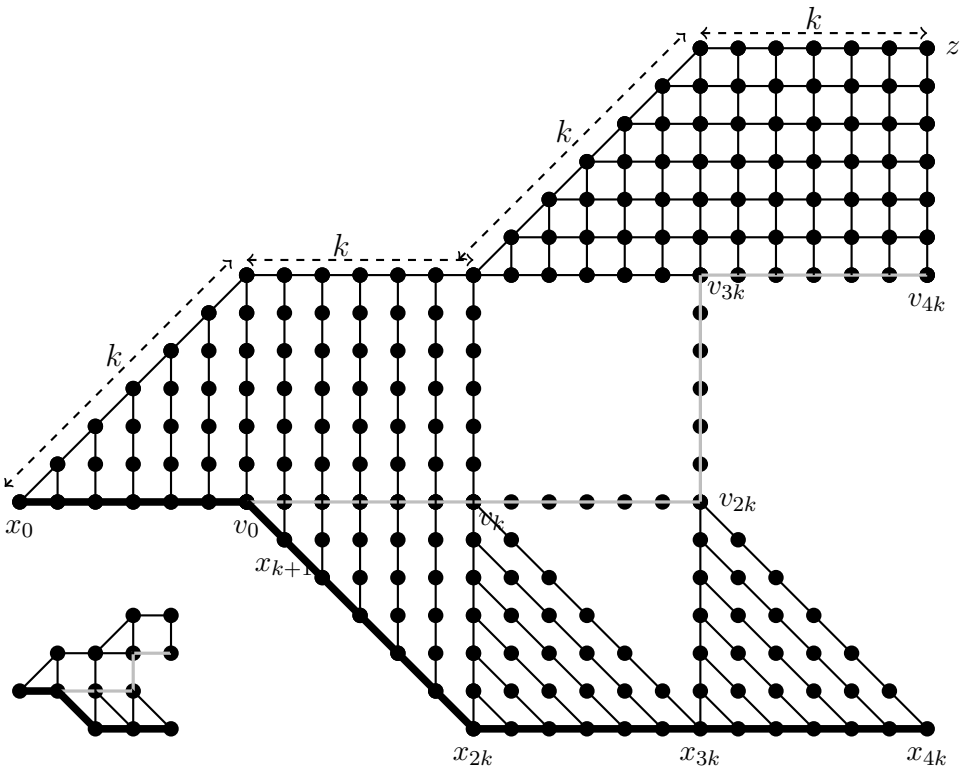

FIGURE 2.5 – Preuve que la born $s(G) \leq 4k(G)$ est atteinte. Le chemin épais $x_0, x_1, ..., x_{4k}$ est un diamètre de taille $4k$ et à distance $4k$ de z ; tandis que le chemin gris $v_0, v_1, ..., v_{4k}$ est un plus court chemin (diamétral) d'excentricité k. Le grand graphe est J_6 (de la suite de graphes $(J_k)_k$ du théorème 3) et le petit en bas à gauche J_1. Les autres membres de la suite peuvent facilement en être déduits.

.

La figure 2.5 montre comment créer une suite de graphes $(J_k)_{k\geq 1}$ (seuls J_1 et J_6 sont dessinés). J_k est un graphe avec un plus court chemin d'excentricité k et avec un diamètre d'excentricité $4k$. L'inégalité $s(G) \leq 4k(G)$ est donc atteinte.

La figure 2.6 montre comment créer une suite de graphes $(H_k)_{k\geq 1}$ (seuls H_1, H_2 et H_6 sont dessinés). H_k est un graphe avec un plus court chemin d'excentricité k et avec un unique diamètre d'excentricité $4k-2$ (H_1 est un cas spécial avec deux diamètres). L'inégalité $l(G) \leq 4k(G) - 2$ est donc atteinte.

□

Nous avons étudié le problème du plus court chemin d'excentricité minimale pour le cas de graphes généraux et proposé une 3-approximation en temps linéaire. De plus nous avons établi des bornes liant le problème MESP au problème de k-laminarité.

On notera que notre algorithme d'approximation du problème MESP consiste à chercher une bonne paire de sommets dans le graphe puis à calculer arbitrairement

un plus court chemin entre eux. En se contentant de cette approche il semble difficile d'obtenir un meilleur résultat qu'une 3-approximation. En effet, comme montré par [Völkel 2016], il existe des graphes tels que la solution au MESP soit un plus court chemin d'excentricité k entre deux sommets s et t et tels qu'il existe également des plus courts chemins entre s et t d'excentricité $3k$.

Quant au problème de laminarité, le calcul de $l(G)$ est NP-complet alors que le calcul de $s(G)$ peut être effectué en temps $O(n^2 m \log n)$ [Völkel 2016]. Il pourrait donc être intéressant de chercher un algorithme d'approximation pour le problème de laminarité. Les méthodes linéaires telles que le BFS, ne peuvent cependant pas être utilisées car il n'est pas possible de calculer $diam(G)$ plus rapidement qu'avec

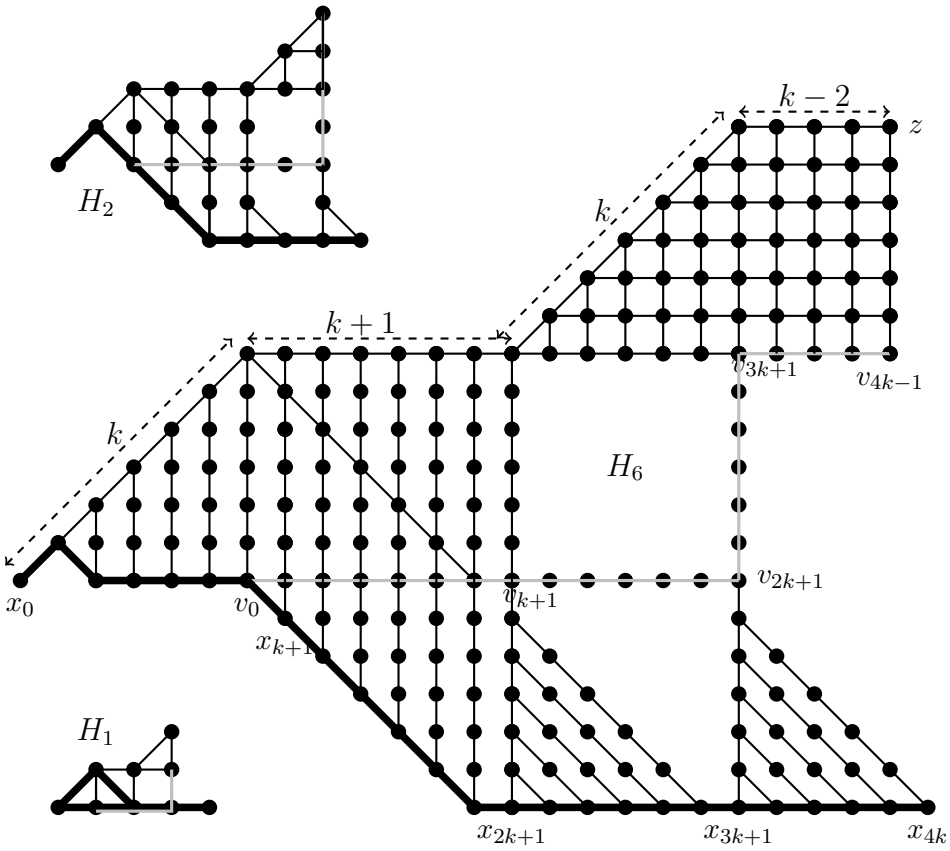

FIGURE 2.6 – Preuve que la borne $l(G) \leq 4k(G) - 2$ est atteinte. La suite de graphe $(H_k)_k$, utilisant les notations du théorème 3. Pour $k \geq 2$, le chemin épais $x_0, x_1, ...x_{4k}$ est l'unique diamètre du graphe. Sa taille est $4k$ et il est à distance $4k-2$ de z. Le chemin gris $v_0, v_1, ...v_{4k-1}$ est un plus court chemin de taille $4k-1$ et d'excentricité k. Les graphes H_2 et H_6 sont ici représentés mais tout graphe H_k avec $k \geq 2$ peut facilement être déduit de H_6. Le petit graphe en bas à gauche est H_1 qui ne respecte ce shèma. Il possède exactement deux diamètres d'excentrité 2 (chemins épais) et un plus court chemin d'excentricité 1 (en gris).

un produit de matrices. De nouvelles techniques devront donc être employées.

Chapitre 3

Cycle isométrique de faible excentricité

Sommaire

3.1 Introduction

Une généralisation naturelle du problème de plus court chemin d'excentricité minimale est celui de la recherche du cycle isométrique d'excentricité minimale (MEIC). Un cycle isométrique est un cycle préservant les distances entre ses sommets.

Définition 5 (Cercle isométrique)**.** *Soit G un graphe et C un cycle dans ce graphe. C est un cercle isométrique si et seulement si, pour tout sommet u, v de C : $d_G(u,v) = d_C(u,v)$.*

En d'autres termes, pour toute paire de sommets sur le cycle, un des chemins du cycle les reliant est un plus court chemin.

Définition 6 (Problème MEIC)**.** *Étant donné un graphe G, trouver un cycle isométrique C tel que pour tout cycle isométrique D dans G : $ecc(C) \leq ecc(D)$.*

La résolution du MEIC permet le plongement de graphe dans un cercle, ce que nous développerons au chapitre 5.

Nous montrons dans un premier temps la NP-complétude du problème grâce à une réduction de MESP à MEIC. Nous proposons ensuite deux approximations du problème MEIC, une première en temps $O(n^{4.752}\log(n))$ utilisant le calcul du plus long cycle isométrique proposé par [Lokshtanov 2009] et une seconde en temps $O(n^3)$ mais nécessitant que le graphe possède un ratio longueur du cycle/domination important.

3.2 NP-complétude

Nous proposons une réduction du problème MESP au problème MEIC.

Soit la fonction f qui à un graphe G de sommets $V(G) = v_1, ...v_n$ associe une famille de graphes $f(G) = \{H_{(i,j)\in\{1...n\}^2}\}$ telle que pour tout $(i,j) \in \{1...n\}^2$, $H_{(i,j)}$ est le graphe G auquel on ajoute un chemin $P_{i,j}$ de longueur $2n$ entre les sommets v_i et v_j.

Montrons que pour tout graphe $H_{i,j}$ de $f(G)$, tout cycle isométrique d'excentricité minimale contient $P_{i,j}$ ainsi qu'un plus court chemin d'excentricité minimale dans G entre v_i et v_j.

Affirmation 1 : Tout cycle isométrique d'excentricité minimale dans $H_{i,j}$ contient $P_{i,j}$.

Soit un cycle quelconque de G. Il ne contient aucun sommet de $P_{i,j}$ autre que v_i et v_j, donc $P_{i,j}$ étant de longueur $2n$, son excentricité dans $H_{i,j}$ est d'au moins n. Considérons à présent un cycle quelconque de $H_{i,j}$ contenant $P_{i,j}$ ainsi qu'un plus court chemin entre v_i et v_j. Ce cycle est isométrique et d'excentricité au plus $n-1$.

Affirmation 2 : Tout cycle isométrique d'excentricité minimale dans $H_{i,j}$ contient $P_{i,j}$ et un plus court chemin d'excentricité minimale dans G entre v_i et v_j.

Nous avons déjà établi qu'un tel cycle contient nécessairement $P_{i,j}$ ainsi qu'un chemin entre v_i et v_j. Notons de plus que si le chemin du cycle entre v_i et v_j dans G n'est pas un plus court chemin alors le cycle n'est pas isométrique. En effet la distance entre v_i et v_j étant d'au plus n et $P_{i,j}$ étant de taille $2n$, ce dernier n'est donc pas un plus court chemin reliant v_i et v_j. Par définition d'un cycle isométrique il est alors nécessaire que le chemin du cycle entre v_i et v_j dans G soit un plus court chemin. Tout plus court chemin entre un sommet de $P_{i,j}$ et un sommet de G passe par v_i ou v_j. L'excentricité du cycle est donc égale à celle du plus court chemin entre v_i et v_j. Étant d'excentricité minimale, il contient donc le plus court chemin d'excentricité minimale entre v_i et v_j.

Finalement, pour tout graphe G, le plus court chemin d'excentricité minimale peut être calculé en cherchant le cycle isométrique d'excentricité minimale dans $f(G)$, nous avons la réduction polynomiale attendue.

Lemme 6. *Étant donné un graphe G, le chemin d'excentricité minimale de G est d'excentricité k si et seulement le cycle isométrique de plus faible excentricité dans $f(G)$ est d'excentricité k.*

Il suit de [Dragan 2017] :

Théorème 4. *Le problème MEIC est NP-complet.*

3.3 Résultat liminaire

On définit pour tout cycle C une orientation arbitraire et pour u, v des sommets du cycle, notons C_{uv} et C_{vu} les deux chemins dans C reliant u et v. Afin de simplifier les démonstrations, définissons l'opérateur ρ suivant :

Définition 7 (Opérateur ρ). *Soit G un graphe avec un cycle isométrique C k-dominant. Soit x un sommet de G, $\rho(x)$ est un sommet de C à distance au plus k de x.*

Exposons tout d'abord un premier lemme (analogue au lemme 1) qui nous servira dans la démonstration de nos deux approximations du problème MEIC.

Lemme 7. *Soit G un graphe contenant un cycle isométrique C k-dominant. Soient u et v deux sommets de G.*

Tout chemin entre u et v $2k$-domine $C_{\rho(u)\rho(v)}$ ou $C_{\rho(v)\rho(u)}$.

Démonstration. Les notations utilisées dans cette preuve sont illustrées par la figure 3.1.

Soit P un chemin entre u et v. Sans perte de généralité, supposons que P ne $2k$-domine pas un sommet b du chemin $C_{\rho(u)\rho(v)}$ et soit a un sommet quelconque de $C_{\rho(v)\rho(u)}$. Alors $\rho(u)$ (resp. $\rho(v)$) est un sommet de C_{ab} (resp. C_{ba}).

Ainsi u est à distance au plus k de C_{ab} et v à distance au plus k de C_{ba}. De plus, tout sommet de G est à distance au plus k d'un des deux chemins, il existe ainsi nécessairement c et d deux sommets successifs de P tels que $\rho(c)$ appartienne à C_{ab} et $\rho(d)$ à C_{ba}.

Comme $d(\rho(c),\rho(d)) \leq d(\rho(c),c)+d(c,d)+d(d,\rho(d)) \leq 2k+1$ et C est un cercle isométrique, $C_{\rho(c)\rho(d)}$ ou $C_{\rho(d)\rho(c)}$ est de taille au plus $2k+1$ et est donc $2k$ dominé par $\{c,d\}$.

De plus b et a appartiennent au même sous-chemin de C entre $\rho(c)$ et $\rho(d)$, ainsi a ou b est $2k$ dominé par $\{c,d\}$. Par hypothèse b ne pouvant être $2k$-dominé par P, il découle que a est $2k$-dominé par $\{c,d\}$, donc par P.

Cette affirmation étant valide pour tout a dans $C_{\rho(v)\rho(u)}$, le lemme est vérifié. □

3.4 Approximation avec le plus long cycle isométrique en temps $O(n^{4.752}\log(n))$

Nous montrons dans cette section qu'une approximation du problème MEIC peut être calculée en temps polynomial en utilisant le plus long cycle isométrique du graphe considéré.

Pour le calcul du plus long cycle isométrique, nous utiliserons l'algorithme en temps polynomial proposé par [Lokshtanov 2009].

Théorème 5 ([Lokshtanov 2009]). *Le problème du calcul du plus long cercle isométrique est polynomial et peut s'effectuer en temps $O(n^{4.752}\log(n))$.*

Nous nous servons de ce résultat pour approximer le problème MEIC en montrant que le plus long cycle isométrique d'un graphe en est une 6-approximation, et même, si le ratio longueur du cycle/domination est suffisamment grand, une 3-approximation.

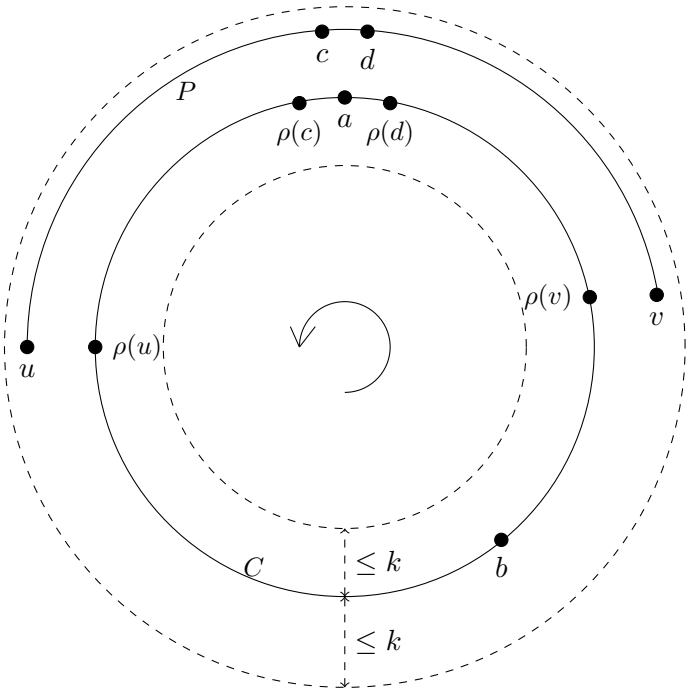

FIGURE 3.1 – Notations utilisées dans le lemme 7

Lemme 8. *Soit G un graphe contenant un cycle isométrique C k-dominant G. Soit D un plus long cycle isométrique de G.*

Alors, tout sommet de C est à distance au plus $5k$ de D. De plus, si D est de taille supérieure à $12k+2$, tout sommet de C est à distance au moins $2k$ de D.

Démonstration. Les notations utilisées dans cette preuve sont illustrées par la figure 3.2.

Soit $C = c_1, ..., c_p, c_1$ et supposons qu'il existe c_l tel que tout sommet de D est à distance supérieure à $2k$ de c_l.

Soit c_i (resp. c_j) des sommets à distance au plus k de D et tels que $C_{c_{i+1}c_l}$ (resp. $C_{c_lc_{j-1}}$) ne contiennent aucun sommet à distance k ou moins de D.

Soit a (resp. b) le sommet de D à distance au plus k de c_i (resp. c_j). Par le lemme 7 nous savons que $D_{a,b}$ et $D_{b,a}$ $2k$-dominent séparément soit $C_{c_ic_j}$, soit $C_{c_jc_i}$. Par hypothèse, c_l est à distance plus de $2k$ de D. $D_{a,b}$ et $D_{b,a}$ $2k$-dominent donc tous deux $C_{c_jc_i}$.

Soit m un sommet au milieu de $D_{a,b}$ et $\rho(m)$ un sommet de C à distance au plus k de m. Par les hypothèses précédentes nous savons que $\rho(m)$ appartient à $C_{c_jc_i}$.

Comme $C_{c_jc_i}$ est $2k$-dominé par $D_{b,a}$, il existe m_2 appartenant à $D_{b,a}$ à distance au plus $2k$ de $\rho(m)$ et

$$d(m, m_2) \leq d(m, \rho(m)) + d(\rho(m), m_2) \leq 3k$$

D étant un cycle isométrique, $|D_{m,m_2}| \leq 3k$ ou $|D_{m_2,m}| \leq 3k$. Supposons sans perte de généralité le premier cas, on sait que b appartient à D_{m,m_2}. Donc, $|D_{m,b}| \leq 3k$ et ainsi, $|D_{a,b}| \leq 6k+1$.

En choisissant m au milieu de $D_{b,a}$ (au lieu de $D_{a,b}$), nous montrons de la même manière que $D_{b,a}$ est de taille au plus $6k+1$.

Finalement D est de taille au plus $12k+2$. Ainsi, si D est de taille au moins $12k+3$, D $2k$-domine C.

Supposons maintenant que D est de longueur inférieure à $12k+2$. Soit deux sommets opposés u et v appartenant à D, c'est à dire à distance l'un de l'autre d'au moins $\lfloor \frac{p}{2} \rfloor$.

$$d(\rho(u),\rho(v)) \geq d(u,v) - d(\rho(u),u) - d(v,\rho(v)) \geq \left\lfloor \frac{p}{2} \right\rfloor - 2k$$

Il découle,

$$|C_{\rho(v)\rho(u)}| \leq |C| - d(\rho(u),\rho(v)) \leq \left\lceil \frac{p}{2} \right\rceil + 2k \leq 8k+1$$

De façon analogue, $|C_{\rho(u)\rho(v)}| \leq 8k+1$. Ainsi, pour tout c_l appartenant à C, $d(\rho(u),c_l) \leq 4k$ ou $d(\rho(v),c_l) \leq 4k$.

Comme u (resp. v) est à distance au plus k de $\rho(u)$ (resp. $\rho(v)$), $d(u,c_l) \leq d(u,c_i) + d(\rho(u),c_l) \leq 5k$ ou $d(v,c_l) \leq 5k$

□

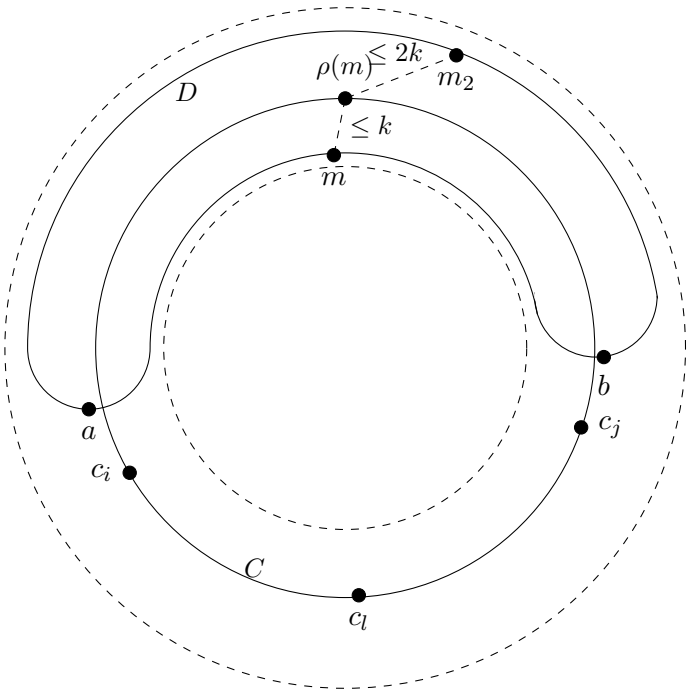

FIGURE 3.2 – Notations utilisées dans le lemme 8

Il découle de ce lemme et du théorème 5,

Théorème 6. *Soit un graphe contenant un cycle isométrique k-dominant de longueur ℓ, un cycle isométrique $6k$-dominant peut être calculé en temps $O(n^{4.752}\log(n))$. De plus si $\ell > 12k+2$, le cycle calculé est $3k$-dominant.*

3.5 Approximation en temps $O(n(m+kn)$

Nous proposons maintenant une approximation du problème MEIC de meilleure complexité mais requérant un graphe de plus grand diamètre.

Considérons un graphe comprenant un cycle isométrique d'excentricité k. L'algorithme *PremierCycle* calcule dans un premier temps un cycle U^0 d'excentricité inférieure à $3k$ mais non nécessairement isométrique. De plus, un sommet peut éventuellement apparaitre plusieurs fois dans le cycle. La taille du cycle est ensuite itérativement diminuée tout en conservant une excentricité inférieure à $3k$ à chaque itération. L'algorithme s'arrête quand le dernier cycle calculé est isométrique.

```
1 PremierCycle
  Entrée : Un graphe G
  Sortie : Un cycle U^0
2 Soit a un sommet quelconque de G
3 Calculer b un sommet à distance maximale de a
4 Calculer P un plus court chemin entre a et b
5 Soit m un sommet au milieu de P
6 Si a et b sont connectés dans G' = G \ B(m, 3/12|P|) alors
7 |   Soit P' un plus court chemin entre a et b dans G'
8 |   Renvoyer P ∪ P'
9 Renvoyer une erreur "Cycle trop court"
```

Algorithme 2: Pseudo-code de la fonction *PremierCycle*

Lemme 9. *Soit G un graphe contenant un cycle isométrique C k-dominant.*

Si C est de taille au moins $26k+6$ alors $PremierCycle(G)$ renvoie un cycle U^0 $2k$-dominant C. De plus U^0 est de taille au plus $|C|+4k$ et au moins $|C|-2k-2$.

Démonstration. Si $PremierCycle(G)$ retourne un cycle U^0, alors ce cycle est l'union de deux chemins P et P'.

Montrons dans un premier temps que P est plus long que $12k$. Soit $\overline{\rho(a)}$ un sommet de C à distance maximale de $\rho(a)$.

$$ecc(a) \geq d(a,\overline{\rho(a)}) \geq d(\rho(a),\overline{\rho(a)}) - k \geq \left\lfloor \frac{C}{2} \right\rfloor - k \geq 12k+3$$

Comme b est un sommet de G à distance maximale de a, nous avons le résultat.

Soit m un sommet au milieu de P. $\rho(m)$ appartient à $C_{\rho(a)\rho(b)}$ ou $C_{\rho(b)\rho(a)}$. Supposons sans perte de généralité le premier cas et montrons alors que P $2k$-domine $C_{\rho(a)\rho(b)}$. Pour ceci, montrons d'abord que P_{am} $2k$ domine $C_{\rho(a)\rho(m)}$.

Supposons que ce ne soit pas le cas. Alors, par le lemme 7, P_{am} $2k$-domine $C_{\rho(m)\rho(a)}$. $\rho(b)$ appartenant à $C_{\rho(m)\rho(a)}$, il existe un sommet x dans P_{am} à distance au plus $2k$ de $\rho(b)$. Par conséquence,

$$|P_{am}| = d(a,x) + d(x,m)$$
$$|P_{am}| \geq d(a,b) - 3k + d(b,m) - 3k$$
$$\frac{|P|}{2} \geq \frac{3|P|}{2} - 6k$$
$$6k \geq |P|$$

Ceci contredit le fait que P soit de taille supérieure à $12k$ donc P_{am} $2k$-domine $C_{\rho(a)\rho(m)}$. De la même manière, on montre que P_{mb} $2k$ domine $C_{\rho(m)\rho(b)}$ et donc que P $2k$-domine $C_{\rho(a)\rho(b)}$.

Montrons maintenant que $B(m, \frac{3}{12}|P|)$ ne déconnecte pas a et b. Pour ceci nous montrons dans un premier temps que tout sommet de $C_{\rho(b)\rho(a)}$ est à distance plus de $\frac{3}{12}|P|$ de m. Donc,

$$d(\rho(m),\rho(a)) \geq d(a,m) - d(a,\rho(a)) - d(m,\rho(m)) \geq \left\lfloor \frac{|P|}{2} \right\rfloor - 2k$$
$$d(\rho(m),\rho(b)) \geq d(b,m) - d(b,\rho(b)) - d(m,\rho(m)) \geq \left\lfloor \frac{|P|}{2} \right\rfloor - 2k$$

Soit y un sommet dans $C_{\rho(b)\rho(a)}$,

$$d(m,y) \geq d(\rho(m),y) - d(m,\rho(m)) \geq min(d(\rho(m),\rho(a)), d(\rho(m),\rho(b))) - k \geq \left\lfloor \frac{|P|}{2} \right\rfloor - 3k$$
$$\geq \left\lfloor \frac{3|P|}{12} \right\rfloor + \left\lfloor \frac{3|P|}{12} \right\rfloor - 3k > \left\lfloor \frac{3|P|}{12} \right\rfloor$$

$C_{\rho(b)\rho(a)}$ est donc à distance au moins $\frac{3}{12}|P|$ de m. De plus,

$$d(a,m) = \left\lfloor \frac{|P|}{2} \right\rfloor \geq \left\lfloor \frac{3}{12}|P| \right\rfloor + 3k$$
$$d(b,m) = \left\lfloor \frac{|P|}{2} \right\rfloor \geq \left\lfloor \frac{3}{12}|P| \right\rfloor + 3k$$

Tout chemin de taille au plus k partant de a ou b est donc à distance au plus $\frac{3}{12}|P| + 2k$ de m. Il suit qu'il existe un chemin entre a (resp. b) et $\rho(a)$ (resp. $\rho(b)$) à distance plus de $\frac{3}{12}|P|$ de m. Les sommets a et b sont donc connectés dans G'.

Par le lemme 7 on sait que P' $2k$-domine $C_{\rho(a)\rho(b)}$ ou $C_{\rho(b)\rho(a)}$. Comme $\frac{3}{12}|P|$ est supérieur à $3k$ on sait qu'aucun sommet à distance au plus $2k$ de $\rho(m)$ n'est dans G'. Ainsi P' ne $2k$-domine pas $C_{\rho(a)\rho(b)}$, il $2k$-domine donc $C_{\rho(b)\rho(a)}$. Il découle que $PremierCycle$ renvoie bien un cycle $2k$-dominant C.

Montrons maintenant que $P \cup P'$ est de longueur au plus $|C| + 4k$:

$$|P| \leq |C_{\rho(a)\rho(b)}| + 2k$$
$$|P'| \leq |C_{\rho(b)\rho(a)}| + 2k$$
$$|P| + |P'| \leq |C| + 4k$$

Finalement, montrons que $P \cup P'$ est de longueur au moins $|C| - 2k - 2$. Considérons à nouveau un sommet $\overline{\rho(a)}$ de C à distance $\frac{|C|}{2}$ de $\rho(a)$.

$$d(a,b) \geq d(a, \overline{\rho(a)}) \geq \left\lfloor \frac{|C|}{2} \right\rfloor - k$$

d'où,

$$|P| + |P'| \geq 2d(a,b) \geq |C| - 2k - 2$$

□

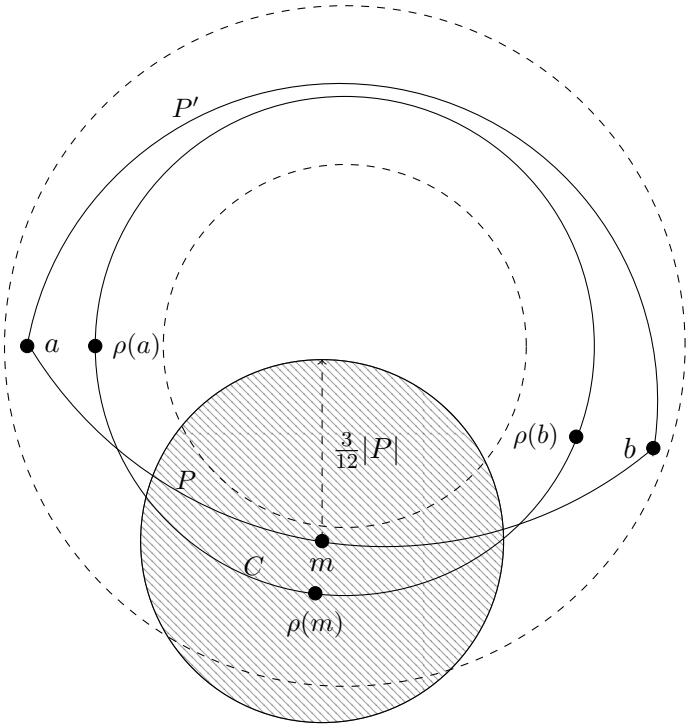

FIGURE 3.3 – Notations utilisées dans le lemme 9

Nous venons par la procédure *PremierCycle* de créer un cycle U^0 non nécessairement isométrique 3-dominant le graphe. A partir de U^0, nous allons réduire itérativement la taille du cycle tout en conservant la propriété de $3k$-domination.

Lemme 10. *Soit G un graphe contenant un cycle isométrique C k-dominant G. Soit a et b deux sommets, éventuellement confondus, reliés par un chemin P qui $2k$-domine $C_{\rho(a)\rho(b)}$.*

Alors, $|P| \geq |C_{\rho(a)\rho(b)}| - 6k - 2$.cEn particulier, si U est un cycle $2k$-dominant C, U est de taille au moins $|C| - 6k - 2$.

CycleSuivant
Entrée : Un graphe G et un cycle U^i
Sortie : Un cycle U^{i+1}
Pour $a \in U^i$ **faire**
  Si $S = \{(a,b) \mid d_{U^i}(a,b) = \left\lfloor \frac{|U^i|}{2} \right\rfloor$ *et* $d_G(a,b) < d_{U^i}(a,b)\} \neq \emptyset$ **alors**
    Calculer Q un plus court chemin entre a et b
    Calculer l'excentricité de $Q \cup U^i_{a,b}$ et $Q \cup U^i_{b,a}$
    Renvoyer l'ensemble d'excentricité la plus faible
Renvoyer U^i

Algorithme 3: Pseudo-code de la fonction *CycleSuivant*

Démonstration. Considérons deux sommets a et b quelconques et un chemin P les reliant qui $2k$-domine $C_{\rho(a)\rho(b)}$.

Soit m le milieu de $C_{\rho(a)\rho(b)}$ et z un sommet de P qui $2k$-domine m. $C_{\rho(a)m}$ étant de longueur au plus $\left\lceil \frac{|C|}{2} \right\rceil$, $|C_{\rho(a)m}| \leq d(\rho(a), m) + 1$ (la constante 1 n'étant en fait nécessaire que si $\rho(a) = \rho(b)$ et que C est de longueur impaire). De même, $|C_{m\rho(b)}| \leq d(m, \rho(b)) + 1$.

Ainsi,

$$\begin{aligned} |C_{\rho(a)\rho(b)}| &\leq d(\rho(a), m) + d(m, \rho(b)) + 2 \\ &\leq (k + |P_{a,z}| + 2k) + (k + |P_{z,b}| + 2k) + 2 \\ &\leq |P_{a,b}| + 6k + 2 \end{aligned}$$

L'affirmation concernant le cycle en découle immédiatement en choisissant $a = b$. □

Lemme 11. *Soit G un graphe contenant un cycle isométrique C k-dominant G. Pour tout chemin P, il existe un arc $I(P)$ de C tel que :*

- *tous les sommets de C à distance au plus k de P appartiennent à $I(P)$ et les extrémités de $I(P)$ sont de tels sommets ;*
- *P $2k$-domine $I(P)$*

Démonstration. Soit $\rho(u)$ et $\rho(v)$ deux sommets de C tels que u et v appartiennent à P et tels que P $2k$-domine $C_{\rho(u)\rho(v)}$ (une telle paire existe en prenant $\rho(u) = \rho(v)$). Supposons l'existence d'un sommet $\rho(w)$ à distance au plus k de $w \in P$ et n'appartenant pas à $C_{\rho(u)\rho(v)}$. D'après le lemme 7, P $2k$-domine soit $C_{\rho(u)\rho(w)}$ soit $C_{\rho(w)\rho(u)}$. Dans le second cas P $2k$-domine $C_{\rho(w)\rho(u)}$ et $C_{\rho(u)\rho(v)}$ donc $C_{\rho(w)\rho(v)}$. Comme $\rho(v)$ est dans $C_{\rho(u)\rho(w)}$ et $\rho(u)$ dans $C_{\rho(w)\rho(v)}$, les tailles de $C_{\rho(u)\rho(w)}$ et $C_{\rho(w)\rho(v)}$ sont strictement plus grandes que celle de $C_{\rho(u)\rho(v)}$. La longueur de C étant finie, il existe une paire de sommets vérifiant la propriété et tel que l'arc considéré est maximal. □

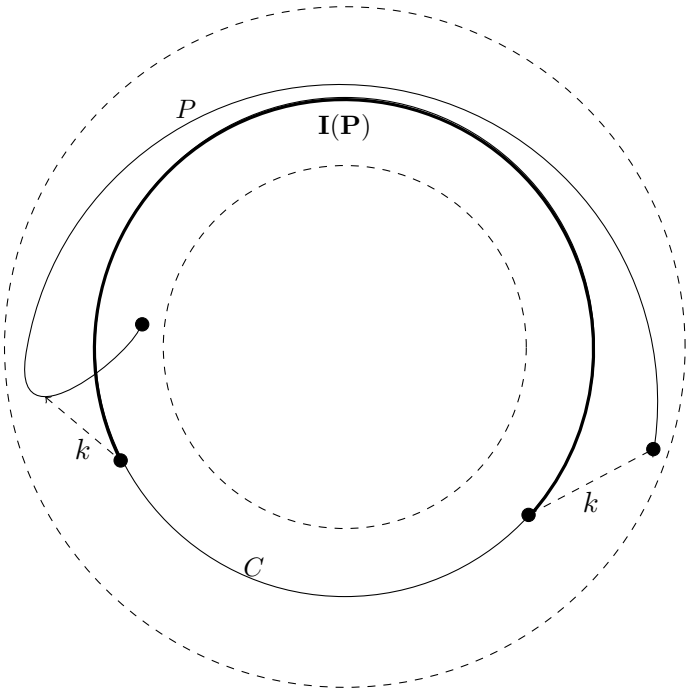

FIGURE 3.4 – Illustration du lemme 11

Lemme 12. *Supposons que* $|C| \geq 32k + 7$. *Soit* U *un cycle de longueur au plus* $|C| + 4k$ *qui* $3k$*-domine* G *et soient* a *et* b *deux sommets de* U.

Alors, $C_{\rho(a)\rho(b)}$ *et* $C_{\rho(b)\rho(a)}$ *sont chacun inclus soit dans* $I(U_{ab})$, *soit dans* $I(U_{ba})$.

Démonstration. Supposons que ce n'est pas le cas. Par symétrie, on peut supposer que $C_{\rho(a)\rho(b)}$ n'est inclus ni dans $I(U_{ab})$, ni dans $I(U_{ba})$.

Soit $\rho(w_1)$ l'extrémité de $I(U_{ab})$ telle que $C_{\rho(a)\rho(w_1)} \subset C_{\rho(a)\rho(b)}$ et $\rho(w_2)$ l'extrémité de $I(U_{ba})$ telle que $C_{\rho(a)\rho(w_2)} \subset C_{\rho(a)\rho(b)}$. Par symétrie, supposons que $C_{\rho(a)\rho(w_2)} \subset C_{\rho(a)\rho(w_1)}$ et posons $w = \rho(w_1)$. Il existe alors $w \in U_{ab}$ tel que $d(w, \rho(w)) = k$.

Cas 1 : $d(\rho(w), \rho(b)) \leq 4k + 1$

Alors $|C_{\rho(b)\rho(w)}| \geq |C| - 4k - 2$. Or, par définition de $\rho(w)$, $C_{\rho(b)\rho(w)}) \subset I(U_{wb})$ donc est $2k$-couvert par U_{wb}. Le lemme 10 implique alors que $|U_{wb}| \geq |C_{\rho(b)\rho(w)}| - 6k - 2 \geq |C| - 10k - 4$.

La définition de $\rho(w)$ implique aussi que $C_{\rho(a)\rho(w)} \subset I(U_{aw}$. Or, $C_{\rho(b)\rho(a)} \subset I(U_{ba})$ par le lemme 8 puisque, par hypothèse, $I(U_{ba})$ ne contient pas $C_{\rho(a)\rho(b)}$. Par conséquent, $C_{\rho(b)\rho(w)}) \subset I(U_{bw})$ et par le lemme 10, $|U_{bw}| \geq |C_{\rho(b)\rho(w)}| - 6k - 2 \geq |C| - 10k - 4$.

Finalement, $|U| = |U_{bw}| + |U_{wb}| \geq 2|C| - 20k - 8$. Or, $|U| \leq |C| + 4k$, ce qui implique $|C| \leq 24k + 8$, c'est-à-dire une contradiction.

Cas 2 : $d(\rho(w), \rho(b)) \geq 4k + 1$

Soit z le sommet de $C_{\rho(a)\rho(b)}$ appartenant à $C_{z\rho(b)}$ et qui se trouve à distance $4k+1$ de $\rho(w)$. U $3k$-couvrant G par hypothèse, z est $3k$-couvert par un sommet y de U. Le fait que $d(\rho(w), z) > 4k$ implique alors que $\rho(y)$ n'appartient pas à $C_{\rho(a)\rho(w)}$. Il reste à distinguer deux sous-cas, suivant que $y \in U_{ab}$ ou $y \in U_{ba}$.

Soit $y \in U_{ab}$. Décomposons alors U en $U = U_{wy} \cup (U_{yb} \cup U_{ba} \cup U_{aw})$.

Par définition de $\rho(w)$, $I(U_{wy})$ contient $C_{\rho(y)\rho(w)}$ et donc $|U_{wy}| \geq |C_{\rho(y)\rho(w)}| - 6k - 2 \geq |C| - d(\rho(y), z) - d(z, \rho(w)) - 6k - 2 \geq |C| - 14k - 3$.

De plus, $I(U_{yb})$ contient $C_{\rho(y)\rho(b)}$ par définition de $\rho(w_2)$, $I(U_{ba})$ contient $C_{\rho(b)\rho(a)}$ par hypothèse et $I(U_{aw})$ contient $C_{\rho(a)\rho(w)}$ par définition de $\rho(w)$, donc $I(U_{yb} \cup U_{ba} \cup U_{aw})$ contient $C_{\rho(y)\rho(w)}$. Sa longueur est donc également supérieure ou égale à $|C| - 14k - 3$.

Finalement, $2(|C| - 14k - 3) \leq |C| + 4k$, soit $|C| \leq 32k + 6$, ce qui est une contradiction.

Supposons qu'au contraire $y \in U_{ba}$. Décomposons alors U en $U = (U_{wb} \cup U_{by}) \cup (U_{ya} \cup U_{aw})$.

$I(U_{wb})$ contient $C_{\rho(b)\rho(w)}$ par définition de $\rho(w)$ et $I(U_{by})$ contient $C_{\rho(y)\rho(b)}$ par définition de $\rho(w_1)$. Par conséquent, $I(U_{wb} \cup U_{by})$ contient $C_{\rho(y)\rho(w)}$ et $|U_{wb} \cup U_{by}| \geq |C| - 14k - 3$.

De plus, $I(U_{ya})$ contient $C_{\rho(y)\rho(a)}$ car il ne contient pas $C_{\rho(a)\rho(y)}$ par hypothèse et $I(U_{aw})$ contient $C_{\rho(a)\rho(w)}$ par définition de $\rho(w)$. D'où $I(U_{ya} \cup U_{aw})$ contient $C_{\rho(y)\rho(w)}$ et $|U_{ya} \cup U_{aw}| \geq |C| - 14k - 3$.

A nouveau, cela implique que $|C| \leq 32k + 6$. □

Proposition 1. *Soit G un graphe contenant un cycle isométrique C k-dominant G et de taille au moins $32k + 7$.*

Soit U^i un cycle tel que :

— *U^i $2k$-domine C.*

— *$|C| - 6k - 2 \leq |U^i| \leq |C| + 4k$*

Alors $U^{i+1} = CycleSuivant(U^i)$ vérifie les mêmes propriétés et $|U^{i+1}| < |U^i|$.

Démonstration. Les inégalités sur les longueurs sont évidentes au vu du Lemme 10 et du fait que U^{i+1} est obtenu en remplaçant un arc de U^i par un chemin plus court. Il suffit donc de démontrer que U^{i+1} $2k$-domine C.

Considérons le chemin Q reliant deux sommets a et b de U^i tel que selectionné par l'algorithme. Le lemme 7 permet de supposer sans perte de généralité que $I(Q)$ contient $C_{\rho(a)\rho(b)}$ et le $2k$-domine donc. De plus, le Lemme 12 implique que $C_{\rho(b)\rho(a)}$ est inclus dans $I(U^i_{ab})$ ou $I(U^i_{ba})$. Par symétrie, supposons qu'il est inclus dans $I(U^i_{ab})$.

Alors, $I(Q \cup U^i_{ab}) = C$, ce qui implique que $Q \cup U^i_{ab}$ $2k$-domine C et est donc d'excentricité au plus $3k$. La propriété est donc vérifiée si $U^{i+1} = Q \cup U^i_{ab}$.

De plus, si $U^{i+1} = Q \cup U^i_{ba}$, $Q \cup U^i_{ba}$ $3k$-domine G également puisqu'il est d'excentricité plus faible. Le Lemme 12 implique alors que U^i_{ba} ou Q $2k$-domine $C_{\rho(b)\rho(a)}$, et ainsi $Q \cup U^i_{ba}$ $2k$-domine C.

□

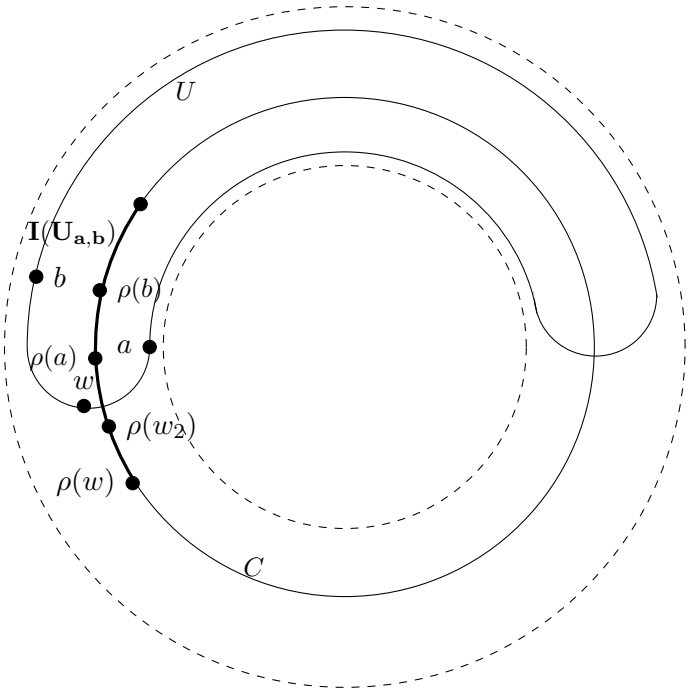

FIGURE 3.5 – Notations utilisées dans le **cas 1** du Lemme 12

```
1 ApproximationCycle
  Entrée : Un graphe G
  Sortie : Un cycle
2 U=PremierCycle(G)
3 V=CycleSuivant(G,U)
4 Tant que V ≠ U faire
5 |   U=V
6 |   V=CycleSuivant(G,V)
7 Renvoyer V
```

Algorithme 4: Pseudo-code de la fonction *ApproximationCycle*

Étude de la complexité :

L'algorithme *PremierCycle* requiert trois calculs d'arbre BFS, sa complexité est $O(m)$.

Le goulot d'étranglement de complexité est de tester la condition **Si** sur n^2 paires de sommets à la ligne 3 de *CycleSuivant*. Les distances dans G peuvent être pré-calculées en temps $O(nm)$ en utilisant n BFS, le calcul de la distance d_{U^i} dans le cycle peut être effectuée en temps constant si les sommets sont numérotés. Le test de la condition prend donc un temps $O(n^2)$. Quand une paire de sommets (a, b) est trouvée, le bloc jusqu'à la ligne 6 demande le calcul de trois BFS, soit une complexité en temps $O(m)$, ceci est effectué une fois par appel de *CycleSuivant*. A supposer la matrice des distances pré-calculéé, l'algorithme 6 (*CycleSuivant*) prend a donc

une complexité de $O(n^2)$.

Nous savons par le lemme 1 que la taille des cycles de la suite U^i décroit strictement et est d'au moins $|C| - 6k - 2$. Comme $|U^0| \leq |C| + 4k$, il suit que la suite converge après au plus une exécution de *PremierCycle* et au plus $10k + 2$ exécutions de *CycleSuivant*. Sans oublier le pré-calcul des distances en temps $O(nm)$ nous avons :

Théorème 7. *Soit un graphe contenant un cycle isométrique k-dominant de taille $\ell \geq 32k + 4$, un cycle isométrique $3k$-dominant peut être calculé en temps $O(n(m + kn))$.*

Nous avons analysé le problème du cercle isométrique de plus faible excentricité pour le cas de graphes généraux et proposé une 5-approximation en temps $O(n^{4.752} \log(n))$ et une 3-approximation en temps $O(n(m+kn))$ mais nécessitant le cycle isométrique d'excentricité minimale d'être suffisamment grand relativement à k.

Les problèmes MESP et MEIC maintenant étudiés, nous allons dans le chapitre suivant nous en servir comme briques de base pour définir la décomposition hub-laminaire. Cette dernière permettant une modélisation des graphes de read, sujet initial de la présente thèse.

Chapitre 4

Décomposition hub-laminaire

Sommaire

4.1 Introduction

Nous développons dans ce chapitre une modélisation des graphes de reads présentés en introduction. Pour rappel, chaque sommet d'un graphe de read correspond à une séquence d'ADN d'environ 300 caractères. Deux sommets sont reliés si les séquences ont un taux de similarité suffisant. Ainsi la reconstruction d'un brin d'ADN — proche de la construction d'un graphe d'intervalle — produit un graphe laminaire comme développé au chapitre 2. Dans le cas où une séquence génétique est présente dans plusieurs contextes génomiques, plusieurs laminaires peuvent se croiser au niveau de cette séquence. Nous proposons une décomposition de ces graphes en leurs différents brins laminaires et zones d'intersections.

Cette modélisation peut être vu comme une extension des problèmes MESP et MEIC en une décomposition de graphes en chemins de faibles excentricités. Plus précisément, nous introduisons la notion de décomposition hub-laminaire constituée d'un ensemble de plus courts chemins, nommés *chemins laminaires*, se croisant uniquement en leurs extrémités, nommées *centres de hubs*. L'ensemble des sommets k-dominés par un chemin laminaire est appelé *laminaire*. La boule de rayon r centrée sur un centre de hub est appelée *hub*. L'union des laminaires et hubs contient

l'intégralité des sommets du graphe. De plus, notre définition requiert que toute arête reliant deux laminaires distincts doit appartenir à un hub, ainsi les laminaires ne se croisent qu'au niveau des hubs. Le *degré* d'un hub est le nombre de laminaires se rejoignant dans ce hub.

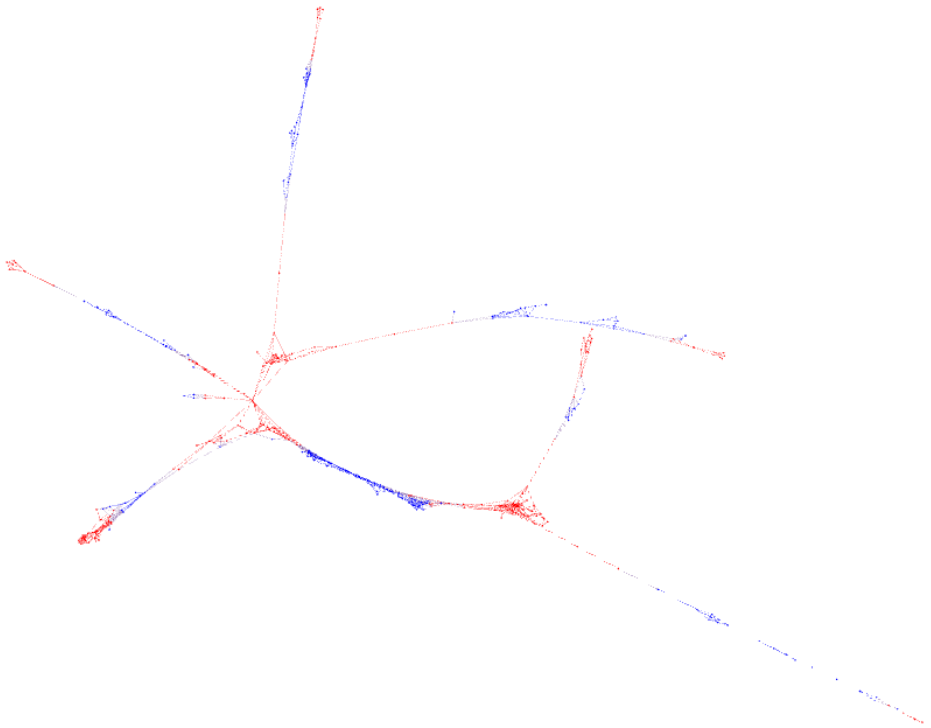

FIGURE 4.1 – Exemple d'un graphe de reads décomposé en hubs et laminaires.

On peut noter que la recherche d'une décomposition contenant deux hubs de rayon k et un seul laminaire de largeur k les reliant correspond à la recherche d'un plus court chemin d'excentricité k. Le problème MESP peut donc être vu comme un cas particulier du problème hub-laminaire.

Nous montrons dans ce chapitre qu'une décomposition hub-laminaire est détectable en temps polynomial si les hubs sont suffisamment éloignés les uns des autres. En dernière partie nous confronterons notre algorithme de décomposition hub-laminaire à des graphes générés aléatoirement et aux graphes issus de nos données biologiques.

Hormis la dernière partie simulation, les résultats de ce chapitre ont été présentés à ISAAC 2017 [Birmelé 2017]. De plus ces résultats seront prochainement soumis à un journal, l'écriture est en cours.

Dans ce chapitre les preuves de certains lemmes sont en section 4.3.6 pour améliorer la lisibilité de l'ensemble.

4.2 Définitions

4.2.1 Décomposition hub-laminaire

Définition 8 (Décomposition hub-laminaire)**.** *Soit un graphe G, deux entiers r et k avec $k \leq r$, $H = \{h_1, \ldots, h_q\}$ un ensemble de sommets de G appelés* centres de hub*, et $\mathcal{P} = \{P_1, \ldots, P_p\}$ un ensemble de chemins de G appelés* chemins laminaires*.*

Une boule $B(h, r)$ avec $h \in H$ est appelée un hub*, et un ensemble $B(P, k)$ avec $P \in \mathcal{P}$ est appelée un* laminaire*. $(H, \mathcal{P})$ est une* (r, k)-décomposition hub-laminaire *de G si les conditions suivantes sont satisfaites :*

1. tout laminaire lie deux centres de hubs : *les extrémités h, h' de tout $P \in \mathcal{P}$ appartiennent à H et pour tout centre de hub $h'' \in H \setminus \{h, h'\}$,*

$$B(P, k) \cap B(h'', r + 1) = \emptyset$$

2. Les laminaires et les hubs couvrent G : $V(G) = \bigcup_{h \in H} B(h, r) \cup \bigcup_{P \in \mathcal{P}} B(P, k)$

3. tout chemin laminaire est localement un plus court chemin : *tout chemin $P \in \mathcal{P}$ d'extrémités h et h' est un plus court chemin dans $G[B(P, k) \cup B(h, r) \cup B(h', r)]$*

4. les laminaires se rejoignent uniquement aux hubs : *pour tout $i \neq j$ et $uv \in E(G)$ tels que $u \in B(P_i, k)$ et $v \in B(P_j, k)$, il y a un centre de hub $h \in H$ tel que P_i et P_j ont tous deux h comme extrémité et tel que $u, v \in B(h, r)$.*

La longueur laminaire minimale *d'une décomposition $(H, \mathcal{P})$, notée ℓ, est la longueur minimale des chemins de $\mathcal{P}$. La* taille laminaire*, notée λ, correspond au nombre de chemins dans $\mathcal{P}$.*

Une décomposition hub-laminaire $(H, \mathcal{P})$ avec $\ell \geq 2r+1$ forme une partition des arêtes de G : toute arête (u, v) est soit dans exactement un hub, c'est à dire qu'il existe un unique $h \in H$ tel que $u, v \in B(h, r)$; ou alors dans un unique laminaire, c'est à dire qu'il existe un unique $P \in \mathcal{P}$ tel que $u, v \in B(P, k)$.

La figure 4.2 illustre cette définition ainsi que celle de graphe quotient que nous allons définir au paragraphe suivant. Une décomposition hub-laminaire est intuitivement un ensemble de λ k-voisinages de plus courts chemins disjoints en leurs parties centrales. La définition peut sembler intriquée mais nous pensons qu'elle rend compte d'une façon minimale de cette intuition. L'axiome 4 traduit la nécessité d'avoir les chemins disjoints en leurs parties centrales. Les axiomes 1 et 2 indiquent que le graphe est décomposé en laminaires qui sont le k-voisinage de certains chemins et en boules centrées aux extrémités de ces chemins. L'axiome 3 demande à chaque chemin laminaire d'être un plus court chemin dans le graphe induit par le laminaire. Ceci permet à plusieurs laminaires de longueurs différentes de relier le même hub.

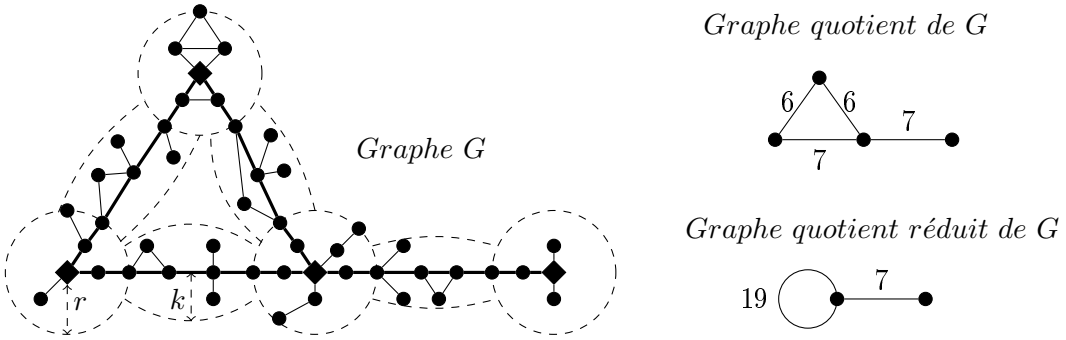

FIGURE 4.2 – Illustration d'une décomposition hub-laminaire avec $r = 2, k = 1$. Tout sommet est à distance au plus r d'un centre de hub (sommets carrés) ou à distance au plus k d'un chemin laminaire (chemins en gras reliant les centres de hubs)

4.2.2 Graphe quotient et équivalence entre les décompositions

Une décomposition hub-laminaire produit assez naturellement un *graphe quotient.*

Définition 9 (graphe quotient et quotient réduit)**.** *Soit un graphe G et une décomposition (r, k) hub-laminaire $(H, \mathcal{P})$ de G. Le* quotient *de cette décomposition est un multigraphe avec pour sommets l'ensemble H et ayant pour tout $P \in \mathcal{P}$ d'extrémités h et h', une arête hh' de label la longueur de P.*

Le degré d'un hub *est le degré du sommet correspondant dans le graphe quotient, ou de manière équivalente, le nombre de chemins laminaires ayant le centre de ce hub pour extrémité.*

Le graphe quotient réduit *d'une décomposition $(H, \mathcal{P})$ est le multigraphe obtenu à partir du graphe quotient en supprimant itérativement chaque sommet de degré 2 : pour tout sommet u du quotient incident à deux arêtes uv et uw de labels a et b, u et les deux arêtes sont supprimées, une nouvelle arête vw est ajoutée, de label $a + b$. (C'est une boucle quand $v = w$.)*

Définition 10 (équivalence entre décompositions)**.** *Deux décompositions hub-laminaires d'un graphe G - possiblement avec des paramètres r et k différents - sont* D-équivalentes *si :*

- *Elles ont le même graphe quotient réduit.*
- *Il existe un isomorphisme ϕ entre les centres de hubs de degré différent de* 2 *tel que pour tout tel centre h, $d_G(h, \phi(h)) \leq D$.*

4.3 Approximation polynomiale

4.3.1 Contexte d'étude

Le graphe quotient réduit d'un graphe respecte trivialement une des propriétés suivantes : c'est soit un chemin, c'est soit un cycle, soit il a un sommet de degré 3. Nous traitons séparément les trois cas.

Dans le premier cas, le graphe a un plus court chemin d'excentricité au plus $\max\{3k, 2r\}$ qui peut être calculé en temps polynomial suivant l'algorithme MESP développé au chapitre 2. La borne $\max\{3k, 2r\}$ est une conséquence du lemme 1 montré au chapitre 2 et rappelé en section 4.3.2. Dans le second cas, le graphe a un cycle isométrique d'excentricité au plus $\max\{3k, 2r\}$ qui peut être calculé en temps polynomial par l'algorithme MEIC développé au chapitre 3. Nous nous intéressons donc dans la suite de ce chapitre au troisième cas, c.à.d. aux décompositions possédant un sommet de degré 3 dans leur graphe quotient réduit.

Notons que l'algorithme que nous allons présenter produit tout de même des résultats pertinents dans les deux premiers cas. Dans le cas du MESP, l'exécution de l'algorithme est équivalente à la 5-approximation développée au chapitre 2. Dans le cas du MEIC, il ne renvoie pas un cycle isométrique mais bien deux plus courts chemins de mêmes extrémités, K-dominant le graphe. Les figures 4.3 et 4.4 présentent des retours de l'algorithme pour ces deux cas.

La reconnaissance d'une décomposition dans le troisième cas n'est pas un problème bien défini. En effet plusieurs décompositions distinctes peuvent être valides suivant des valeurs différentes de r et k. Cependant, quand les laminaires sont suffisamment longs, toutes les décompositions (r, k)-hub-laminaire sont $O(k)$-équivalentes. Ceci peut être vu comme une conséquence du résultat suivant, théorème principal de ce chapitre dont la preuve sera développée dans les parties suivantes.

Théorème 8. *Soit un graphe G possédant une décomposition (r, k)-hub-laminaire $(H, \mathcal{P})$ contenant λ laminaires et de longueur laminaire minimale $\ell \geq 10r + 60k + 4$ et des entiers K, R tels que $K \geq 3k$, $R \geq 4K + 3r$ et $2R + 8K < \ell - 4r - 4k - 4$.*

Il est possible de calculer en temps $O(\lambda m)$ une décomposition (K, R)-hub-laminaire qui est $(K + 2r + k)$-équivalente à $(H, \mathcal{P})$.

4.3.2 Présentation de l'algorithme et résultats préliminaires

Nous présentons dans un premier temps l'algorithme et les résultats généraux le justifiant. Afin de préserver une certaine clarté, les preuves techniques et les résultats intermédiaires seront présentés dans la section 4.3.6.

Dans tout le reste de cette partie, les hypothèses du théorème 8 seront considérées comme respectées, c'est à dire $K \geq 3k$, $R \geq 4K + 3r$ et $2R + 8K < \ell - 4r - 4k - 4$.

L'idée sous-jacente à l'algorithme est d'utiliser un BFS pour calculer des plus courts chemins et leurs K-voisinages, afin d'appliquer le lemme 1 central présenté au chapitre 2 et que nous rappelons ci dessous. Ce lemme implique que tout plus

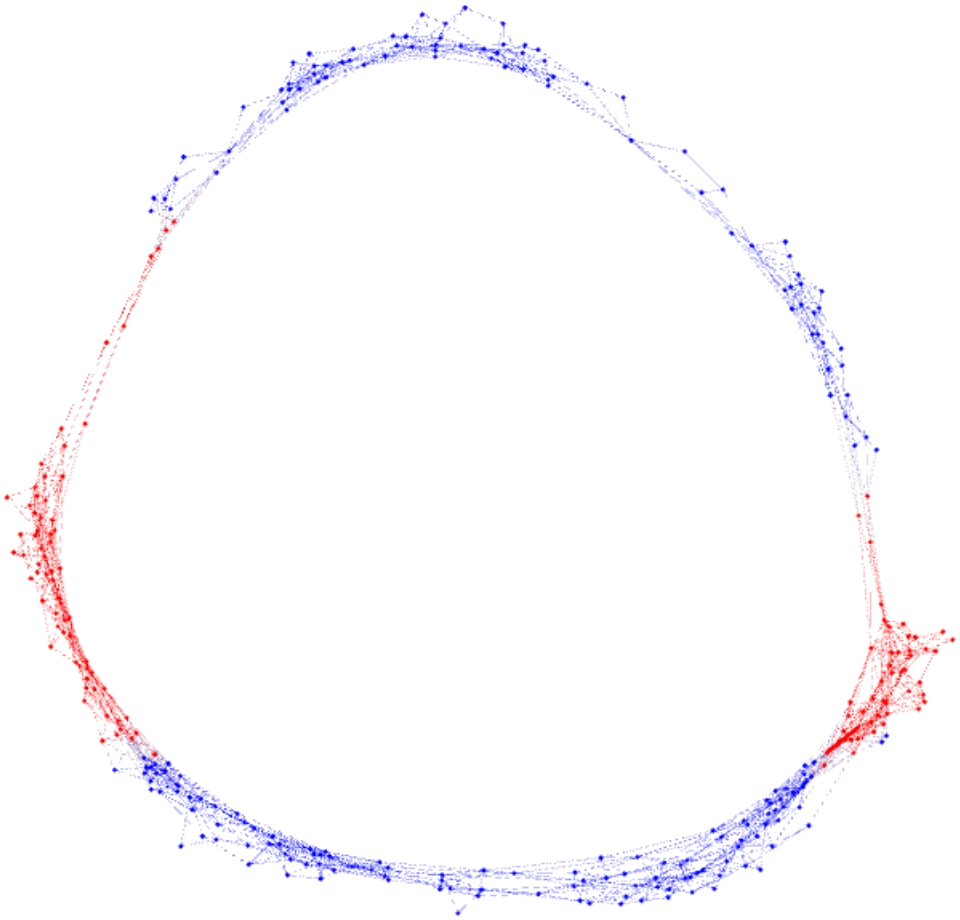

FIGURE 4.3 – Retour de l'algorithme sur un graphe possédant un MESP d'excentricité 1. Les sommets en rouge correspondent aux hubs, en bleu aux laminaires.

court chemin traversant un laminaire $3k$-domine sa partie centrale, c'est à dire tous les sommets suffisamment éloignés des deux centres de hubs situés aux extrémités du laminaire.

Lemme 1. *Soit G un graphe, Q un chemin quelconque et $P = v_0, v_1, ...v_l$ un plus court chemin d'excentricité k. Soient i et j $(i \leq j)$, tels que v_i et v_j sont à distance au plus k de Q.*

Alors, tout sommet de $P_{v_{i-k},v_{j+k}}$ est à distance au plus $2k$ de Q. Par conséquence, tout sommet de G à distance au plus k de $P_{v_{i-k},v_{j+k}}$ est à distance au plus $3k$ de Q.

Grâce à ce lemme, choisir $K \geq 3k$ garantira la domination de tout laminaire traversé par un plus court chemin. Cependant, les extrémités de ces plus courts chemins doivent être choisies en premier pour approximer H.

Définition 11 (critères d'approximation des hubs)**.** *Un sommet a* domine *un centre de hub $h \in H$ si $d(a, h) \leq K + 2r + k$.*

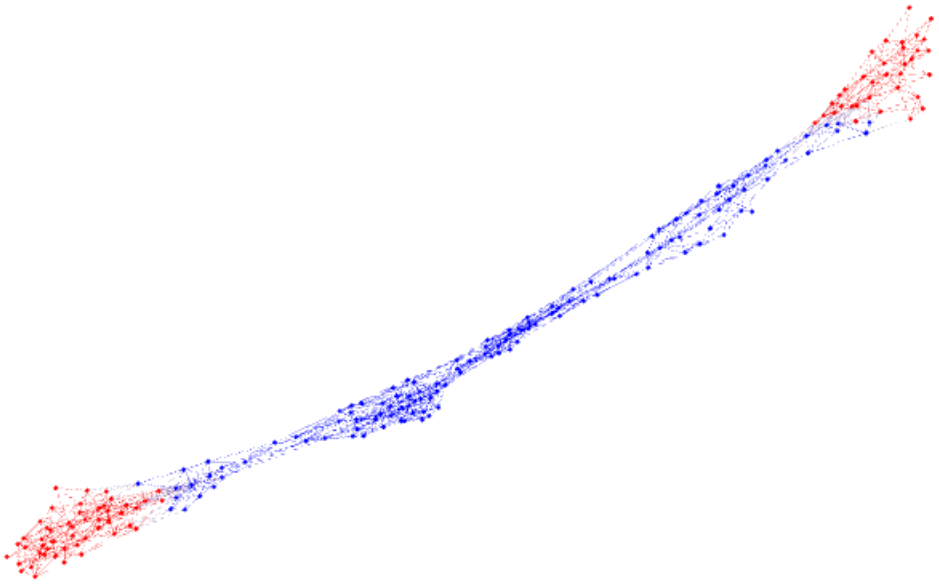

FIGURE 4.4 – Retour de l'algorithme sur un graphe possédant deux hubs de degré 2 et des laminaires de largeur 1. Les sommets en rouge correspondent aux hubs, en bleu aux laminaires.

Un ensemble de sommets A est H-proche *si tout sommet de A domine un sommet de H et si aucun sommet de H n'est dominé par deux sommets de A.*
Un ensemble de sommets A est H-dominant *si il est H-proche et si tout sommet de H correspondant à un hub de degré différent de 2 est dominé par un sommet de A.*

La première partie de notre algorithme, appelée *TrouverHubs*, qui sera développée en section 4.3.4, détermine un ensemble H-dominant A de sommets qui correspondront aux centres de hubs dans la décomposition calculée.

Notons que $\ell > 2(K+2r+k)$ implique qu'aucun sommet de A ne peut dominer deux sommets de H. Ainsi un ensemble A H-dominant est une approximation de H dans le sens où A contient pour chaque hub de degré différent de 2 exactement un sommet le dominant. A peut contenir ou non des sommets dominant des hubs de degré 2. Dans tous les cas, un sommet ne peut toujours dominer qu'un seul hub et un hub ne peut être dominé que par un seul sommet.

Les hubs de degré 2 respectent des propriétés différentes car ils peuvent être entièrement dominés par le K-voisinage d'un plus court chemin. Cependant, si le cas des hubs de degré 2 empêche de définir de façon unique le graphe quotient, ce n'est pas le cas du graphe quotient réduit qui est quant à lui défini sans ambiguïtés.

Ce dernier point fait apparaitre une difficulté particulière dans les configurations correspondant à un cycle où le graphe quotient ne contient qu'un seul hub de degré 3. Les trois laminaires à gauche de la figure 4.2 en sont un exemple. Ceci sera appelé une *configuration problématique*. Dans cette configuration, il existe au moins un hub $h \in H$ de degré 2 dans le cycle mais il est possible de ne pas réussir à le détecter.

Dans ce cas, un sommet b est ajouté au milieu du cycle et est ajouté à un ensemble B qui sera renvoyé au côté de A par $TrouverHubs$. Si les sommets de A sont prouvés comme correspondant à des hubs de H, ceux de B pourront être modifiés par la suite.

Les laminaires sont déterminés dans une seconde étape par la fonction $ChercheLaminaires$, cette dernière correspondant grossièrement à relier les hubs précédemment calculés par des plus courts chemins. Le cas particulier des hubs de degré 2 doit être cependant pris en compte, certains pouvant être détectés à cette étape alors qui ne l'avaient pas été par la fonction $TrouverHubs$. Dans ce cas, l'ensemble de sommets A est modifié et les nouveaux hubs sont ajoutés. Des sommets peuvent être également supprimés de B si nécessaire.

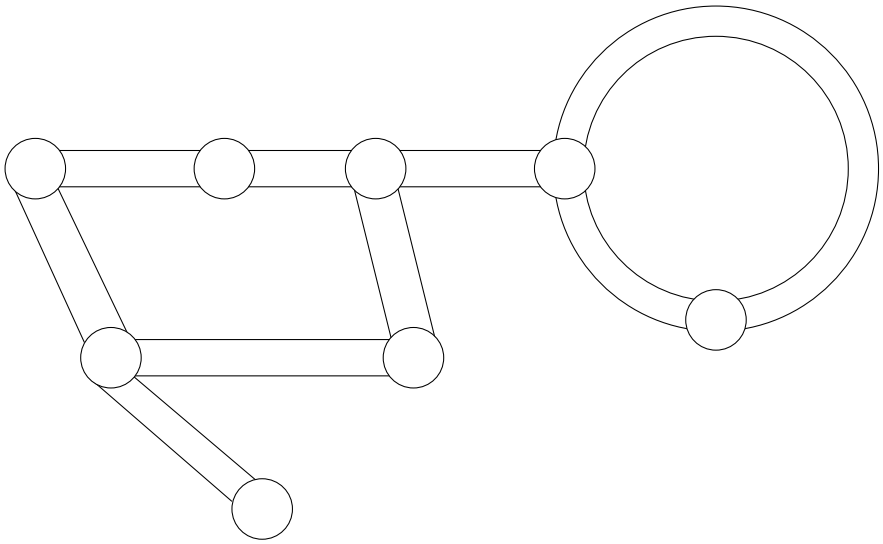

FIGURE 4.5

Les figures 4.5, 4.6 et 4.7 illustrent ces deux étapes en exhibant une sortie possible de $TrouverHubs$ et $ChercheLaminaires$ sur un exemple donné. La figure 4.8 montre la sortie de l'algorithme sur un petit exemple possédant un hub de degré 3 et un cycle.

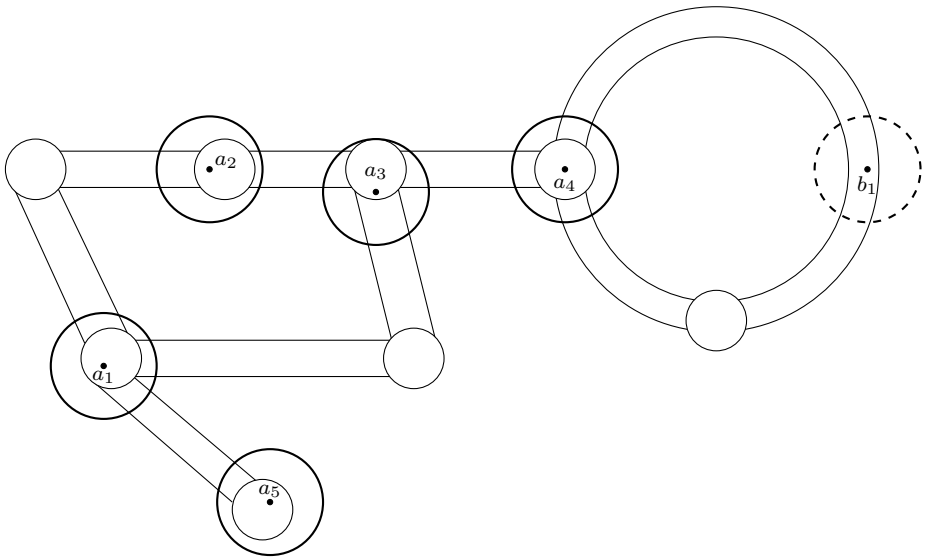

FIGURE 4.6 – Durant l'exécution de *TrouverHubs*, les centres de hub sont calculés de façon à ce que tout hub $B(h, r), h \in H$ de degré différent de 2 est couvert par $B(a_i, R), a_i \in A$. Les centres de hubs déplaçables comme b_1 peuvent être ajoutés au milieu des configurations problématiques.

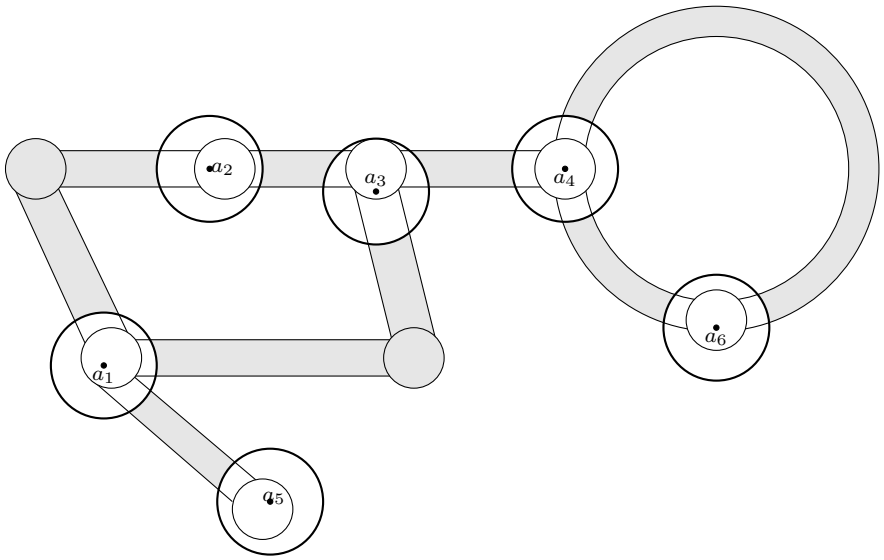

FIGURE 4.7 – Durant l'exécution de *ChercheLaminaires*, des hubs déplaçables peuvent être modifiés, ici b_1 devient a_6. Des hubs de degrés 2 suffisamment fins ne sont pas détectés et appartiennent alors à des K-laminaires. Nous illustrons le cas où un hub de degré 2 n'est trouvé que lors de la seconde étape, ici a_6. Dans tous les cas, que les hubs de degré 2 soient détectés ou non, le graphe quotient réduit est identique.

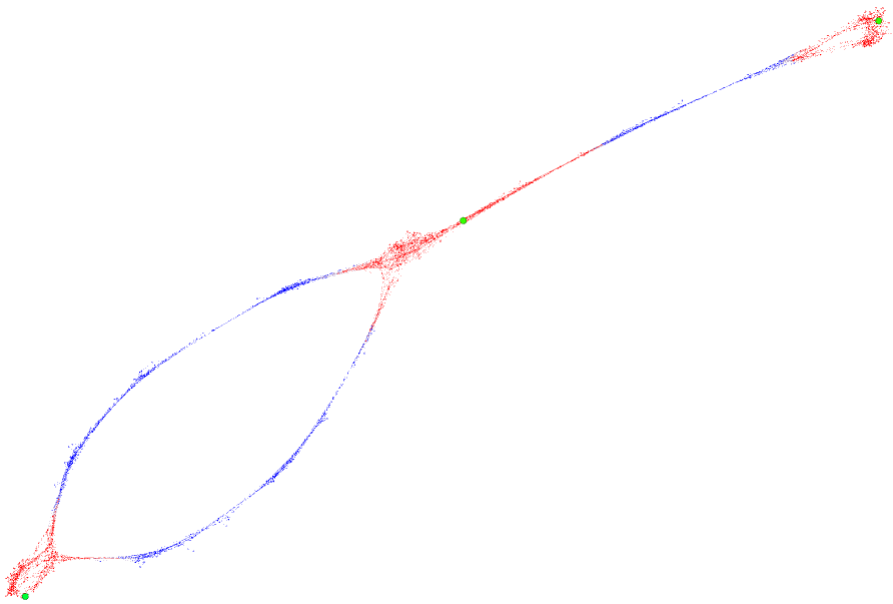

FIGURE 4.8 – Exemple d'une sortie de l'algorithme. Les sommets en rouge appartiennent à des hubs, ceux en bleu à des laminaires et ceux en vert sont des centres de hub.

4.3.3 Topologie de $G \setminus B(A, R)$

Les fonctions *TrouverHubs* et *ChercheLaminaires* utilisent des arbres *BFS* qui couvrent les composantes connexes de $G \setminus B(A, R)$, A étant H-proche. Avant de rentrer dans le détail des fonctions, explicitons les différentes topologies que de telles composantes peuvent avoir. Les figures 4.9, 4.10, 4.11 illustrent les topologies possibles.

Lemme 13.

Soit A un ensemble H-proche et g une composante connexe de $G \setminus B(A, R)$. g respecte une de ces topologies, mutuellement exclusives :

Type a) : *g ne contient aucun hub et touche un unique ensemble $B(a, R)$, $a \in A$;*

Type b) : *g contient un hub de degré au moins trois ;*

Type c) : *il existe une suite de hubs et laminaires H_0, L_1, H_1,..., L_z, H_z, $z \geq 1$, telle que le centre h_0 de H_0 est dominé, H_z est de degré 1, tous les autres hubs (si $z \geq 2$) sont de degré 2. g est composé de l'union des sommets de ces hubs et laminaires sauf ceux de $B(A, R)$.*

Type d) : *il existe une suite de hubs et laminaires H_0, L_1, H_1,..., L_z, H_z, $z \geq 1$ et $H_z \neq H_0$, telle que les centres de H_0 et H_z sont dominés, tous les autres hubs (si $z \geq 2$) sont de degré 2. g est composé de tous les sommets de ces ensembles sauf ceux de H_0, H_z et ceux de L_1 et L_z inclus dans $B(A, R)$;*

Type e) (configuration problématique) : *il existe une suite de hubs et laminaires H_0, L_1, H_1,..., L_z, H_0, $z \geq 1$, telle que le centre de H_0 est dominé,*

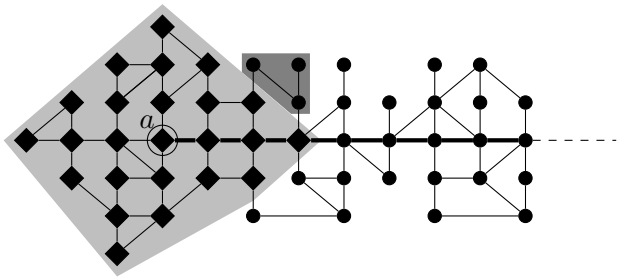

FIGURE 4.9 – Exemple d'une composante connexe de Type a). Le chemin laminaire est en gras, $k = 2$, $R = 3$ et les sommets carrés sont dans $B(a, R)$ en gris clair. La composante de Type a), en gris foncé, est formée de 3 sommets. Ils sont tous trois à distance plus de R de a et à distance au plus de k du chemin laminaire. Cependant $B(a, R)$ les sépare du reste du laminaire.

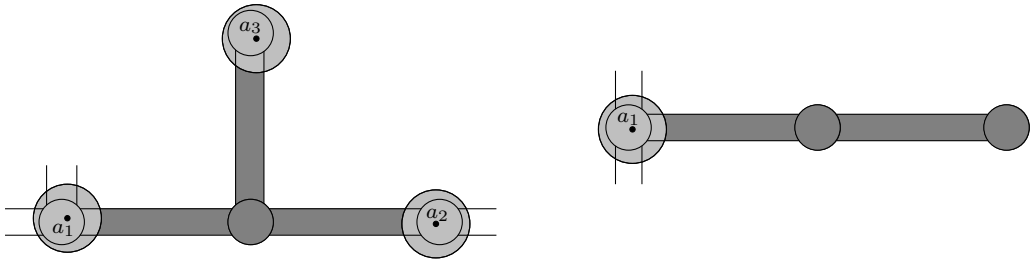

FIGURE 4.10 – Exemples d'une composante connexe de Type b) (à gauche) et d'une composante de Type c) (à droite). Les sommets a_1, a_2 et a_3 correspondent à des sommets de A déjà détectés. $B(A, R)$ est coloriée en gris clair. Les composantes connexes considérées sont coloriées en gris foncé.

tous les autres hubs sont de degré deux. g est composé de tous les sommets de ces ensembles sauf ceux de H_0 et ceux de L_1 et L_z inclus dans $B(A, R)$.

De plus, tout sommet voisin de $B(a, R)$, $a \in A$, appartient à un laminaire incident à $B(h, r)$, $h \in H$ étant le sommet dominé par a.

Démonstration. Comme $r + (K + 2r + k) < R$, si $a \in A$ domine $h \in H$ alors $B(h, r) \subset B(a, R)$. Réciproquement, $B(a, R)$ ne peut toucher deux hubs différents car ceci impliquerait $\ell < 2(R + r)$. Ainsi, si A est H-proche, $B(h, r)$ n'intersecte pas $B(A, R)$ si $B(h, r) \notin B(A, R)$. Tout hub de H est donc soit totalement inclus, soit d'union disjointe avec $B(A, R)$.

Soit une composante connexe de $G \setminus B(a, R)$ qui ne contient pas de hub de H. Cette composante est donc incluse dans un laminaire. Soit elle est voisine d'un unique ensemble $B(a, R)$ correspondant au Type a), soit elle relie deux ensembles $B(a, R)$ et $B(a', R)$, correspondant au Type d) avec $z = 1$.

Considérons maintenant le graphe quotient tel que ses sommets sont colorés en rouge si le hub correspondant est inclus dans $B(A, R)$, en noir sinon. Une composante connexe de $G \setminus B(A, R)$ contenant un hub correspond à un sous-graphe connecté maximal contenant des sommets noirs. Si ce sous-graphe contient un sommet de

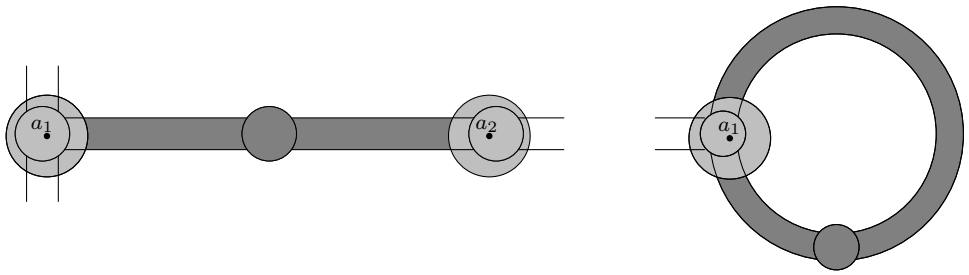

FIGURE 4.11 – Exemples d'une composante connexe de Type d) (à gauche) et d'une composante de Type e) (à droite). Les sommets a_1, a_2 et a_3 correspondent à des sommets de A déjà détectés. $B(A, R)$ est coloriée en gris clair. Les composantes connexes considérées sont coloriées en gris foncé.

degré 3, la composante est de Type b). Sinon, cela correspond à un chemin dans le graphe quotient. Si seulement une des extrémités a un voisin rouge, la composante est de Type c). Si les deux extrémités ont des voisins rouges différents, alors la composante est de Type d) avec $z \geq 2$. Si les deux extrémités ont le même voisin rouge, alors la composante est de Type e).

Pour finir, un laminaire liant deux hubs non-dominés est déconnecté du reste du graphe par ces deux hubs, donc aucun ensemble $B(a, R)$ ne peut le toucher ou en être voisin, ce qui implique la dernière affirmation du lemme. □

4.3.4 Recherche des hubs

L'algorithme de détection des hubs *TrouverHubs* utilise une coloration des sommets de G, ces derniers étant initialement non colorés. Les sommets sont colorés graduellement par la procédure *HubSuivant*, et certains sont ajoutés aux ensembles A ou B de façon à ce que les invariants suivants soient vérifiés à chaque étape :

Invariant 1 : L'ensemble U des sommets non colorés est une union de composantes connexes de $G \setminus B(A, R)$.

Invariant 2 : A est H-proche.

Invariant 3 : Tout $h \in H$ coloré correspondant à un hub de degré différent de 2 est dominé par un sommet de A.

Pour commencer, *TrouverHubs* requiert un premier sommet $a \in A$ dominant un sommet $h \in H$. colorer les sommets de $B(a, R)$ garantit ensuite que les trois invariants sont vérifiés. *TrouverHubs* consiste ensuite à exécuter tant que possible la fonction *HubSuivant*, introduite en section 4.3.4.2, qui colore un ensemble non vide de sommets non colorés en préservant les trois invariants. *HubSuivant* est appelée jusqu'à ce qu'aucun sommet de A n'ait un sommet non coloré à distance $R + 1$. L'invariant 1 garantit que l'intégralité du graphe est coloré, les invariants 2 et 3 impliquent ensuite que A est H-dominant.

L'initialisation de la fonction correspondant au choix du premier sommet $a \in A$ est reporté à la section 4.3.4.3 car elle utilise la fonction *HubSuivant*.

HubSuivant fonctionne à partir d'un BFS prenant racine à la frontière de $B(A,R)$ dans une composante connexe de $G \setminus B(A,R)$. Les composantes de type a) ou e) sont caractérisées par le fait que le sommet le plus éloigné dans le BFS est proche de la boule $B(a,R)$ dont la racine est voisine. Ainsi, les sommets non colorés proches de a ont besoin d'être *marqués* pour tester si l'on s'arrête près de a ou non. Ce marquage est effacé quand le BFS est terminé.

4.3.4.1 La fonction StopBFS

La fonction **StopBFS** prend en entrée un sommet marqué et non coloré d ainsi qu'une couleur c, elle calcule ensuite un BFS commençant sur d et respectant les règles suivantes :

- seuls les sommets non colorés sont mis dans la file du BFS.
- si un sommet visité a un voisin coloré d'une couleur différente de c, le BFS s'arrête, le dernier sommet visité est noté f.
- si le BFS se finit sans que le cas précédent ne soit apparu, on note f la feuille la plus profonde telle qu'il existe un sommet non marqué sur la branche de d à f. Si une telle feuille n'existe pas, tous les sommets parcourus par le BFS étaient marqués et alors $f := d$.
- la fonction $StopBFS(d,c)$ renvoie l'arbre BFS -potentiellement partiel- T ainsi que le chemin Q de d à f dans cet arbre.

La fonction *StopBFS* est appelée à la première étape de *HubSuivant*. Notons que l'invariant 1 implique alors que tous les sommets explorés correspondent à une composante connexe de $G \setminus B(A,R)$, ou à un sous-graphe d'une telle composante si une autre couleur que c a été rencontrée lors du parcours BFS.

4.3.4.2 La fonction HubSuivant

La fonction *HubSuivant* peut maintenant enfin être décrite. Elle dépend du résultat suivant, qui permet de détecter des hubs, c'est à dire d'ajouter à A des sommets proches de sommets de H.

Lemme 14 (Détection de hub)**.** *Soit Q un plus court chemin calculé par un StopBFS ou même n'importe quel plus court chemin de G. Notons $r_{3K}(Q)$ le sous-chemin de Q obtenu en supprimant les $3K$ premiers et derniers sommets de ce chemin.*

Supposons qu'il existe triplet de sommets (u,v,w) tel que $u \in r_{3K}(Q)$, $vw \in E(G)$, $d(u,v) = K$ et $d(Q,w) = K+1$. Alors il existe un centre de hub $h \in H$ dominé par u.

Réciproquement, supposons que l'ensemble des sommets explorés par le StopBFS contienne $B(h, 4K+3k+r+2)$ avec $h \in H$ un centre de hub de degré au moins 3, les sommets d et f du StopBFS étant hors de cet ensemble. Supposons de plus que le chemin Q renvoyé par le StopBFS intersecte le hub $B(h,r)$. Alors, il existe un sommet $u \in r_{3K}(Q)$ et une arête $vw \in E(G)$ tels que $d(u,v) = K$ et $d(Q,w) = K+1$.

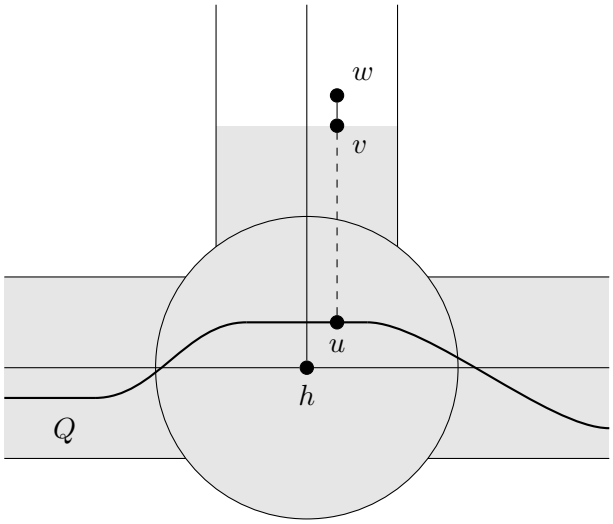

FIGURE 4.12 – Illustration du Lemme 14

Le lemme suivant décrit le comportement de la fonction *StopBFS* suivant la structure du sous-graphe exploré, ceci permettant l'existence d'un algorithme basé sur ce que retourne la fonction *StopBFS*.

Lemme 15. *Supposons que les invariants* 1 *et* 3 *sont satisfaits et soit* $a \in A$. *Supposons de plus que l'ensemble des sommets marqués est* $B(a, R + 2K + 1)$ *et qu'un StopBFS est calculé à partir d'un sommet* d *tel que* $d(a, d) = R+1$. *Soit* g *la composante connexe de* $G \setminus B(A, R)$ *explorée -partiellement ou complètement- par le StopBFS, notons* Q *le chemin renvoyé par le StopBFS et* f *son dernier sommet.*

Suivant le type de topologie de g *définie au lemme 13, les assertions suivantes sont vérifiées :*

Type a) : *tous les sommets explorés sont marqués et donc* $f = d$.

Type b) : *il existe un triplet de sommets satisfaisant la condition du lemme 14.*

Type c) : f *et son voisinage ne sont ni marqués, ni colorés, et* f *domine* h_z.

Type d) : f *n'est ni marqué, ni coloré et est voisin d'un sommet coloré appartenant à* $B(a', R)$, $a \neq a'$.

Type e) : f *est marqué et* $f \neq d$.

La fonction *HubSuivant*, dont le pseudo-code est donné par l'algorithme 5, consiste à déterminer lequel de ces cinq cas correspond à ce que renvoie la fonction *StopBFS*. Ceci est effectué en testant la présence d'un triplet de sommets satisfaisant le lemme 14, ou en regardant la coloration de f et de ses voisins. Notons qu'un triplet satisfaisant le lemme 14 peut aussi exister dans des composantes de type c), d) ou e), le hub détecté est alors de degré 2.

Dans tous les cas, soit un sommet dominant un hub $h \in H$ non coloré est ajouté à A, soit une composante connexe de $G \setminus B(A, R)$ ne contenant ou aucun ou uniquement des hubs de degré 2 est entièrement colorée. Dans le cas correspondant à

HubSuivant
Entrée : Un graphe G, des entiers R et K, des ensembles de sommets A et B (B peut être vide) et un sommet $a \in A$
Sortie : Les ensembles A, B mis à jour et une coloration des sommets
Marquer tous les sommets non colorés de $B(a, R + 2K + 1)$
Sélectionner un sommet d marqué et à distance $R + 1$ de a
Soit $(T, Q) = stopBFS(d, col(s))$ et f le dernier sommet de Q
Si $f = d$ **alors**
    `/* Cas a) */`
    colorer tous les sommets de T avec la couleur lam
sinon si $\exists w, u$ *t.q.* w *est non coloré,* $u \in r_{3K}(Q)$, $d(w,u) = K+1$ *et* $d(w,Q) = K+1$ **alors**
    `/* Un triplet satisfaisant le lemme 14 est trouvé */`
    Ajouter à A le premier sommet u de $r_{3K}(Q)$ satisfaisant la condition ci-dessus
    colorer tous les sommets dans $B(u, R)$ avec une nouvelle couleur $col(u)$
sinon si f *est marqué* **alors**
    `/* Cas e) */`
    Ajouter à B le sommet b au milieu de Q
    colorer tous les sommets de T avec la couleur lam
sinon si f *n'est pas coloré mais a un voisin qui l'est* **alors**
    `/* Cas d) */`
    colorer tous les sommets de T avec la couleur lam
sinon
    `/* Cas c) */`
    Ajouter f à A
    colorer tous les sommets de $B(f, R)$ avec une nouvelle couleur $col(f)$
Effacer le marquage de tous les sommets

Algorithme 5: Pseudo-code de la fonction *HubSuivant*

la configuration problématique, un hub temporaire $b \in B$ est ajouté arbitrairement au milieu de Q.

En supposant l'initialisation *PremierHub* correcte, le lemme suivant implique la validité de *TrouverHubs*.

Lemme 16. *Si HubSuivant est appelé avec en entrée un graphe G coloré et un ensemble A vérifiant les invariants 1, 2 et 3, la sortie de HubSuivant vérifie également ces invariants.*

Démonstration. L'invariant 1 est conservé puisque dans chaque configuration, soit un sommet est ajouté à A et son R-voisinage est coloré, soit une composante connexe de $G \setminus B(A, R)$ est colorée avec la couleur lam.

```
1 TrouverHubs
  Entrée : Un graphe G, des entiers R et K
  Sortie : Des ensembles de sommets A et B
2 Soit A et B deux ensembles vides
3 Ajouter à A PremierHub(G, R, K)
4 Tant que ∃d non coloré à distance R + 1 de a ∈ A faire
5 |  HubSuivant(G, R, K, A, B, a)
6 Renvoyer (A, B)
```

Algorithme 6: Pseudo-code de la fonction $TrouverHubs$

La conservation des Invariants 2 et 3 est une conséquence du lemme 14 si un triplet satisfaisant la condition est trouvé, et est une conséquence du lemme 15 dans le cas c). Les trois autres cas ne modifient pas A et ne colorent pas des hubs de degré différent de 2. □

4.3.4.3 Calcul du premier hub

Pour exécuter la fonction $HubSuivant$ tout en respectant les invariants, un premier sommet de A doit être défini. Ceci est fait en utilisant $HubSuivant$ comme suit.

Un premier BFS est calculé à partir d'un sommet s choisi arbitrairement. Soit x un sommet le plus éloigné de s et Q le plus court chemin de s à x calculé par le BFS. Si Q contient un triplet de sommets (u, v, w) tel que défini dans le Lemme 14 alors u est choisi comme premier hub. Sinon soit m un sommet au milieu de Q et soit d le sommet de Q à distance $R + 1$ de m le plus proche de s.

La fonction $HubSuivant$ est appliquée avec $A = \{m\}$, les sommets de $B(m, R)$ sont colorés et le $stopBFS$ prend pour racine d. Un sommet u dominant un sommet de H est alors forcément détecté d'après le résultat suivant.

Lemme 17. *Supposons que (H, P) contient au moins un hub de degré au moins 3. La procédure pour calculer le premier hub renvoie un sommet u correspondant à la configuration décrite par le lemme 14.*

Il est à noter que ce résultat permet de conclure que nous sommes dans un laminaire ou une configuration cyclique (les deux premiers cas de notre trichotomie) si nous échouons à trouver un premier centre de hub. Ces deux cas ont été traités séparément dans respectivement les chapitres 2 et 3. Dans la suite de ce chapitre nous supposerons que nous sommes dans le cas général, c'est à dire qu'il existe une décomposition avec un hub de degré au moins 3.

4.3.5 Recherche des laminaires

A partir d'un ensemble A H-dominant et un ensemble B correspondant aux configurations problématiques, la fonction $ChercheLaminaires$ calcule des plus

PremierHub
Entrée : Un graphe G, des entiers R et K
Sortie : Un sommet a
Soit s un sommet quelconque
Soit $(T, Q) = BFS(s)$ et x le dernier sommet de Q
Si $\exists w, a$ *t.q.* w *est non coloré,* $a \in r_{3K}(Q)$, $d(w, a) = K+1$ *et* $d(w, Q) = K+1$ **alors**
    Renvoyer a
sinon
    Soit m un sommet au milieu de Q
    Soit d le sommet de Q à distance $\frac{|Q|}{2} - R - 1$ de s
    Soit A un ensemble vide
    Calculer $HubSuivant(G, R, K, A, \emptyset, d)$
    Renvoyer a l'unique sommet de A

Algorithme 7: Pseudo-code de la fonction *PremierHub*

courts chemins entre les sommets de A qui seront les chemins laminaires de la décomposition finale renvoyée.

Ceci est effectué en calculant plusieurs BFS, cette fois avec pour racines les sommets de A. Pour tout chemin Q calculé entre deux centres de hubs a et a', les sommets du laminaire construit correspondant ($B(Q, K)$) sont supprimés du graphe exceptés ceux dans les hubs $B(a, R)$ et $B(a', R)$. Ces derniers sont marqués comme *non-supprimables*. Ce processus est répété jusqu'à ce que le graphe ne contienne plus que les hubs centrés autour des sommets de A.

Une difficulté technique doit cependant être prise en compte. Il s'agit de la possible découverte de nouveaux hubs de degré 2 qui ont pu ne pas être détectés par *TrouverHubs*. Dans le cas général, ceci peut facilement être réglé en ajoutant ces nouveaux hubs à A. Cependant dans le cas de la configuration problématique, la difficulté ne peut être résolue si simplement, le nouveau hub détecté pouvant alors être trop proche d'un hub de B. Afin de résoudre ce problème, nous traitons les sommets de B en premier. A partir de chaque sommet b de B, deux BFS sont lancés jusqu'à atteindre le sommet a de A le plus proche. Ces deux BFS sont lancés dans les deux directions opposées, c'est à dire dans les deux composantes connexes disjointes liant $B(b, R)$ et $B(a, R)$. Si un de ces BFS détecte un nouveau hub suivant le lemme 14 alors ce hub est ajouté à A et b est supprimé de B. Sinon cela signifie que les deux chemins obtenus K-dominent tous les sommets de la composante de type e), b est alors transféré dans A.

Le pseudo-code de *ChercheLaminaires* est donné dans l'algorithme 8.

Lemme 18. *Pendant une exécution de ChercheLaminaires, ces propriétés sont vérifiées :*

1. *Après chaque itération de la boucle* ***Pour*** *ou* ***Tant que****, l'ensemble des som-*

ChercheLaminaires
Entrée : Un graphe G, des entiers R et K
Sortie : Une décomposition hub-laminaire $(A, \mathcal{Q})$
$(A, B) = TrouverHubs(G, R, K)$;
$\mathcal{Q} = \emptyset$;
Marquer tous les sommets comme supprimables ;
Pour tout $a \in A$ **faire**
    Marquer tous les sommets de $B(a, R)$ comme non-supprimables ;
Pour tout $b \in B$ **faire**
    Calculer un BFS avec pour racine b et s'arrêtant au premier sommet $a \in A$;
    Soit Q_1 le chemin de b à a calculé par ce BFS ;
    Calculer un BFS avec pour racine b, n'utilisant pas les sommets de $B(Q_1, K) \setminus (B(b, R) \cup B(a, R))$ et s'arrêtant à a ;
    Soit Q_2 le chemin de b à a calculé par ce BFS ;
    Calculer g, l'union de $B(a, R)$ et de la composante connexe de $G \setminus B(a, R)$ contenant b ;
    colorer dans g les sommets de $B(a, R)$, $B(b, R)$, $B(Q_1, K)$, $B(Q_2, K)$
    Si $\exists$ *un sommet* c *non coloré dans* g **alors**
        Ajouter c à A ;
        Marquer les sommets de $B(c, R)$ comme non-supprimables ;
    sinon
        Ajouter b à A ;
        Marquer les sommets de $B(b, R)$ comme non-supprimables ;
        Supprimer de G les sommets supprimables $B(Q_1, K) \cup B(Q_2, K)$
        Ajouter Q_1 et Q_2 à $\mathcal{Q}$;
Tant que *il existe* $a \in A$ *tel que* $B(a, R+1) \neq B(a, R)$ **faire**
    Calculer un BFS avec pour racine a et s'arrêtant sur le premier sommet $a' \in A$, $a' \neq a$;
    Soit Q le chemin de a à a' calculé par le BFS ;
    Si $\exists w, u$ *t.q.* $u \in r_{3K}(Q)$, $d(w, u) = K+1$, $d(w, Q) = K+1$ **alors**
        Ajouter à A le premier sommet h de Q qui satisfait la condition ci-dessus ;
        Marquer les sommets de $B(h, R)$ comme non-supprimables ;
    sinon
        Ajouter à $\mathcal{Q}$ le chemin Q de a à a' calculé par ce BFS ;
        Supprimer de G les sommets supprimables de $B(Q, K)$;

Algorithme 8: Pseudo-code de la fonction *ChercheLaminaires*

mets supprimables est une union de composantes connexes de $G \setminus B(A, R)$ *de*

type a), d) ou e) ;

2. *Toute composante supprimable de type a) est incluse dans un laminaire qui est aussi le premier ou dernier d'une composante de type d) ou e) ;*
3. *Toute itération de la boucle* ***Pour*** *supprimant un sommet, supprime ou marque comme non supprimable les sommets d'exactement une composante de type e) et toutes les composantes de type a) situées dans son premier ou dernier laminaire ;*
4. *Toute itération de la boucle* ***Tant que*** *supprimant un sommet, supprime ou marque comme non-supprimable exactement une composante de type d) et toutes les composantes de type a) situées dans son premier ou dernier laminaire ;*

Ainsi, ChercheLaminaires se termine avec tous les sommets de G supprimés ou marqués comme non-supprimables.

Présentons maintenant le résultat final servant à prouver la validité de notre algorithme.

Lemme 19. *La sortie $(A, \mathcal{Q})$ de ChercheLaminaires est une (R, K)-décomposition hub-laminaire.*

Démonstration. Nous devons prouver que la décomposition calculée respecte chaque point de la définition d'une décomposition hub-laminaire (définition 8).

1. *Tout laminaire lie deux centres de hubs. Les extrémités a, a' de tout $Q \in \mathcal{Q}$ appartiennent à A et pour tout hub $a'' \in A \backslash \{a, a'\}$, $B(Q, K) \cap B(a'', R+1) = \emptyset$.*
   La première partie de cette affirmation est une conséquence directe de la création des chemins laminaires dans *ChercheLaminaires*. La seconde partie est une conséquence de la dernière affirmation du lemme 13. En effet, $B(Q, K) \cap B(a'', R+1) \neq \emptyset$ impliquerait que la composante connexe couverte par $B(Q, K)$ contiendrait trois laminaires incidents à des hubs dominés et donc contiendrait un hub de H de degré au moins 3 non dominé, ce qui est impossible par le premier résultat du lemme 18.
2. *Les laminaires et les hubs dominent G.* $V(G) = \bigcup_{a \in A} B(a, R) \cup \bigcup_{Q \in \mathcal{Q}} B(Q, K)$.
   Ceci est une conséquence directe de la dernière affirmation du lemme 18.
3. *Tout chemin laminaire est localement un plus court chemin. Tout chemin $Q \in \mathcal{Q}$ d'extrémités a et a' est un plus court chemin dans $G[B(Q, K) \cup B(a, R) \cup B(a', R)]$.*
   Tout chemin $Q \in \mathcal{Q}$ d'extrémités a et a' est calculé par un BFS dans un sous-graphe contenant $B(Q, K) \cup B(a, R) \cup B(a', R)$. C'est donc bien un plus court chemin dans ce dernier ensemble.
4. *Les laminaires se rencontrent uniquement aux hubs. Pour tout $i \neq j$ et $uv \in E(G)$ tels que $u \in B(Q_i, K)$ et $v \in B(Q_j, K)$, il y a un centre de hub $a \in A$ tel que Q_i et Q_j ont tous deux a comme extrémité et tel que $u, v \in B(a, R)$.*

Ceci est une conséquence des deux derniers points du lemme 18, qui affirment qu'une composante connexe de sommets supprimables ne peut à aucun moment être seulement partiellement supprimée.

En effet, supposons qu'il existe des tels sommets u et v qui ne sont pas dans un hub $B(a, R)$, $a \in A$, et supposons sans perte de généralité que Q_i est ajouté à $\mathcal{Q}$ avant Q_j. Considérons l'itération pendant laquelle Q_i est construit. La composante connexe contenant u et v reste soit supprimable, contredisant $u \in B(Q_i, K)$, soit u et v sont tous deux supprimés, contredisant $v \notin B(Q_i, K)$.

□

La $(K + 2r + k)$-équivalence est une conséquence du fait que A est H-dominant. Ceci nous permet de construire une bijection ϕ entre les centres de hubs de degré différents de deux de $(H, \mathcal{P})$ et $(A, \mathcal{Q})$. De plus notre décomposition $(A, \mathcal{Q})$ a au plus λ hubs, puisqu'elle n'a pas plus de hubs de degrés 2 que $(H, \mathcal{P})$. En effet notre algorithme ajoute des hubs de degré 2 uniquement quand les conditions du lemme 14 sont réunies ou quand un sommet de B est transféré à A, ce dernier cas n'arrivant que quand un hub de degré 2 n'a pas été détecté.

Concernant la complexité temporelle, à part dans le cas (a), chaque itération de la boucle **Tant que** dans *TrouverHubs* correspond à la détection d'un hub ou d'un laminaire. Il y a donc au plus $O(|A| + |\mathcal{Q}|)$ itérations de ce type, leur coût total est de $O(nm)$. Lors des itérations correspondant au cas (a), tous les sommets visités par le *StopBFS* sont colorés : le coût total de ces itérations est donc $O(m)$. De la même manière, *ChercheLaminaires* consiste en λ itérations, chacune de complexité $O(m)$.

4.3.6 Preuves

4.3.6.1 Deux lemmes techniques

Deux lemmes techniques sont nécessaires avant de s'attaquer aux lemmes non prouvés de la section précédente. Le premier montre qu'un plus court chemin qui entre dans un laminaire mais ne le traverse pas, ne peut y entrer profondément. Ceci est illustré par la figure 4.13.

Lemme 20. *Soit un plus court chemin Q dans le graphe induit par $B(P, K)$ avec $P \in \mathcal{P}$ et soit trois sommets successifs a, m, b dans Q avec a', m', b' dans P tels que $d_G(a, a') \leq k$, $d_G(m, m') \leq k$ et $d_G(b, b') \leq k$.*

Si a' est entre b' et m' dans P, alors $d_G(a, m) \leq 3k$.

Démonstration.

$$\begin{aligned} d(m, a) &= d(a, b) - d(m, b) \\ &\leq d(a', b') + 2k - d(m, b) \\ &\leq d(m', b') - d(m', a') + 2k - d(m, b) \end{aligned}$$

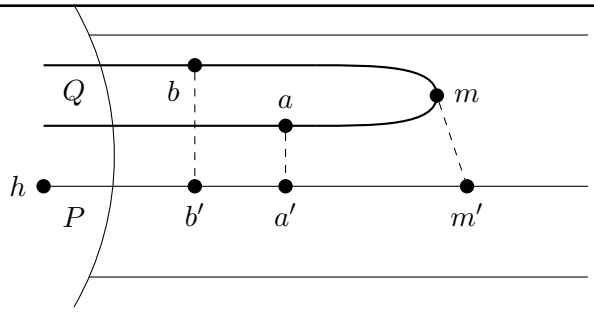

FIGURE 4.13 – Preuve du Lemme 20

Comme $d(m', b') \leq d(m, b) + 2k$ et $d(m', a') \geq d(m, a) - 2k$, il découle,

$$d(m, a) \leq 6k - d(m, a)$$
$$d(m, a) \leq 3k$$

□

Le second lemme montre que si un hub $B(h, r)$ déconnecte une composante connexe couverte par un $StopBFS$, alors le sommet f retourné par le $StopBFS$ ne peut être proche de h.

Lemme 21. *Soit un sous-graphe connexe g de G, un hub $B(h, r)$ inclus dans g, un sommet $d \in g \setminus B(h, r)$, et un laminaire L incident à $B(h, r)$. Soit $v_0 = h, v_1, \ldots, v_q$, $q > 2r + 1$, un sous-chemin du chemin laminaire L appartenant à g et supposons que $B(h, r)$ sépare v_q de d dans g.*

Notons f un sommet le plus éloigné de d dans g. Alors $f \notin L \cap B(h, q - 2r - 1)$.

Démonstration. Notons S un plus court chemin dans g de d à v_q. Comme $B(h, r)$ sépare d de v_q, S contient un sommet u de $B(h, r)$. Alors,

$$d_g(d, v_q) = d_g(d, u) + d_g(u, v_q) \geq d_g(d, u) + d_g(h, v_q) - d_g(h, u) \geq d_g(d, u) + q - r$$

De plus, pour tout $w \in B(h, q - 2r - 1)$,

$$d_g(d, w) \leq d_g(d, u) + d_g(u, h) + d_g(h, w) \leq d_g(d, u) + r + q - 2r - 1 < d_g(d, v_q)$$

Donc, $f \notin L \cap B(h, q - 2r - 1)$. □

4.3.6.2 Preuve du lemme 14

Démonstration du premier paragraphe

Supposons l'existence d'un triplet (u, v, w) de sommets satisfaisant la condition et, par l'absurde, supposons qu'aucun hub h n'existe à distance au plus $K + 2r + k$ de u. u appartient donc à un laminaire $B(P, k)$, $P \in \mathcal{P}$ reliant deux centres de hubs h_1 et h_2 de H.

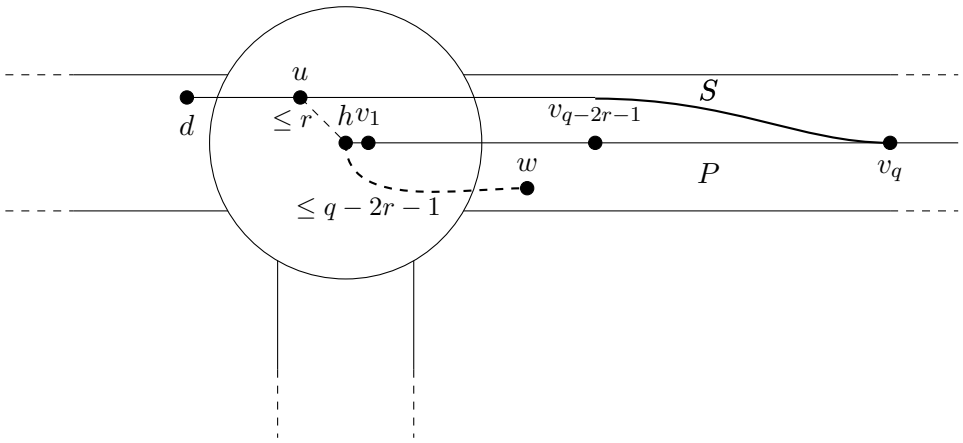

FIGURE 4.14 – Notations utilisées dans la preuve du Lemme 21

Supposons en premier que w n'appartienne ni à $B(P,k)$, ni à un hub $B(h_1,r)$ ou $B(h_2,r)$. Le plus court chemin de u à v contient alors un sommet x dans $B(h_1,r)$ ou $B(h_2,r)$, ainsi $d(u,h_1) \leq d(u,x)+d(x,h_1) \leq K+r$ ou $d(u,h_2) \leq K+r$. u couvre alors un sommet de H.

Supposons maintenant que $w \in B(P,k) \setminus (B(h_1,r) \cup B(h_2,r))$. Soit a et b deux sommets tels que $Q_{ab} \subset B(P,k)$, $u \in Q_{ab}$ et tel que que Q_{ab} soit maximal. Alors a (resp. b) est une extrémité de Q ou un sommet de $B(h_1,r)$ ou $B(h_2,r)$. Dans tous les cas $d(a,u) \geq K+r+k$ et $d(b,u) \geq K+r+k$.

Soit b', a', w' et u' des sommets de P à distance au plus k de b, a, w et u.

Le lemme 20 et le fait que a et b soient à distance plus de $3k$ de u implique que u' est entre a' et b' dans Q. De plus, w' ne peut être entre a' et b' car ceci impliquerait par le lemme 1 que w est à distance au plus $3k$ de $Q'_{a,b}$. On peut donc supposer que h_1, w', a' et u' sont dans cet ordre dans P.

Alors $d(u,a) \leq d(u',a')+2k \leq d(u',w')+2k \leq d(u,w)+4k = K+4k+1$. Comme $u \in r_{3K}(Q)$, a n'est pas une extrémité de Q, et on peut supposer que $a \in B(h_1,r)$. Alors $d(h_1,w') \leq d(h_1,a')-1 \leq r+k-1$ et donc $d(u,h_1) \leq d(u,w)+d(w,w')+d(w',h_1) \leq K+1+k+r+k-1 \leq K+2r+k$. Ainsi u domine h_1.

Démonstration du second paragraphe

Supposons maintenant $h \in H$ correspondant à un hub de degré au moins 3 et tel que le *StopBFS* a sa racine hors de $B(h, \frac{\ell}{2}-R)$ mais explore cet ensemble. Supposons de plus que le chemin Q en sortie du *StopBFS* traverse $B(h,r)$, d et f n'appartenant pas à $B(h, \frac{\ell}{2}-R)$.

Soit trois chemins P_i, P_k et P_l de $\mathcal{P}$ avec h comme extrémité et respectivement les sommets x'_i, x'_j et x'_l sur ces chemins, chacun à distance $r+K+3k+2$ de h, comme illustré par la figure 4.15.

Supposons dans un premier temps que ces trois sommets sont respectivement à distance au plus K de sommets x_i, x_j et x_l dans Q. Aucun de ces trois sommets n'appartient au hub $B(h,r)$ comme $d(h,x_i) \geq d(h,x'_i)-d(x'_i,x_i) \geq r+3k+2$. De

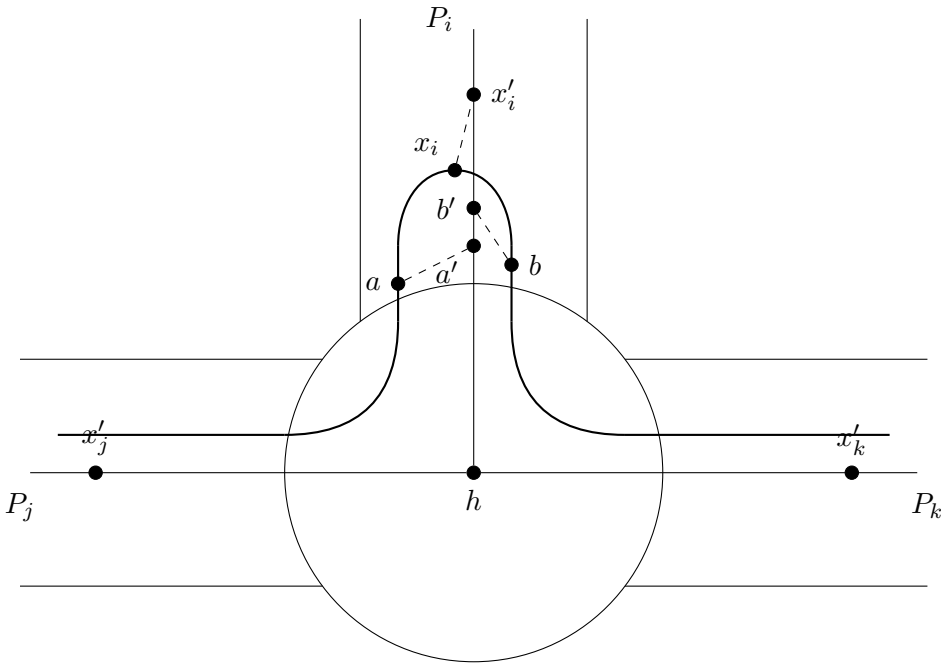

FIGURE 4.15 – Preuve du Lemme 14

plus, on peut supposer sans perte de généralité que x_j, x_i et x_l apparaissent dans cet ordre dans Q. Il existe alors un sous-chemin maximal Q_{ab} de Q qui est inclus dans $B(P_i, k) \setminus B(h, r)$ et contient x_i.

Soit a' et b' les sommets de P_i tels que $d(a, a') \leq k$ et $d(b, b') \leq k$. Alors $d(h, a') \leq d(h, a) + k \leq r + k + 1$ et pareillement pour b. Comme $d(h, x'_i) > r + k + 1$, en appliquant le lemme 20 à a, x_i et b nous avons $d(a, x_i) \leq 3k$ ou $d(b, x_i) \leq 3k$. Dans tous les cas comme $d(h, a) = d(h, b) = r + 1$ et $d(x_i, x'_i) \leq K$, nous avons $d(h, x'_i) \leq r + K + 3k + 1$, ce qui est une contradiction.

Un de ces sommets, x'_i, x'_j ou x'_k est donc à distance plus de K de Q, par exemple x'_i. En prenant P_i de h à x'_i, soit v le dernier sommet à distance K de Q, soit w le sommet suivant de P_i et soit u le sommet de Q à distance K de v. Alors $d(Q, w) = K + 1$.

Si u n'appartient pas au laminaire $B(P_i, k)$, le plus court chemin de u à v doit traverser $B(h, r)$ et $d(u, v) \leq K + r$. Si $u \in B(P_i, k)$, soit u' un sommet de P_i à distance au plus k de u. Par définition de v, u' est entre h et v, et $d(u', v) \geq K - k$ comme $d(u, v) = K$. Ainsi, $d(h, u') \leq d(h, v) - K + k \leq d(h, x'_i) - K + k \leq r + 4k + 2$ et $d(h, u) \leq r + 5k + 2$. Dans tous les cas, $d(h, u) \leq r + K + 2k + 2 \leq K + 2r + k$.

Par conséquence, $d(u, f) \geq d(h, f) - d(h, u) \geq 4K + 3k + r + 2 - r - K - 2k - 2 > 3K$ et de même $d(u, d) \geq 3K$. u appartient donc à $r_{3K}(Q)$.

4.3.6.3 Preuve du lemme 15

Supposons que les invariants définis en début de section 4.3.4 sont vérifiés et soit a un sommet de A. Supposons que l'ensemble des sommets marqués est $B(a, R + 2K + 1) \setminus B(a, R)$ et qu'un *StopBFS* est calculé à partir d'un sommet d tel que $d(a, d) = R + 1$. Soit g la composante connexe de $G \setminus B(A, R)$ explorée par le *StopBFS*, et notons Q et f le chemin retourné par le *StopBFS* et son dernier sommet. Notons que g est un sous-graphe de G, donc $d_g(u, v) \geq d_G(u, v)$ pour tous sommets u et v.

Avant de nous attaquer au coeur de la preuve, montrons deux résultats intermédiaires concernant la position de f dans g. Soit H_0 le hub ayant pour centre $h_0 \in H$ dominé par a, soit L_1 le laminaire incident à H_0 contenant d, et soit H_1 l'autre hub incident à H_1, h_1 dénotant son centre.

Le premier lemme intermédiaire concerne le comportement du *StopBFS* dans L_1, les distances dans g et G pouvant être ici significativement différentes.

Lemme 22. *Supposons que le StopBFS explore un sommet non marqué. Une de ces affirmations est alors vérifiée :*

1. *g est de type c) avec $z = 1$ et f a un voisin coloré ;*
2. *f domine h_1 ;*
3. *f est hors de $L_1 \cup H_1$;*

En particulier, la composante connexe explorée n'est pas de Type a).

Démonstration. Supposons qu'aucune de ces affirmations ne soit vraie. h_1 n'est alors pas dominé, ou alors la première affirmation serait vraie. De plus, comme $H_1 \subset B(h_1, K + 2r + k)$, f est un sommet de $L_1 \setminus H_1$.

Soit u le premier sommet non marqué de Q, et soit d', u' et f' des sommets de P_1 à distance au plus k de respectivement d, u et f.

Supposons dans un premier temps que $f \in B(a, R+K)$. Alors, comme $d_G(a, d) = R+1$ et $d_G(a, u) \geq R+2K+2$, Q contient deux sommets v_1 et v_2 tels que $d_G(a, v_1) = d_G(a, v_2) = R + K + 1$ et $u \in Q_{v_1v_2}$. De plus, $d_g(v_i, u) \geq d_G(v_i, u) \geq d_G(a, u) - d_G(a, v_i) = K + 1$, $1 \leq i \leq 2$. Les sommets v_1' et v_2' de P_1 qui sont à distance au plus k de v_1 et v_2 sont tous deux plus proches de h_0 que de u'. Comme v_1 et v_2 sont à distance au moins $K + 1$ de $B(a, R)$, $Q_{v_1v_2}$ est également un plus court chemin les reliant dans G, et cette configuration contredit le lemme 20. Ainsi f n'est pas dans $B(a, R + K)$.

Les notations du paragraphe précédent sont illustrées par la figure 4.16.

Le paragraphe précédent implique que d', f' et h_1 sont dans cet ordre dans P_1. Le lemme 1 implique donc que le plus court chemin dans g reliant d à h_1 $3k$-domine f : il existe x sur ce chemin tel que $d_G(x, f) \leq 3k$. Notons dans un premier temps que comme f n'appartient pas à $B(a, R + K)$, aucun des sommets sur le plus court chemin de f à x n'appartient à $B(a, R)$, la même affirmation tient pour le sous-chemin de P_1 reliant x à h_1. Ainsi, h_1 appartient à la même composante que f, c'est-à-dire g. Par conséquence, g n'est pas de Type a).

De plus, comme f est à distance plus de $3k$ de $B(a,R)$, $d_G(x,f) \leq 3k$ implique $d_g(x,f) \leq 3k$. Ainsi, comme f est le sommet le plus éloigné de d dans g et x est sur le plus court chemin de d à h_1, $d_g(x,h_1) \leq 3k$. Finalement, $d_G(h_1,f) \leq d_g(h_1,f) \leq 6k \leq K+2r+k$ et f domine h_1.

Les notations du paragraphe précédent sont illustrées par la figure 4.17. □

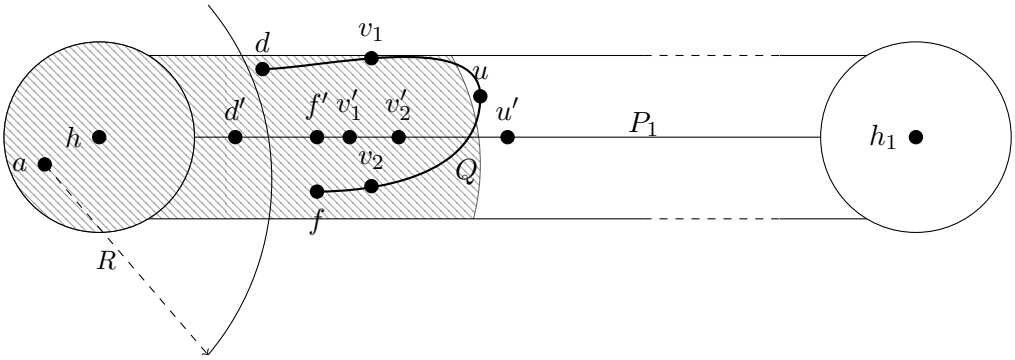

FIGURE 4.16 – Notations utilisées dans la première partie de la preuve du Lemme 22

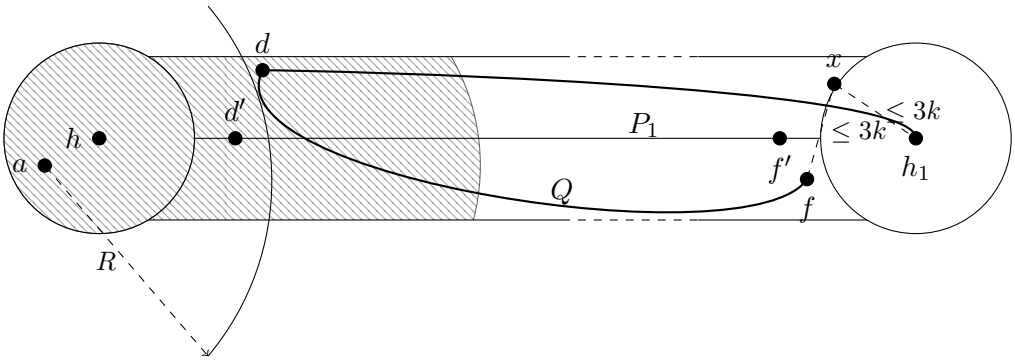

FIGURE 4.17 – Notations utilisées dans la seconde partie de la preuve du Lemme 22

Lemme 23. *Supposons que le StopBFS explore un sommet non marqué. Soit une séquence maximale de laminaires et hubs L_1,H_1,..., L_z,H_z,L_{z+1}, telle que tous les hubs soient dans g et telle que tout hub H_i avec $i \leq z-1$ est de degré 2. Soit v*

le dernier sommet appartenant à g du chemin laminaire P_{z+1} de L_{z+1}. Si L_{z+1} est entièrement inclus dans g, alors v est le centre du second hub incident à L_{z+1}.

Alors, $f \in B(v, 6k)$ ou f n'appartient pas à $\bigcup_{1\leq i\leq z} H_i \cup \bigcup_{1\leq i\leq z+1} L_i$.

Démonstration. Si $z = 0$, la séquence est limitée à L_1 et le lemme 22 s'applique. Supposons $z \geq 1$. Par le lemme 22, f n'appartient pas à L_1.

Soit un hub H_i de centre h_i, $1 \leq i \leq z$ et soit x un sommet au milieu du chemin laminaire P_{i+1} de L_{i+1}. La partie de P_{i+1} entre h_i et x appartient à g et, comme tout hub de H_1 à H_{i-1} est de degré 2, H_i sépare d de x. Par le lemme 21, et comme $r + 6k < \frac{\ell}{2} - 2r - 1$, f n'appartient donc pas à $\cup_{1\leq i\leq z} B(h_i, r + 6k)$.

Supposons maintenant que $z \geq 2$ et soit un sommet u dans $L_i \setminus (B(h_{i-1}, r + 6k) \cup B(h_i, r + 6k))$. Soit S un plus court chemin dans g de d à h_i, et w un sommet dans $S \cup L_i$ voisin de H_{i-1}. Un tel sommet existe nécessairement comme il n'y a que des hubs de degré 2 entre L_1 et L_i. Soit u' et w' des sommets de P_i à distance au plus k de respectivement u et w. Alors,

$$d(h_{i-1}, w') \leq d(h_{i-1}, w) + d(w, w') \leq r + k + 1$$

et

$$d(h_{i-1}, u') \geq d(h_{i-1}, u) - d(u, u') \geq r + 5k$$

Ainsi, u' est entre w' et h_i sur P_i, par le lemme 1, il existe x de S tel que $d_g(x, u) = d_G(x, u) \leq 3k$. Il découle,

$$\begin{aligned} d_g(d, h_i) &= d_g(d, x) + d_g(x, h_i) \\ &\geq d_g(d, x) + d_g(x, u) - 3k + d_g(x, h_i) \\ &\geq d_g(d, u) - 3k + d_g(x, h_i) \end{aligned}$$

De plus comme x est à distance $3k$ de u et ce dernier à distance au moins $6k+1$ de h_i, $d_g(x, h_i) > 3k$. Finalement, $d_g(d, h) > d_g(d, u)$, implique que u ne peut être le sommet à distance maximale de d. f n'est donc pas un sommet de L_i.

Les notations du paragraphe précédent sont illustrées par la figure 4.18.

Pour finir, soit u un sommet dans $L_{z+1} \setminus (B(h_z, r + 6k) \cup B(v, 6k))$. Avec une preuve similaire au dernier paragraphe, v prenant la place de h_i. On démontre que le plus court chemin S dans g de d à v doit $3k$-dominer u. Ceci implique que u est plus proche de d que de v car $d_g(u, v) \geq 6k + 1$. Il découle que f n'est pas dans $L_{z+1} \setminus B(v, 6k)$.

□

Montrons maintenant le lemme 15 en considérant une à une les différentes topologies listées dans le lemme 13.

Type a) Le lemme 22 implique que le *StopBFS* ne rencontre aucun sommet non marqué.

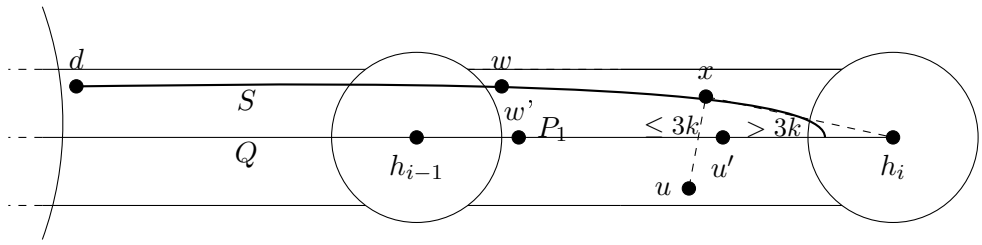

FIGURE 4.18 – Notations utilisées dans la première partie de la preuve du Lemme 23

Type b) En suivant la séquence de laminaires et hubs incidents commençant à L_1 jusqu'à ce qu'un hub de degré au moins 3 soit croisé, g contient une séquence correspondant au lemme 23, avec H_z de degré au moins 3.

Si h_{z+1} n'est pas dominé, f est soit hors de L_{z+1} ou domine h_{z+1} par le lemme 23. Sinon, h_{z+1} est dominé et le dernier sommet non coloré v de P_{z+1} est à distance au plus $6k$ de f, là encore par le lemme 23, et est à distance au plus $R + (K + 2r + k)$ de h_{z+1}. Dans tous les cas, f est à distance au moins $\ell - R - K - 2r - 7k > \frac{\ell}{2} - R$ de h.

En conséquence, le lemme 14 implique qu'un triplet (u, v, w) satisfaisant sa condition existe.

Type c) Si $z = 1$, le lemme 22 garantit que f domine h_1. De plus, $d_G(h_0, h_1) = \ell$ garantit que f n'est pas marqué.

Si $z \geq 2$, le lemme 23 implique que f est dans $B(h_z, 6k)$. Comme $6k \leq K + 2r + k$, f domine h_z. $d_G(h_0, h_z) \geq \ell$ implique alors que f n'est pas marqué.

Type d) Soit v le dernier sommet de P_z appartenant à g. Comme L_z est incident à $B(h_z, r)$, $h_z \neq h_0$, le sommet suivant dans P_{z+1} n'est pas dans g car il appartient à $B(a', R)$, $a' \neq a$. En parcourant g, le *StopBFS* rencontre ainsi un sommet voisin de $B(a', R)$ et s'arrête donc avec f un tel sommet. De plus $d(a, f) \geq d(a, a') - d(a', f) \geq d(h_0, h') - d(h_0, a) - d(h', a') - d(a'f) \geq \ell - 2(K + 2r + k) > 6k$. f est donc non coloré.

Type e) Soit v le dernier sommet de P_z appartenant à g. Comme L_z est incident à $B(h_0, r)$, le sommet suivant de P_{z+1} n'est pas dans g puisqu'il appartient à $B(a, R)$. Le lemme 23 implique de plus que f appartient à $B(v, 6k)$. Finalement, $d_G(a, f) \leq R + 1 + 6k$, donc f est un sommet marqué.

4.3.6.4 Preuve du lemme 17

Soit Q, le chemin calculé par le BFS de s à x. Supposons que Q intersecte un hub de degré au moins 3 et de centre h. Soit y un sommet dans cette intersection. Supposons de plus y à distance plus de $4K + 3k + 2 + 2r$ de s et x. Ces derniers sont alors à distance plus de $4K + 3k + 2 + r$ de h, donc hors de $B(h, 4K + 3k + 2 + r)$. Aucun sommet de G n'étant coloré, le BFS partant de s contient G et a fortiori

$B(h, 4K + 3k + 2 + r)$. Toutes les conditions du Lemme 14 sont vérifiée, un triplet de sommets (u, v, w) est détecté.

Supposons maintenant qu'aucun triplet tel que défini dans le Lemme 14 n'est détecté. Par le paragraphe précèdent, aucun sommet à distance plus de $4K + 3k + 2 + 2r$ de s et x n'appartient à un hub de degré 3 ou plus. Le chemin $Q_{s+4K+3k+2+2r,x-4K+3k+2+2r}$ est donc dans un unique laminaire ou dans une séquence alternant laminaires et hubs de degré 2.

Si s est dans un laminaire, considérons h un centre de hub qui n'est pas une extrémité du laminaire. Si s est dans un hub, considérons h un centre de hub différent de celui de s. Dans tous les cas un chemin de s à h traverse ou commence dans un hub H' avec pour centre de hub $h' \neq h$ et nous avons,

$$d(s, h) \geq d(h, h') - 2r \geq \ell - 2r$$

Le chemin Q est donc de taille au moins $\ell - 2r$. Soit m le milieu de Q. Par les remarques précédentes m est à distance au moins $\frac{\ell}{2} - 4K - 3k - 2r - 2$ d'un hub de degré 3 ou plus. Le sommet m est donc au centre d'une séquence $S = H_0, L_0, ...H_i, L_i, ...H_z$ telle que tous les hubs de S sont de degré 2 sauf les extrémités.. G étant connexe et contenant un hub de degré 3 ou plus, au moins l'un des hubs à une extrémité de la séquence est de degré 3 ou plus. Notons que l'on a potentiellement $H_0 = H_z$.

Si m est dans un laminaire alors $B(m, K)$ déconnecte S par le lemme 1. Si m est dans un hub H, alors $B(m, R)$ contient H et déconnecte S. Soit d le sommet de Q à distance $R + 1$ de m le plus proche de s. Nous avons d à distance au moins $\frac{\ell}{2} - 4K - 3k - R - 2r - 3$ d'un hub de degré 3 ou plus. De plus d est dans une séquence $S' = H_0, L_0, ...H_i, L_i, ...B(m, R)$ telle que H_0 est de degré 1 ou alors H_0 est de degré supérieur à 2 et déconnecte $g = G \setminus B(m, R)$.

Dans ce second cas, par le lemme 21, le BFS dans g partant de d et atteignant f détecte un triplet de sommets (u, v, w) tel que défini dans le Lemme 14.

Il ne nous reste donc plus qu'à prouver que H_0 ne correspond par à un hub de degré 1. Supposons par l'absurde H_0 de degré 1. Le sommet s est alors dans la séquence $S = H_0, L_0, ...H_i, L_i, ...H_z$ avec H_z de degré au moins 3 qui déconnecte S du reste de G. Si x est dans S, alors $B(x, R)$ déconnecte s de $G \setminus S$. Soit h un hub hors de S,

$$d(s, h) \geq d(s, x) - 2R + d(x, h) \geq d(s, x) - 2R + d(h_z, h) - 2r$$

$$\geq d(s, x) + \ell - 2(R + r) > d(s, x)$$

Contredisant le fait que x soit un sommet à distance maximale de s. Si x n'est pas dans S, il est tout de même à distance au plus $4K + 3k + 2 + 2r$ de h_z.

$$d(s, h) \geq d(s, h_z) + d(h_z, h') - 2r \geq d(s, h_z) + \ell - 2r$$

$$d(s, x) \leq d(s, h_z) + d(h_z, x) + 2r \leq d(s, h_z) + 4K + 3k + 2 + 4r \leq d(s, h_z) + \ell - 2r$$

Contredisant à nouveau le fait que x soit un sommet à distance maximale de s.

4.3.6.5 Preuve du lemme 18

Les deux premières propriétés sont vérifiées au début de l'algorithme. En effet, l'ensemble A retourné par $TrouverHubs$ est H-dominant. Ainsi, tous les sommets de degré 1 ou 3 sont dominés et seules les composantes de Type a), d) ou e) sont présentes dans $G \setminus B(A, R)$. De plus, toute composante de Type a) est contenue dans un ensemble $B(a, R+2K+1)$, $a \in A$, en conséquence du lemme 22. Comme $\frac{\ell}{2} > R$, tout sommet x au milieu d'un chemin laminaire contenant une composante de Type a) est supprimable. De plus $d(a, x) \geq d(h, x) - d(a, h) \geq \lfloor \frac{\ell}{2} \rfloor - (K + 2r + k) > R + 2K + 1$, ainsi x appartient à une composante de type d) ou e).

Ces deux premières affirmations étant vérifiées, elles le restent si les deux dernières propriétés restent vraies à chaque itération.

De plus, à chaque itération, le nombre de sommets supprimables décroit strictement jusqu'à ce qu'il n'y ait plus de composante de Type d) ou e), l'algorithme s'arrête avec plus aucun sommet supprimable. Il est donc bien suffisant de montrer que les deux dernières affirmations sont vérifiées en supposant les deux premières.

Soit une itération de la boucle **Pour** qui supprime des sommets. Comme $TrouverHubs$ n'ajoute des sommets dans B que au milieu de composantes de Type e), b appartient à une telle composante.

Tous les sommets de cette composante, ainsi que les sommets de composantes de Type a) incluses dans L_1 ou L_z, sont K-dominés par Q_1 ou Q_2. En effet, si ce n'est pas le cas, un sommet non coloré c est trouvé et est ajouté à A à la ligne 15, et aucun sommet n'est supprimé lors de cette itération.

Pour montrer qu'aucun sommet d'une autre composante n'est supprimé, nous devons montrer que Q_1 (et par symétrie Q_2) ne K-couvre pas un sommet d'un laminaire L incident à $B(a, R)$ et différent de L_1. Soit h le sommet de H dominé par a, et soit L un laminaire incident à $B(h, r)$ différent de L_1.

Les sommets supprimables de L étant à distance au moins $R - (K + 2r + k) \geq r + 3k + K$ de h, aucun n'est supprimé si Q_1 ne traverse pas L.

Supposons donc qu'il existe des sommets x, y et z dans cet ordre dans Q_1, tels que x et z soient dans $L \cap B(h, r)$ et y soit le sommet de Q_1 le plus éloigné de h. Soit x', y' et z' des sommets du chemin laminaire les k-dominant. Si y' est plus proche de h que x' ou z', disons x',

$$d(h_1, y) \leq d(h_1, x') + d(y', y) \leq d(h_1, x) + d(x, x') + d(y', y) \leq r + 2k$$

Sinon, le lemme 20 implique que $d(x, y) \leq 3k$. Dans tous les cas $d(h, y) \leq r + 3k$. En conséquence, tout sommet K-dominé par Q_1 est à distance au plus $r + 3k + K$ de h, et est donc non-supprimable. Il en est de même pour Q_2 par symétrie. Aucun des sommets supprimables de L n'est donc supprimé.

Soit maintenant une itération de la boucle **Tant que** qui supprime des sommets. Comme toutes les composantes de Type e) contiennent des sommets $b \in B$, ils sont supprimés lors de la boucle **Pour**. Lors d'une itération de la boucle **Tant que**, il ne reste donc que des composantes de Type a) ou d).

Tout chemin construit Q liant $B(a, R)$ à $B(a', R)$, $a' \leq a$, intersecte une composante de Type d). Tous les sommets de cette composante, ainsi que les sommets des composantes de Type a) inclus dans L_1 ou L_z sont K-dominés par Q. En effet, dans l'hypothèse inverse, soit w un sommet d'une telle composante et à distance $K+1$ de Q. Soit u sommet de Q tel que $d(u, w) = K+1$. Comme $R \geq 4K+2$ et $w \notin B(a, R)$, u appartient à $r_{3K}(Q)$. Un triplet satisfaisant le lemme 14 est donc détecté et aucun sommet n'est supprimé.

Le fait qu'aucun sommet d'une autre composante n'est supprimé lors d'une itération de boucle **Tant que** est montré de façon analogue à la preuve pour la boucle **Pour**.

4.4 Résultats empiriques

4.4.1 Graphes aléatoires

Dans cette section nous confrontons l'algorithme à des graphes générés aléatoirement afin de comparer les résultats empiriques aux bornes théoriques établies en 4.3.

4.4.1.1 Génération de graphes aléatoires

Afin d'évaluer les performances de notre algorithme, nous devons générer des graphes divers possédant une décomposition (r, k)-hub-laminaire connue. Pour ceci, nous avons développé une procédure de génération de graphes hub-laminaires.

La première étape consiste en la génération d'un graphe aléatoire G_0 de n sommets et possédant un diamètre large.

Tout sommet v_i de ce graphe correspond à un point choisi aléatoirement sur la grille d'entiers $[0 : n] \times [0 : n]$. L'arête (v_i, v_j) est ajoutée avec une probabilité inverse à leur distance sur la grille. Au dessus d'un certain seuil la probabilité est nulle, précisément pour un seuil d_{max} :

$$\mathbb{P}((v_i, v_j) \in E(G_0)) = \begin{cases} 1 - \frac{\|v_i, v_j\|_1}{d_{max}} & \text{si } \|v_i - v_j\|_1 \leq d_{max} \\ 0 & \text{sinon} \end{cases}$$

La valeur de d_{max} est choisie empiriquement de manière à garantir un degré moyen suffisamment grand tout en conservant un diamètre large.

Étant donné r, k, $|H|$ et $|P|$, on construit un sous graphe G de G_0 possédant une décomposition (r,k)-hub-laminaire (H, P) de la manière suivante :

1. *$|H|$ sommets éloignés les uns des autres sont sélectionnés de manière gloutonne et ajoutés à $V(G)$ et H.*

   Le premier sommet est sélectionné aléatoirement et les suivants sont ajoutés un à un parmi les plus éloignés des sommets déjà choisis. Tous les sommets à distance au plus r de H sont ajoutés à $V(G)$.

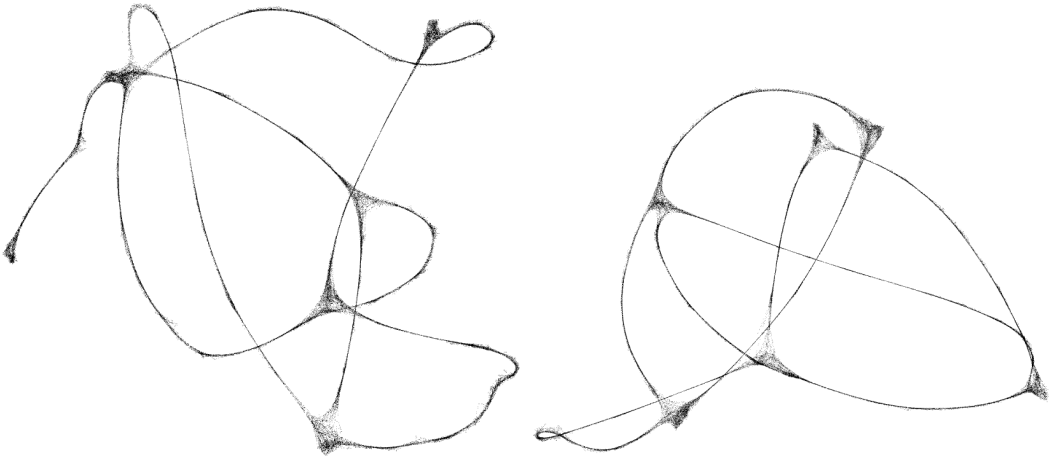

FIGURE 4.19 – Exemples de deux graphes générés par notre procédure avec les mêmes paramètres : $k = 1$, $r = 2$, $nombreDeHubs = 6$, $nombreDeLaminaires = 8$ et $espacementMaxHubs = 17$. La décomposition du premier contient un hub de degré 1, un de de degré 2, deux de degré 3, un de degré 4 et un de degré 5. Celle du second, deux hub de degré 2, deux hubs de degré 3 et deux de degré 4.

2. $|H| - 1$ *plus courts chemins reliant les sommets de* H *sont sélectionnés de manière gloutonne et ajoutés à* $V(G)$ *et* P.

   Un plus court chemin reliant les deux sommets les plus proches de H est ajouté à G. On ajoute ensuite itérativement un plus court chemin entre un sommet connecté et un non connecté de H. A chaque ajout de chemin, les sommets simultanément à distance au plus k du chemin et à distance au moins $r + 1$ de H sont ajoutés à $V(G)$ et supprimés de G_0. Cette étape garantit que tous les sommets de H sont dans la même composante connexe de G.

3. $|P|-(|H|-1)$ *autres plus courts chemins entre des sommets de* H *sont ajoutés à* G *et* P.

   On ajoute itérativement à G un plus court chemin entre les deux sommets de H les plus proches et non déjà directement connectés par un plus court chemin à cette étape ou la précédente. A chaque ajout de chemin, les sommets simultanément à distance au plus k du chemin et à distance au moins $r + 1$ de H sont ajoutés à $V(G)$ et supprimés de G_0.

4. *On supprime chaque arête* (u, v) *telle que* u *ou* v *soit à distance plus de* r *d'un sommet de* H *et telle que le chemin de* P *le plus proche de* u *diffère de celui le plus proche de* v.

Une légère variante peut être ajoutée à cet algorithme. La taille maximale des laminaires ℓ est alors indiquée en entrée de l'algorithme. A l'étape 1, les nouveaux hubs ne sont plus sélectionnés à distance maximale mais parmi les sommets à distance ℓ des hubs déjà existants.

Nous n'avons pas de résultats concernant le support des graphes ainsi générés

(tout graphe (r, k)-laminaire peut-il être généré ainsi ?), ni sur la loi liée à cette génération. Cette procédure nous permet tout de même de générer de nombreux graphes (r, k)-hub-laminaires qui nous permettent d'évaluer les performances de notre algorithme.

4.4.1.2 Évaluation de l'algorithme

Les performances de l'algorithme dépendant fortement du choix de R et K en entrée, des valeurs trop faibles entraineront qu'aucune décomposition ne sera trouvée. A l'inverse si elles sont élevées nous avons le risque de calculer une décomposition de pertinence limitée. Par exemple, en choisissant R de la taille du diamètre du graphe, l'algorithme retournera un unique sommet faisant office de hub. Intuitivement nous cherchons les valeurs de K et R les plus faibles telles qu'une décomposition valide soit trouvée. Cependant, l'optimisation devant être effectuée sur deux paramètres, il peut y avoir plusieurs paires (R, K) minimales, c'est à dire qui retournent une décomposition valide sans que $(R-1, K)$ et $(R, K-1)$ n'en renvoient aucune.

Dans la suite de ce chapitre, nous choisirons parmi les décompositions proposées par l'algorithme celle minimisant $max(4K, 2R)$. Nous faisons ce choix car de telles valeurs minimisent l'erreur maximale du label de distances comme montré dans le chapitre suivant par la Proposition 5.

Le premier paramètre que nous allons tester est la taille du chemin laminaire nécessaire pour calculer une décomposition. La borne supérieure théorique montrée par le Théorème 8 est $\ell \geq 8r + 60k + 4$, ce qui peut sembler assez élevé en termes de facteurs multiplicatifs de r et k. Nous allons donc tester sur différents graphes générés comme en section 4.4.1.1 la valeur de ℓ telle qu'une décomposition soit calculée par l'algorithme. Pour ceci nous allons lors de la sélection de hubs de nos graphes aléatoires forcer leur espacement à ℓ au maximum. Nous fixons de plus un des paramètres, r ou k, afin d'observer l'effet du second sur la taille requise ℓ. La figure 4.20 montre que pour k fixé à 2, la valeur minimale de ℓ requise pour calculer une décomposition est en moyenne de $3r+6$. De plus ℓ semble croitre très lentement avec k, r étant fixé à 8 dans la seconde série de simulations. Ceci dit, dans tous les cas, la variance est très importante, sans doute à cause de la diversité des graphes considérés.

Ces expériences confirment qu'une valeur de ℓ suffisamment importante est nécessaire pour détecter une décomposition hub-laminaire mais que les bornes théoriques pourraient être affinées, au moins pour certaines familles de graphes.

Dans un second temps nous calculons pour k fixé et r variable, la valeur de R retournée par notre algorithme. Nous faisons également l'expérience inverse, c'est à dire pour r fixé, la valeur de K en fonction de k.

Les résultats présentés par la figure 4.21 montrent, comme attendu, une corrélation linéaire entre r et R. Une régression linéaire nous donne une relation $E(R) = 1.6r + 3$. De plus, le pire cas dans nos simulations est $R = 2.8r$, ce qui est significativement meilleur que la borne théorique de $R = 3r + 9k$. Quant à r fixé,

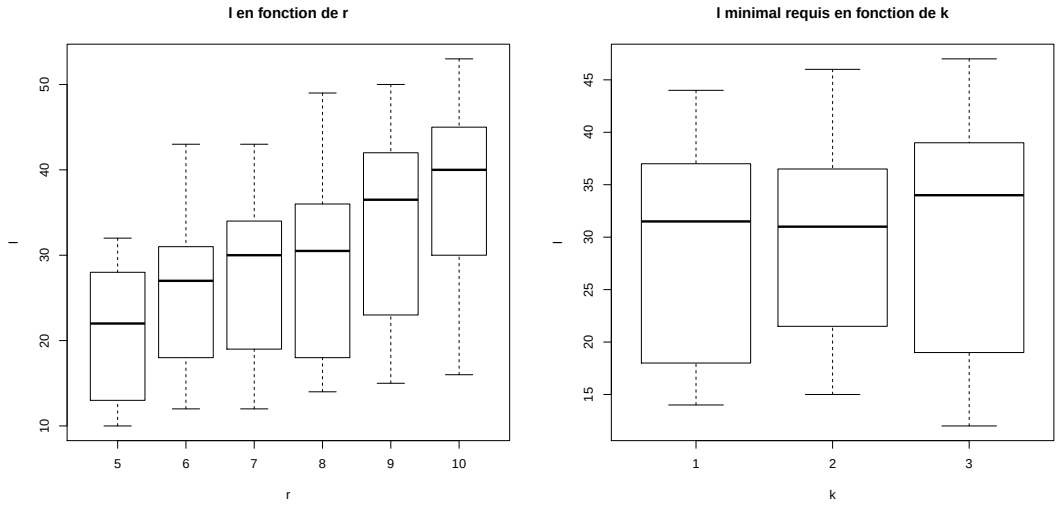

FIGURE 4.20 – Taille du chemin laminaire nécessaire pour calculer une décomposition. Environ 3000 graphes sont générés par expérience.

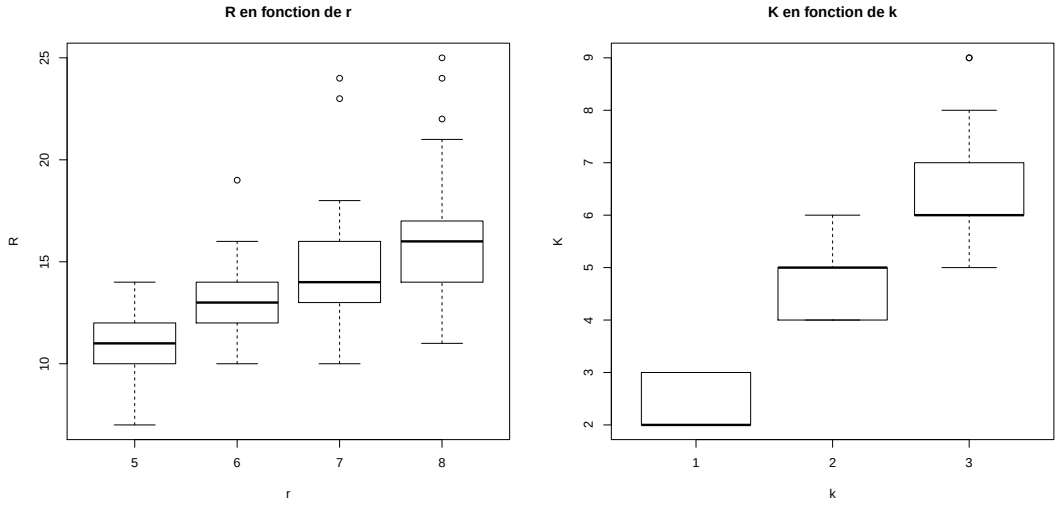

FIGURE 4.21 – Valeurs de R et K nécessaires pour calculer une décomposition en fonction de r et k. Environ 1000 graphes sont générés par expérience.

nous obtenons une relation $K = 2.23k + 0.2$ avec pour pire cas $K = 3k$, il semble donc que la borne théorique sur K est plus fine en pratique que celle sur R.

4.4.2 Graphes de reads

Nous avons appliqué notre algorithme aux graphes de reads afin d'en extraire les données importantes. Après concertation avec les chercheurs en biologie à l'origine du projet, nous renvoyons pour chaque graphe où un hub de degré au moins 3 est

détecté :

— Les valeurs de K et R.

— Le nombre de hubs de degré au moins 3.

— Le degré et la liste des sommets des hubs de degré au moins 3.

Précisément, pour traiter l'ensemble des graphes d'un lézard, nous procédons comme suit. Toute composante connexe dont la taille du fichier est inférieure à $1Ko$ n'est pas traité, cela correspond à un graphe de moins de 30 sommets. Pour tout fichier de taille supérieure, une décomposition est cherchée pour k variant de 1 à 6 et r de $2k$ à 12. Ces valeurs ont été déterminées empiriquement et semblent suffire à circonscrire l'ensemble des décompositions possibles pour nos données. Nous avons testé des valeurs plus larges de r et k sans pour autant déterminer de meilleures décompositions. De même nous avons parfois traité les fichiers de moins de $1Ko$ sans trouver plus de décompositions. Ceci permet de passer la durée du traitement de l'ensemble des graphes d'un lézard de un ou deux jours à une ou deux heures. En effet si le graphe d'un lézard contient en moyenne 500000 composantes connexes, la majorité de ces composantes sont de tailles très faibles.

Par exemple le graphe de read du lézard $PSMF9$ (PSM : omnivore, F : femelle) contient 680079 composantes connexes dont seulement 8306 contenant plus de 30 sommets et 1216 contenant plus de 200 sommets. Finalement 43 composantes connexes du lézard $PSMF9$ sont détectées comme possédant une décomposition hub-laminaire avec au moins un hub de degré 3 ou plus. Parmi celles-ci, une seule possède plusieurs hubs de degré 3 ou plus, cette composante est représentée en figure 4.22. On observe de plus, 32 hubs de degré 3, 11 de degré 4 et 1 de degré 5 (figure 4.23).

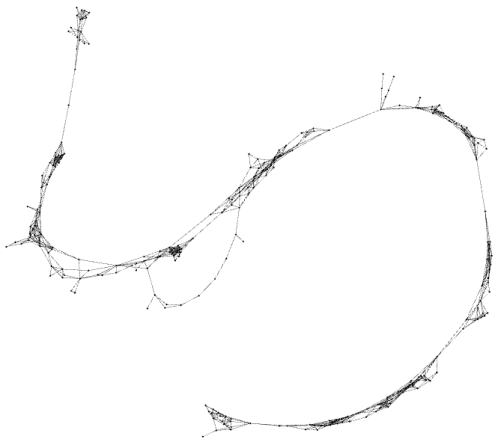

FIGURE 4.22 – Deux hubs de degré 3 formant un cycle.

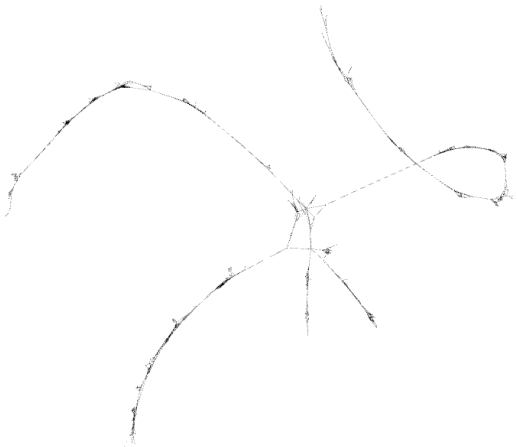

FIGURE 4.23 – Un hub de degré 5.

Les décompositions intéressantes doivent ensuite être étudiées manuellement par les biologistes. La séquence codante présente à l'intérieur des hubs de degré 3 ou plus, doit être comparée aux bases de données génomiques pour en comprendre le rôle. Ce travail d'analyse biologique n'a pas été effectué au moment de la rédaction

de cette thèse et ne peut donc être présenté ici.

L'ensemble du code utilisé dans cette thèse est disponible sur ma page personnelle à http://www.math-info.univ-paris5.fr/~lplanche/. Un grand merci à Finn Völkel pour m'avoir fourni au début de cette thèse le code de son travail déjà effectué sur ces graphes.

Chapitre 5

Labels de distances et plongement de graphes

Sommaire

5.1 Introduction

Une des raisons principales de l'étude du problème MESP par [Völkel 2016] est celle du plongement de graphe dans une ligne.

Nous montrons que le calcul du cycle isométrique d'excentricité minimale permet le plongement d'un graphe dans un cycle avec une distorsion multiplicative faible. En d'autres termes, les distances dans le cycle sont majorées par les distances dans le graphe multipliées par un facteur constant. Notons que le plongement dans un cycle est majoré par celui dans une ligne puisque cette dernière peut être étendue à un cycle de façon isométrique. A l'inverse, la distorsion après plongement dans une ligne peut quant à elle être bien pire que celle dans un cycle.

Comme indiqué en introduction, un problème proche de celui du plongement de graphe est la représentation compacte des distances dans un graphe. Nous montrons dans ce chapitre que la décomposition hub-laminaire permet une représentation compacte des distances avec une distorsion additive bornée.

5.2 Plongement dans le cercle

Un plongement d'un graphe G dans un cycle est une bijection $f : V(G) \to C$ où C est un cercle d'une taille donnée c. Le plongement est de distorsion γ si $d_G(u,v) \leq d_C(f(u), f(v)) \leq \gamma d_G(u,v)$ pour tous u, v sommets de G. On note $cd(G)$ la *distorsion de cercle* correspondant à la distorsion minimale d'un plongement de G dans un cercle.

Proposition 2. *Si un graphe G a une distorsion de cercle γ, il est possible de calculer un plongement de G dans un cercle de distorsion $O(\gamma^2)$ en temps polynomial.*

Cette proposition découle du Théorème 6 montré au chapitre 3 ainsi que des propositions 3 et 4 qui suivent

Proposition 3. *Tout graphe G possédant un plongement dans un cercle de distorsion γ a un plus court chemin ou un cycle isométrique d'excentricité au plus $\lfloor \gamma/2 \rfloor$.*

Démonstration. Soit un plongement de G dans un cercle C de distorsion γ. Supposons que tout plus court chemin de G a une excentricité plus grande que $\lfloor \gamma/2 \rfloor$.

Montrons dans un premier temps que G possède un cycle simple qui $\lfloor \gamma/2 \rfloor$-domine le graphe. Soit un chemin P donné. Deux sommets consécutifs u et v de P sont à distance au plus γ dans le plongement. P $\lfloor \gamma/2 \rfloor$-domine donc tout sommet plongé entre u et v dans le cercle. Nous définissons l'arc P_C de P dans C comme l'arc de C le plus court contenant le plongement des sommets de P. Notons que tous les sommets plongés dans P_C sont $\lfloor \gamma/2 \rfloor$-dominés par P. Considérons un plus court chemin P tel que l'arc P_C soit de longueur maximale et soient a et b les extrémités de P_C. Si P ne $\lfloor \gamma/2 \rfloor$-domine pas G, considérons un sommet c à distance plus de $\lfloor \gamma/2 \rfloor$ de P. Considérons un plus court chemin Q de c à a dans G. La définition de P implique que Q_C ne contient pas P_C. Le chemin Q_C $\lfloor \gamma/2 \rfloor$-domine donc les sommets plongés dans l'arc C_{ca} – par souci de clarté nous confondons un ici un sommet son plongement dans le cercle – de C qui évitent l'intérieur de P_C. De façon similaire, le plus court chemin R de c à b domine les sommets plongés dans l'arc C_{cb} de C qui évite l'intérieur de P_C. Soit a' le premier sommet de Q dans P. Soit Q' le sous-chemin de Q de c à a' et soit P' le sous-chemin de P de a' à b. Notons que l'arc de $Q' \cup P'$ contient l'arc C_{cb} dans $Q_C \cup P_C$. De façon analogue, soit b' le premier sommet de R dans $Q' \cup P'$. Définissons R' comme le sous chemin de R de c à b' et Q'' le sous chemin de $Q' \cup P'$ de c à b'. Notons que R'_C contient l'arc de c à b qui n'est pas dans $R_C \cup P_C$. Comme $Q''_C \cup R'_C = C$, l'union $Q'' \cup R'$ définit un cycle simple qui $\lfloor \gamma/2 \rfloor$-domine G.

Considérons maintenant un cycle simple S de G qui $\lfloor \gamma/2 \rfloor$-domine G et de taille minimale. S est isométrique car il y aurait sinon un chemin P de a à b dans G qui serait plus court que les deux chemins Q et R de S reliant a à b. Considérons l'arc A de C de a à b inclus dans P_C. Sans perte de généralité, Q domine les sommets plongés dans l'autre partie $C \setminus A$ du cercle. Nous pouvons alors construire à partir de $P \cup Q$ (comme précédemment) un cycle simple qui $\lfloor \gamma/2 \rfloor$-domine G, en contradiction avec la minimalité de la longueur de S. □

Proposition 4. *Soit un graphe G et un cycle isométrique d'excentricité k de G, un plongement de G dans un cycle de distorsion $O(k \cdot cd(G))$ peut être calculé en temps polynomial.*

Démonstration. La construction du plongement est similaire à celle proposée par [Dragan 2017] avec des cycles eulériens et des arbres de profondeur k. Cependant nos arbres ont leurs racines dans un cycle et non dans une ligne.

Soit un cycle isométrique C de G et d'excentricité k. Nous construisons une forêt F ayant les racines dans C comme une union de plus courts chemins : pour tout

sommet $u \in V(G)$ nous sélectionnons un sommet u' tel que $d(u,u') = d(u,C)$ et ajoutons à F un plus court chemin de u à u' ($u' = u$ pour $u \in C$). Pour tout arbre T de F ayant pour racine $c \in C$, nous construisons un cycle eulérien E_c qui est une séquence des sommets de l'arbre commençant par c, visitant tous les sommets de T suivant un tour d'Euler et finissant sur c. Toute arête est utilisée deux fois et la longueur de E_c est $2(|T|-1)$. Nous obtenons ainsi un cycle eulérien sur tout notre graphe en considérant la séquence $E_C = E_{c_1}, c_1c_2, E_{c_2}, \ldots, c_{p-1}c_p, E_{c_p}, c_pc_1$ où p est la taille de C et $c_1, \ldots, c_p$ sont les sommets de C suivant leur ordre dans le cycle. Notons que ce cycle contient $2n$ arêtes au plus et peut se plonger dans un cercle C' de même taille.

Nous étudions maintenant la distorsion de ce plongement dans C'. Soit une arête uv de G, notons u' et v' les racines de respectivement u et v. Soit S l'union des arbres enracinés sur le plus court chemin de u' à v' dans C. Notons que la distance de u à v dans le cycle E_C est au plus deux fois la taille de S. Afin de majorer $|S|$, nous considérons un plongement de G dans le cycle C_{opt} de distorsion $\gamma = cd(G)$. Comme nous avons $d(u',v') \leq 2k+1$, le diamètre de S est au plus $4k+1$. Deux sommets de S sont donc à une distance au plus $\gamma(4k+1)$ dans le cycle C_{opt}. Nous avons donc $|S| \leq 2\gamma(4k+1)$, et notre plongement C' a une distorsion $O(\gamma k)$.

□

5.3 Labels de distances

La décomposition hub-laminaire d'un graphe G nous permet de calculer une représentation compacte des distances de G de distorsion additive en assignant des labels à chaque sommet.

Une *labellisation des distances* d'un graphe G consiste en l'assignation d'un label L_u à chaque sommet $u \in V(G)$ et en une fonction d'estimation des distances f qui renvoie une approximation de $d(u,v)$ à partir de L_u et L_v.

On dit que la distorsion est additive s'il existe α tel que $d(u,v) \leq f(L_u, L_v) \leq d(u,v) + \alpha$ pour tout u, v dans G. Comme il existe un compromis entre la distorsion et la taille des labels utilisés, cette dernière est également un paramètre important d'un label des distances.

Proposition 5. *Soit une (r,k)-hub-laminaire décomposition $(H, \mathcal{P})$ avec λ laminaires d'un graphe G. Une labellisation de distorsion additive* $\max(4k, 2r)$ *et de labels de $O(\lambda \log n)$ bits peut être calculée en temps polynomial.*

Démonstration. Supposons les centres de hub numérotés de 1 à q, $q \leq 2\lambda$ et les laminaires numérotés de 1 à λ. Pour tout $u \in V(G)$, nous définissons un label de hub H_u qui consiste en l'intégralité des paires $(h, d(u,h))$ pour $h \in H$. Pour tout sommet u dans un hub, c.a.d. tel qu'il existe $h \in H$ à distance au plus r de u, son label est son label de hub, c.a.d. $L_u := H_u$. Pour un sommet u dans un laminaire α, il existe par définition $P \in \mathcal{P}$ d'extrémité $h_1 < h_2$ tel que $u \in B(P,k) \setminus B(\{h_1, h_2\}, r)$. On inscrit dans le label de u son label de hub ainsi que $(d_P(h_1, u'), d(u', u), \alpha)$ avec

u' un sommet quelconque de P à distance au plus k de u. C'est à dire, $L_u := (d_P(h_1, u'), d(u', u), \alpha), H_u$.

La distance $d(u, v)$ entre deux sommets $u, v \in V(G)$ est alors estimée à partir de leurs labels L_u et L_v de la façon suivante. Nous calculons d'abord une estimation de leur distance à partir des centres de hubs :

$$g(u, v) = \min_{h \in H} d(u, h) + d(v, h)$$

Si L_u et L_v commencent tous deux avec des triplets $(d(h_1, u'), d(u', u), \alpha)$ et $(d(h'_1, v'), d(v', v), \alpha')$ avec $\alpha = \alpha'$, nous détectons que u et v sont dans le même laminaire. On renvoie alors l'estimation de distance $f(u, v) = \min(g(u, v), g'(u, v))$ avec :

$$g'(u, v) = d(u', u) + |d_P(h_1, u') - d_P(h_1, v')| + d(v', v)$$

Sinon, nous retournons simplement $f(u, v) = g(u, v)$ comme estimation de la distance.

Montrons maintenant que $d(u, v) \leq f(u, v) \leq d(u, v) + \max(4k, 2r)$. Par l'inégalité triangulaire nous avons $d(u, v) \leq d(u, h) + d(v, h)$ pour tout $h \in H$ et nous obtenons $d(u, v) \leq g(u, v)$. Dans le cas où u et v appartiennent tous deux au même laminaire $B(P, k)$, notons que $g'(u, v)$ est la longueur du chemin les reliant et passant par $u', v' \in P$, donc $g'(u, v) \leq d(u, v)$. Nous avons donc $d(u, v) \leq f(u, v)$ dans tous les cas. Considérons maintenant un plus court chemin Q de u à v. Supposons dans un premier temps que Q intersecte un hub : il existe $h \in H$ tel que $Q \cap B(h, r) \neq \emptyset$. Soit $x \in Q \cap B(h, r)$. Nous avons alors $d(u, v) = d(u, x) + d(x, v) \leq d(u, h) + d(h, x) + d(v, h) + d(h, x) \leq d(u, h) + d(v, h) + 2r$ ce qui implique $g(u, v) \leq d(u, v) + 2r$. Supposons maintenant que Q n'intersecte aucun hub, le chemin est donc inclus dans un laminaire suivant la définition 8 d'une décomposition hub-lamaire et les axiomes 2 et 4. Soit $P \in \mathcal{P}$ avec pour extrémités $h_1 < h_2$ telles que $Q \subseteq B(P, k) \setminus B(\{h_1, h_2\}, r)$. Alors u et v appartiennent tous deux au laminaire et leurs labels contiennent des triplets $(d(h_1, u'), d(u', u), \alpha)$ et $(d(h'_1, v'), d(v', v), \alpha')$. Considérons le sous-graphe induit $G_P = B(P, k)$. Par l'inégalité triangulaire, nous avons $d_{G_P}(u', v') \leq d_{G_P}(u, u') + d_{G_P}(u, v) + d_{G_P}(v, v')$. Comme Q est inclus dans G_P, nous avons $d(u, v) = d_{G_P}(u, v)$ et nous obtenons $|d_P(h_1, u') - d_P(h_1, v')| = d_{G_P}(u', v') \leq d(u, v) + 2k$, il découle $f(u, v) \leq g'(u, v) \leq d(u, v) + 4k$. Dans tous les cas, nous avons $f(u, v) \leq d(u, v) + \max(4k, 2r)$.

□

De cette Proposition et du Théorème 8 montré au chapitre 4 il découle :

Proposition 6. *Soit G un graphe possédant une décomposition (r, k)-hub-laminaire avec λ laminaires. Il est possible de calculer en temps polynomial une labellisation des distances de distorsion additive $O(r)$ avec des labels de $O(\lambda \log n)$ bits.*

5.4 Simulations

Nous reprenons la génération de graphes aléatoires développée au chapitre 4 afin de tester l'efficacité empirique des labels de distances. Nous comparons l'écart moyen entre la distance entre deux sommets et leur distance calculée avec les labels. Les résultats de la comparaison sont présentés à la figure 5.1, l'écart maximal théorique étant de $max(4K, 2R)$ comme montré en proposition 5.

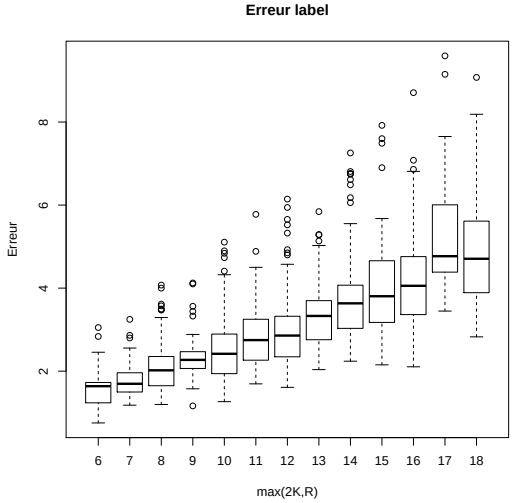

FIGURE 5.1 – Erreur moyenne sur les distances calculées avec les labels en fonction de $max(2K, R)$

Nous avons une corrélation linéaire entre l'erreur sur les distances et $max(4K, 2R)$. En moyenne celle-ci est de $0.14 \times max(4K, 2R)$ et le pire cas pour un graphe obtenu dans nos simulations correspond à un écart moyen de de $0.26 \times max(4K, 2R)$.

Chapitre 6

Conclusion et perspectives

L'objet principal de cette thèse fut l'étude des graphes de reads. Nous pensons que le modèle hub-laminaire que nous proposons permet une description efficace de ces données. Les graphes pouvant avoir un intérêt biologique important ont pu être détectés et transmis aux chercheurs afin d'être étudiés. Si ces résultats soulèvent de nouvelles questions d'un point de vue de la structure des graphes, une nouvelle étude sera peut être nécessaire. En plus des intérêts biologiques de ces données, dont nous espérons avoir amélioré la compréhension, ces graphes ont soulevés des questions théoriques importantes. Les chapitres 2, 3 et 5 en étant les objets.

Le premier problème abordé fut celui du plus court chemin d'excentricité minimale, et son analogue, le problème laminaire. Sur ces deux problèmes, les questions pouvant se poser naturellement sont les suivantes :

— L'existence de meilleurs algorithmes d'approximation.
— L'existence d'algorithmes FPT.
— Le lien entre ces problèmes et d'autres de théorie des graphes.

Nous avons déjà abordé le premier point lors du chapitre en question. A cause du lemme 1, il semble qu'il soit fondamentalement plus difficile de calculer une α-approximation avec $\alpha < 3$ qu'une 3-approximation.

Il serait intéressant de tester ces algorithmes sur données réelles pour comparer les bornes théoriques et pratiques. On sait par exemple que si le double-BFS est une 2-approximation du diamètre, il est en pratique un algorithme bien plus redoutablement efficace.

Pour l'existence d'algorithmes FPT, un paramètre que l'on pourrait avoir envie de fixer et celui de la longueur de chemin. Cependant nous pouvons construire des graphes de longueur de chemin 3 où le calcul d'un plus court chemin d'excentricité minimal est NP-difficile. Fixer la largeur de chemin permet effectivement des algorithmes polynomiaux mais ceci découle directement du fait que la largeur de chemin borne l'excentricité minimale du plus court chemin d'un graphe. Le serpent se mort la queue.

Le problème MESP est directement borné par celui de la largeur de chemin, tout problème lié à l'un l'est donc à l'autre. Un graphe possédant un chemin d'excentricité faible voit son diamètre plus facilement calculable. Très précisément, on peut l'approximer en temps linéaire avec un facteur additif de $6k$. Ceci découle immédiatement de la 3-approximation en temps $O(m)$ présentée au chapitre 2.

Les mêmes questions peuvent naturellement se poser pour le problème MEIC mais seront sans doute d'un intérêt moindre. Les graphes possédant des plus courts

chemins de faible excentricité sont nombreux à être étudiés – on pensera immédiatement aux graphes d'intervalles – mais ceux possédant des cycles isométriques de faibles excentricités le sont moins. Il serait sans doute plus intéressant de mettre les considérations d'excentricité de côté pour se concentrer sur une meilleure compréhension des cycles isométriques en général. Assez peu de recherches semblent avoir été faites à ce sujet, le résultat principal semblant être celui du calcul du plus long cycle isométrique réalisable en temps polynomial [Lokshtanov 2009]. Nous pensons que déterminer si il existe un cycle isométrique passant par deux sommets quelconques est NP-difficile. Les cycles isométriques semblent donc – en termes de complexité algorithmique – des objets difficiles à manier, mais pouvant avoir des applications très concrètes. Ceci en justifierait une étude approfondie.

Le problème de la représentation compacte des distances est extrêmement vaste et possède un champ de recherche immense. Dans le cas particulier de la représentation compacte des distances pour les graphes possédant une décomposition hub-laminaire, les labels que nous proposons permettent une représentation efficace des distances qui semble difficile à améliorer.

La question qui se pose est plutôt de déterminer s'il est possible de transposer cette méthode de labélisation des sommets à d'autres graphes possédant des structures similaires. De même, il serait intéressant de se demander quels types de graphes sont susceptibles de posséder une décomposition hub-laminaire. De tels graphes doivent nécessairement avoir un diamètre important, et sont sans doute à chercher du côté des graphes représentant des objets physiques en deux ou trois dimensions.

Bibliographie

[Aingworth 1999] Donald Aingworth, Chandra Chekuri, Piotr Indyk et Rajeev Motwani. *Fast Estimation of Diameter and Shortest Paths (Without Matrix Multiplication)*. SIAM J. Comput., vol. 28, no. 4, pages 1167–1181, 1999. (Non cité.)

[Alon 1997] Noga Alon, Raphael Yuster et Uri Zwick. *Finding and counting given length cycles*. Algorithmica, vol. 17, no. 3, pages 209–223, 1997. (Cité en page 8.)

[Arnborg 1987] Stefan Arnborg, Derek G. Corneil et Andrzej Proskurowski. *Complexity of Finding Embeddings in a K-tree*. SIAM J. Algebraic Discrete Methods, vol. 8, no. 2, pages 277–284, Avril 1987. (Cité en page 5.)

[Assouad 1979] Patrice Assouad. *Étude d'une dimension métrique liée à la possibilité de plongements dans* $\mathbf{R}^n$. C. R. Acad. Sci. Paris Sér. A-B, vol. 288, no. 15, pages A731–A734, 1979. (Cité en page 6.)

[B. Tenenbaum 2000] Joshua B. Tenenbaum, Vin Silva et John C. Langford. *A Global Geometric Framework for Nonlinear Dimensionality Reduction*. vol. 290, pages 2319–2323, 01 2000. (Cité en page 6.)

[Bacso 2007] G. Bacso, Zs. Tuza et M. Voigt. *Characterization of graphs dominated by induced paths*. Discrete Mathematics, vol. 307, no. 7, pages 822 – 826, 2007. Cycles and Colourings 2003. (Cité en pages 5 et 6.)

[Badoiu 2005a] Mihai Badoiu, Julia Chuzhoy, Piotr Indyk et Anastasios Sidiropoulos. *Low-distortion embeddings of general metrics into the line*. In Proceedings of the thirty-seventh annual ACM symposium on Theory of computing, pages 225–233. ACM, 2005. (Cité en page 7.)

[Badoiu 2005b] Mihai Badoiu, Kedar Dhamdhere, Anupam Gupta, Yuri Rabinovich, Harald Räcke, R. Ravi et Anastasios Sidiropoulos. *Approximation Algorithms for Low-distortion Embeddings into Low-dimensional Spaces*. In Proceedings of the Sixteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '05, pages 119–128, Philadelphia, PA, USA, 2005. Society for Industrial and Applied Mathematics. (Cité en page 7.)

[Bandyopadhyay 2006] Deepak Bandyopadhyay, Jun Huan, Jinze Liu, Jan Prins, Jack Snoeyink, Wei Wang et Alexander Tropsha. *Structure-based function inference using protein family-specific fingerprints*. Protein Science, vol. 15, no. 6, pages 1537–1543, 2006. (Cité en page 9.)

[Birmelé 2016] Etienne Birmelé, Fabien de Montgolfier et Léo Planche. *Minimum Eccentricity Shortest Path Problem : An Approximation Algorithm and Relation with the k-Laminarity Problem*. In Combinatorial Optimization and Applications - 10th International Conference, COCOA 2016, Hong Kong, China, December 16-18, 2016, Proceedings, pages 216–229, 2016. (Cité en pages 10 et 14.)

[Birmelé 2017] Etienne Birmelé, Fabien de Montgolfier, Léo Planche et Laurent Viennot. *Decomposing a Graph into Shortest Paths with Bounded Eccentricity.* In Yoshio Okamoto et Takeshi Tokuyama, editeurs, 28th International Symposium on Algorithms and Computation (ISAAC 2017), volume 92 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 15 :1–15 :13, Dagstuhl, Germany, 2017. Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik. (Cité en pages 10 et 40.)

[Bodlaender 1996] H. Bodlaender. *A Linear-Time Algorithm for Finding Tree-Decompositions of Small Treewidth.* SIAM Journal on Computing, vol. 25, no. 6, pages 1305–1317, 1996. (Cité en page 5.)

[Bodlaender 2008] Hans L Bodlaender et Arie MCA Koster. *Combinatorial optimization on graphs of bounded treewidth.* The Computer Journal, vol. 51, no. 3, pages 255–269, 2008. (Cité en page 5.)

[Compeau 2011] Phillip EC Compeau, Pavel A Pevzner et Glenn Tesler. *How to apply de Bruijn graphs to genome assembly.* Nature biotechnology, vol. 29, no. 11, page 987, 2011. (Cité en page 3.)

[Corneil 1985] Derek G. Corneil, Yehoshua Perl et Lorna K Stewart. *A linear recognition algorithm for cographs.* SIAM Journal on Computing, vol. 14, no. 4, pages 926–934, 1985. (Cité en page 8.)

[Corneil 1995] Derek G. Corneil, Stephan Olariu et Lorna Stewart. *A Linear Time Algorithm to Compute a Dominating Path in an AT-Free Graph.* Inf. Process. Lett., vol. 54, no. 5, pages 253–257, 1995. (Cité en page 6.)

[Corneil 1997] D. Corneil, S. Olariu et L. Stewart. *Asteroidal Triple-Free Graphs.* SIAM Journal on Discrete Mathematics, vol. 10, no. 3, pages 399–430, 1997. (Cité en page 6.)

[Corneil 1999] Derek G. Corneil, Stephan Olariu et Lorna Stewart. *Linear Time Algorithms for Dominating Pairs in Asteroidal Triple-free Graphs.* SIAM J. Comput., vol. 28, no. 4, pages 1284–1297, 1999. (Cité en page 6.)

[Corneil 2003] Derek G. Corneil, Dragan Feodor F. et Kohler Ekkehard. *On the power of BFS to determine a graph's diameter.* Networks, vol. 42, no. 4, pages 209–222, 2003. (Cité en page 14.)

[Cygan 2012] Marek Cygan et Marcin Pilipczuk. *Bandwidth and Distortion Revisited.* Discrete Appl. Math., vol. 160, no. 4-5, pages 494–504, Mars 2012. (Cité en page 7.)

[Deogun 1995] Jitender S Deogun et Dieter Kratsch. *Diametral path graphs.* In International Workshop on Graph-Theoretic Concepts in Computer Science, pages 344–357. Springer, 1995. (Cité en pages 5 et 19.)

[Deogun 2002] Jitender S Deogun et Dieter Kratsch. *Dominating pair graphs.* SIAM Journal on Discrete Mathematics, vol. 15, no. 3, pages 353–366, 2002. (Cité en page 5.)

[Deshpande 2005] Mukund Deshpande, Michihiro Kuramochi, Nikil Wale et George Karypis. *Frequent substructure-based approaches for classifying chemical compounds.* IEEE Transactions on Knowledge and Data Engineering, vol. 17, no. 8, pages 1036–1050, 2005. (Cité en page 9.)

[Dourisboure 2004] Yon Dourisboure et Cyril Gavoille. *Small Diameter Bag Tree-Decompositions*, Mai 2004. Rapport de recherche. (Cité en page 5.)

[Dragan 2017] Feodor F. Dragan et Arne Leitert. *On the minimum eccentricity shortest path problem.* Theoretical Computer Science, vol. 694, pages 66 – 78, 2017. (Cité en pages 10, 13, 14, 26, 76, 89 et 90.)

[Fakcharoenphol 2004] Jittat Fakcharoenphol, Satish Rao et Kunal Talwar. *A tight bound on approximating arbitrary metrics by tree metrics.* Journal of Computer and System Sciences, vol. 69, no. 3, pages 485–497, 2004. (Cité en page 7.)

[Fellows 2009] Michael R. Fellows, Fedor V. Fomin, Daniel Lokshtanov, Elena Losievskaja, Frances A. Rosamond et Saket Saurabh. *Distortion Is Fixed Parameter Tractable.* In Susanne Albers, Alberto Marchetti-Spaccamela, Yossi Matias, Sotiris Nikoletseas et Wolfgang Thomas, editeurs, Automata, Languages and Programming, pages 463–474, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg. (Cité en page 7.)

[Fellows 2013] Michael R. Fellows, Fedor V. Fomin, Daniel Lokshtanov, Elena Losievskaja, Frances A. Rosamond et Saket Saurabh. *Distortion is Fixed Parameter Tractable.* TOCT, vol. 5, no. 4, pages 16 :1–16 :20, 2013. (Cité en page 7.)

[Fomin 2011] Fedor V. Fomin, Daniel Lokshtanov et Saket Saurabh. *An exact algorithm for minimum distortion embedding.* Theoretical Computer Science, vol. 412, no. 29, pages 3530 – 3536, 2011. (Cité en page 7.)

[Gao 2010] Xinbo Gao, Bing Xiao, Dacheng Tao et Xuelong Li. *A survey of graph edit distance.* Pattern Analysis and applications, vol. 13, no. 1, pages 113–129, 2010. (Cité en page 8.)

[Gavoille 2004] Cyril Gavoille, David Peleg, Stéphane Pérennes et Ran Raz. *Distance labeling in graphs.* J. Algorithms, vol. 53, no. 1, pages 85–112, 2004. (Cité en page 8.)

[Gavoille 2005] Cyril Gavoille et Olivier Ly. *Distance Labeling in Hyperbolic Graphs.* In ISAAC 2005, volume 3827 of *Lecture Notes in Computer Science*, pages 1071–1079. Springer, 2005. (Cité en page 8.)

[Grotzsch 1958] H Grotzsch. *Zur Theorie der diskreten Gebilde. VII. Ein Dreifarbensatz far dreikreisfreie Netze auf der Kugel.* Wiss. Z. Martin-Luther-Univ. Halle-Wittenberg. Math. Nat. Reihe8 (1958/59), pages 109–120, 1958. (Cité en page 8.)

[Habib 2005] Michel Habib et Christophe Paul. *A simple linear time algorithm for cograph recognition.* Discrete Applied Mathematics, vol. 145, no. 2, pages 183–197, 2005. (Cité en page 8.)

[Handler 1973] G. Y. Handler. *Minimax Location of a Facility in an Undirected Tree Graph.* Transportation Science, vol. 7, no. 3, pages 287–293, 1973. (Cité en page 14.)

[Heggernes 2008] Pinar Heggernes, Daniel Meister et Andrzej Proskurowski. *Minimum Distortion Embeddings into a Path of Bipartite Permutation and Threshold Graphs.* In Joachim Gudmundsson, editeur, Algorithm Theory – SWAT 2008, pages 331–342, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg. (Cité en page 7.)

[Heggernes 2010] Pinar Heggernes et Daniel Meister. *Hardness and approximation of minimum distortion embeddings.* Information Processing Letters, vol. 110, no. 8, pages 312 – 316, 2010. (Cité en page 7.)

[Herrel 2008] Anthony Herrel, Katleen Huyghe, Bieke Vanhooydonck, Thierry Backeljau, Karin Breugelmans, Irena Grbac, Raoul Van Damme et Duncan J Irschick. *Rapid large-scale evolutionary divergence in morphology and performance associated with exploitation of a different dietary resource.* Proceedings of the National Academy of Sciences, vol. 105, no. 12, pages 4792–4795, 2008. (Cité en page 1.)

[Huan 2004] Jun Huan, Wei Wang, Deepak Bandyopadhyay, Jack Snoeyink, Jan Prins et Alexander Tropsha. *Mining protein family specific residue packing patterns from protein structure graphs.* In Proceedings of the eighth annual international conference on Resaerch in computational molecular biology, pages 308–315. ACM, 2004. (Cité en page 9.)

[Indyk 2001] P. Indyk. *Algorithmic Applications of Low-Distortion Geometric Embeddings.* In Proceedings of the 42Nd IEEE Symposium on Foundations of Computer Science, FOCS '01, pages 10–, Washington, DC, USA, 2001. IEEE Computer Society. (Cité en page 6.)

[Indyk 2004] Piotr Indyk et Jirí Matousek. *Low-Distortion Embeddings of Finite Metric Spaces.* In Handbook of Discrete and Computational Geometry, 2nd Ed., 2004. (Cité en page 6.)

[Leitert 2017] Arne Leitert. *Tree-Breadth of Graphs with Variants and Applications.* PhD thesis, Kent State University, 2017. (Cité en page 5.)

[Lekkerkerker 1962] J. Lekkerkerker C. Boland. *Representation of a finite graph by a set of intervals on the real line.* Fundamenta Mathematicae, vol. 51, no. 1, pages 45–64, 1962. (Cité en page 6.)

[Li 2012] Geng Li, Murat Semerci, Bülent Yener et Mohammed J Zaki. *Effective graph classification based on topological and label attributes.* Statistical Analysis and Data Mining : The ASA Data Science Journal, vol. 5, no. 4, pages 265–283, 2012. (Cité en page 9.)

[Lokshtanov 2009] Daniel Lokshtanov. *Finding the longest isometric cycle in a graph.* Discrete Applied Mathematics, vol. 157, no. 12, pages 2670–2674, 2009. (Cité en pages 25, 27 et 82.)

[Nevo 1972] Eviatar Nevo, George Gorman, Michael Soulé, Suh Yung Yang, Robert Clover et Vojislav Jovanović. *Competitive exclusion between insular Lacerta species (Sauria, Lacertidae)*. Oecologia, vol. 10, no. 2, pages 183–190, 1972. (Cité en page 1.)

[Papadopoulos 1999] Apostolos N Papadopoulos et Yannis Manolopoulos. *Structure-based similarity search with graph histograms*. In Database and Expert Systems Applications, 1999. Proceedings. Tenth International Workshop on, pages 174–178. IEEE, 1999. (Cité en page 9.)

[Peleg 2000] David Peleg. *Proximity-preserving labeling schemes*. Journal of Graph Theory, vol. 33, no. 3, pages 167–176, 2000. (Cité en page 7.)

[Robertson 1983] Neil Robertson et P.D. Seymour. *Graph minors. I. Excluding a forest*. Journal of Combinatorial Theory, Series B, vol. 35, no. 1, pages 39 – 61, 1983. (Cité en page 4.)

[Robertson 1984] Neil Robertson et P.D Seymour. *Graph minors. III. Planar tree-width*. Journal of Combinatorial Theory, Series B, vol. 36, no. 1, pages 49 – 64, 1984. (Cité en page 4.)

[Stoica 2009] Alina Stoica et Christophe Prieur. *Structure of neighborhoods in a large social network*. In Computational Science and Engineering, 2009. CSE'09. International Conference on, volume 4, pages 26–33. IEEE, 2009. (Cité en page 9.)

[Thorup 2005] Mikkel Thorup et Uri Zwick. *Approximate distance oracles*. J. ACM, vol. 52, no. 1, pages 1–24, 2005. (Cité en pages 7 et 8.)

[Völkel 2016] Finn Völkel, Eric Bapteste, Michel Habib, Philippe Lopez et Chloe Vigliotti. *Read networks and k-laminar graphs*. CoRR, vol. abs/1603.01179, 2016. (Cité en pages 4, 6, 9, 10, 13, 19, 23, 75, 89 et 90.)

[Yan 2007] Shuicheng Yan, Dong Xu, Benyu Zhang, Hong-Jiang Zhang, Qiang Yang et Stephen Lin. *Graph embedding and extensions : a general framework for dimensionality reduction*. vol. 29, pages 40–51, 02 2007. (Cité en page 13.)

[Zerbino 2008] Daniel Zerbino et Ewan Birney. *Velvet : algorithms for de novo short read assembly using de Bruijn graphs*. Genome research, pages gr–074492, 2008. (Cité en page 3.)

[Zhao 2012] Xiang Zhao, Chuan Xiao, Xuemin Lin et Wei Wang. *Efficient graph similarity joins with edit distance constraints*. In Data Engineering (ICDE), 2012 IEEE 28th International Conference on, pages 834–845. IEEE, 2012. (Cité en page 8.)

Résumé : En collaboration avec des chercheurs en biologie à Jussieu, nous étudions des graphes issus de données biologiques afin de d'en améliorer la compréhension. Ces graphes sont constitués à partir de fragments d'ADN, nommés reads. Chaque read correspond à un sommet, et deux sommets sont reliés si les deux séquences d'ADN correspondantes ont un taux de similarité suffisant. Ainsi se forme des graphes ayant une structure bien particulière que nous nommons hub-laminaire. Un graphe est dit hub-laminaire s'il peut être résumé en quelques plus courts chemins dont tous les sommets du graphe soient proche. Nous étudions en détail le cas où le graphe est composé d'un unique plus court chemin d'excentricité faible, ce problème a été initialement défini par [Dragan 2017]. Nous améliorons la preuve d'un algorithme d'approximation déjà existant et en proposons un nouveau, effectuant une 3-approximation en temps linéaire. De plus, nous analysons le lien avec le problème de k-laminarité défini par [Völkel 2016], ce dernier consistant en la recherche d'un diamètre de faible excentricité. Nous étudions ensuite le problème du cycle isométrique de plus faible excentricité. Nous montrons que ce problème est NP-complet et proposons deux algorithmes d'approximations. Nous définissons ensuite précisément la structure "hub-laminaire" et présentons un algorithme d'approximation en temps $O(nm)$. Nous confrontons cet algorithme à des graphes générés par une procédure aléatoire et l'appliquons à nos données biologiques. Pour finir nous montrons que le calcul du cycle isométrique d'excentricité minimale permet le plongement d'un graphe dans un cercle avec une distorsion multiplicative faible. Le calcul d'une décomposition hub-laminaire permet quant à lui une représentation compacte des distances avec une distorsion additive bornée.

Mots clés : Théorie des graphes, Cycle isométrique, Excentricité, Plus court chemin, Domination, Graphe de reads, Label de distances, Laminaire

Abstract : In collaboration with reserchears in biology at Université Pierre et Marie Curie, we study graphs coming from biological data in order to improve our understanding of it. Those graphs come from DNA fragments, named reads. Each read is a vertex and two vertices are linked if the DNA sequences are similar enough. Such graphs have a particuliar structure that we name *hub-laminar*. A graph is said to be hub-laminar if it may be represented as a (small) set of shortest paths such that every vertex of the graph is close to one of those paths. We first study the case where the graph is composed of an unique shortest path of low eccentricity. This problem was first definied by [Dragan 2017]. We improve the proof of an approximation algorithm already existing and propose a new one, a 3-approximation running in linear time. Furthermore we show its link with the k-laminar problem defined by [Völkel 2016], consisting in finding a diameter of low eccentricity. We then define and study the problem of the isometric cycle of minimal eccentricity. We show that this problem is NP-complete and propose two approximation algorithms. We then properly define what is an hub-laminar decomposition and we show an approximation algorithm running in $O(nm)$. We test this algorithm with randomly generated graphs and apply it to our biolgical data. Finaly we show that computing an isometric cycle of low eccentricity allows to embed a graph into a cycle with a low multiplicative distortion. Computing an hub-laminar decomposition allows a compact representation of distances with a low additive distortion.

Keywords : Graph theory, Isometric Cycle, Eccentricity, Shortest path, Domination, Read graphs, Distance Labeling, Laminarity