Decomposing a Graph into Shortest Paths with Bounded Eccentricity

Etienne Birmelé[1], Fabien de Montgolfier[2], Léo Planche[3], and Laurent Viennot[4]

1 **MAP5, UMR CNRS 8145, Univ. Sorbonne Paris Cité, France**
`etienne.birmele@parisdescartes.fr`
2 **IRIF, UMR CNRS 8243, Univ. Sorbonne Paris Cité, France**
`fm@liafa.univ-paris-diderot.fr`
3 **MAP5, UMR CNRS 8145, Univ. Sorbonne Paris Cité and IRIF, Paris, France**
`Leo.Planche@liafa.univ-paris-diderot.fr`
4 **INRIA and IRIF, Paris, France**
`Laurent.Viennot@inria.fr`

Abstract

We introduce the problem of hub-laminar decomposition which generalizes that of computing a shortest path with minimum eccentricity (MESP). Intuitively, it consists in decomposing a graph into several paths that collectively have small eccentricity and meet only near their extremities. The problem is related to computing an isometric cycle with minimum eccentricity (MEIC). It is also linked to DNA reconstitution in the context of metagenomics in biology. We show that a graph having such a decomposition with long enough paths can be decomposed in polynomial time with approximated guaranties on the parameters of the decomposition. Moreover, such a decomposition with few paths allows to compute a compact representation of distances with additive distortion. We also show that having an isometric cycle with small eccentricity is related to the possibility of embedding the graph in a cycle with low distortion.

1998 ACM Subject Classification G.2.2 Graph Theory

Keywords and phrases Graph Decomposition, Graph Clustering, Distance Labeling, BFS, MESP

Digital Object Identifier 10.4230/LIPIcs.ISAAC.2017.15

1 Introduction

The goal of this paper is to extend the MESP (Minimum Eccentricity Shortest Path) Problem from Dragan and Leitert [5] and the related problem of recognizing k-laminar graphs from Völkel *et al.* [16]. Both consist in finding a shortest path (in the sense that no path joining the same endpoints is shorter) k-dominating a graph (every vertex is at distance at most k from that path). The k-laminar problem additionally requires that path to be a diameter (there is no longer shortest path in the graph). Relationships between the two parameters are derived in [4].

To generalize this problem to more complex underlying structures, we introduce the problem of decomposing a graph into subgraphs with bounded shortest-path eccentricity. More precisely, we introduce the hub-laminar decomposition as a set of paths that k-dominates the graph and meet only near their extremities. To formalize this property, we introduce the notion of hub, that is a ball with fixed radius r centered at a path endpoint. The laminar associated to a path is the set of nodes k-dominated by the path. Our definition requires that an edge between two nodes belonging to two different laminars must also belong to a

28th International Symposium on Algorithms and Computation (ISAAC 2017).
Editors: Yoshio Okamoto and Takeshi Tokuyama; Article No. 15; pp. 15:1–15:13
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

hub. The degree of a hub is then the number of laminars that meet in the hub. The main result of the paper is that computing such a decomposition becomes tractable when hub centers are far enough one from another, or equivalently when paths are long enough. The MESP problem is equivalent to a hub-laminar decomposition with one laminar.

Such a generalization is naturally interesting in networks where one might want to identify a set of speedy linear routes that are "highly accessible" with applications in communication networks, transportation planning and water resource management. It is also motivated by DNA assembly in biology. DNA sequencing proceed through the reading of DNA fragments that must be assembled. When a single DNA strand is sequenced, comparison of fragments leads to a graph with "laminar" structure [16] that is with large diameter and small shortest path eccentricity. In the context of metagenomics, several DNA strands are sequenced together and more complex structures appear (see Figure 1 in [16]). Identifying the laminar structures of such graphs is typically encountered in metagenomic approaches for evolution questions (see e.g. [13]). The problem of the assembly (gluing DNA fragments to reconstruct a DNA strand) is then mixed with that of binning (sort DNA strands into groups that represent an individual genome or genomes from closely related organisms). See [14] for a presentation of assembly and binning problems in the context of metagenomics. Efficient decomposition of a graph into laminars could thus enhance the techniques for assembly and binning in this context.

The problem of decomposing a graph into λ laminars that k-cover the graph is not well defined as there may be several trade-offs of parameters λ and k. However, we show that when laminars are long enough compared to parameters r and k, then all (r,k)-hub-laminar decompositions are equivalent (same global structure) and have closely located hubs (except for hubs of degree two that do not affect the global structure). This implies for example that the positions of the extremities of the minimum eccentricity shortest path (MESP) can be approximated within $O(k)$ distance when the diameter of a graph is large with respect to the eccentricity k of the MESP.

From a graph perspective, a very natural generalization of MESP is the problem of finding a minimum eccentricity isometric cycle (MEIC), that is a cycle preserving distances that has minimum eccentricity k. Note that such a cycle can be seen as a hub-laminar decomposition with two laminars and two hubs with degree two. An important motivation for the MESP problem is its relationship with embedding a graph into the line with small multiplicative distortion [5]. We similarly show that the MEIC problem is related to embedding a graph into a circle with low multiplicative distortion, i.e. such that distances in the circle are within a constant factor of distances in the graph. Note that circle distortion is bounded by line distortion as a line segment can isometrically be embedded in a sufficiently long circle. (However, line distortion can be much larger than circle distortion.) Graph embedding in classical metrics is a well studied problem [9, 10]. Another related subject with abundant literature is that of compactly representing the distances of a graph [15, 12]. We show that a decomposition with few laminars ensures a compact representation of distances with bounded additive distortion.

Related works

Finding a MESP is NP-complete but can be approximated within a constant factor [5]. Better trade-off between computation time and approximation factor for MESP is obtained in [4]. The problem of efficiently representing the distances in a graph encompasses a vast literature dating from metric embedding [1]. Approximating embedding with low distortion is introduced in [2] where some results are provided in the case of the line. The case of embedding the metric induced by an unweighted graph is studied in [3]. Embedding a graph metric into the line with minimum distortion is NP-complete but fixed parameter tractable with respect to distortion [6]. Approximate distance oracles, i.e. compact data-structures for representing an approximation of distances, are investigated in [15]. A particular approach introduced by Peleg [12] resides in assigning a label to each node of a graph such that the distance between two nodes can be estimated from their labels. Several results exist about the trade-off between label size and approximation quality. Exact distance estimation is investigated in [8] and requires $\Omega(n)$ bits labels for general graphs. Approximation with a constant factor and sub-linear label size is derived in [15]. Some results concern additive approximation such as [7] in the case of hyperbolic graphs. A longest isometric cycle can be found in polynomial time [11].

2 Definitions

We consider finite, undirected and *connected* graphs (the connectivity is always assumed within the paper). Given a graph G, with vertex set $V(G)$ and edge set $E(G)$, we let $d_G(u,v)$ denote the *distance* between two vertices, i.e. the length of a shortest path from u to v. When the graph G is clear from the context, we omit the G subscript and simply write $d(u,v)$. Let $B(u,r) = \{v \in V(G) \mid d(u,v) \le r\}$ denote the *ball* of radius r centered at u. Given a set of vertices U we set $B(U,r) = \cup_{u \in U} B(u,r)$. Given two sets U and W of vertices, we say that U *k-dominates* W when every vertex in W is at distance at most k from some vertex in U, i.e. $W \subseteq B(U,k)$. We say that U has *eccentricity* k, denoted $ecc(U) = k$, when k is the smallest integer such that $B(U,k) = V(G)$. A path P in G is a sequence of nodes such that any two consecutive nodes are linked by an edge of G. We consider only simple paths: a node appears at most once in the sequence. The first node of the sequence and the last one are called the *endpoints* of P. For the simplicity of notations, we also let P denote the set of nodes appearing in the sequence. For any vertices u and v on P, we denote by P_{uv} the subpath of P having u and v as endpoints.

2.1 Hub-laminar decomposition

▶ **Definition 1** (Hub-laminar decomposition)**.** Consider a connected undirected graph G, two positive integers r and k, $H = \{h_1, \ldots, h_q\}$ a set of vertices of G called *hub centers*, and $\mathcal{P} = \{P_1, \ldots, P_p\}$ a set of paths of G called *laminar paths.* A ball $B(h,r)$ with $h \in H$ is called a *hub*, and a set $B(P,k)$ with $P \in \mathcal{P}$ is called a *laminar.* $(H,\mathcal{P})$ is an (r,k)*-hub-laminar decomposition* of G if the following conditions are satisfied:

1. each laminar links two hubs centers: the endpoints h, h' of any $P \in \mathcal{P}$ belong to H and for every other hub $h'' \in H \setminus \{h, h'\}$, $B(P,k) \cap B(h'', r+1) = \emptyset$,
2. the laminars and the hubs dominate G: $V(G) \subseteq \bigcup_{h \in H} B(h,r) \cup \bigcup_{P \in \mathcal{P}} B(P,k)$,
3. each laminar path is locally a shortest path: any path $P \in \mathcal{P}$ with endpoints h and h' is a shortest path of the graph $G[B(P,k) \cup B(h,r) \cup B(h',r)]$,
4. laminars meet at hubs only: for all $i \ne j$ and $uv \in E(G)$ such that $u \in B(P_i,k)$ and $v \in B(P_j,k)$, there is a hub center $h \in H$ such that P_i and P_j both have h as endpoint and $u, v \in B(h,r)$.

The *minimal laminar length* of a decomposition $(H,\mathcal{P})$, denoted l, is the minimal length of the paths in $\mathcal{P}$. Its *laminar size*, denoted λ, is the number of paths in $\mathcal{P}$.

A hub-laminar decomposition $(H,\mathcal{P})$ with $l \ge 2r+1$ forms a partition of the edges of G in the following sense: each edge is either inside exactly one hub (possibly touching many

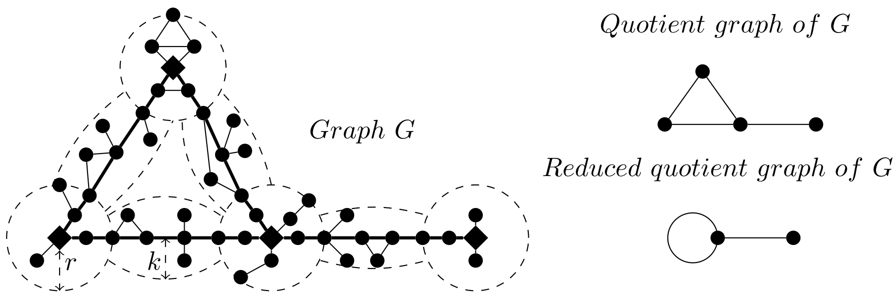

Figure 1 Illustration of an hub-laminar decomposition with $r = 2, k = 1$. Every vertex is at distance r from a hub center (diamond vertices) or at distance k from a laminar path (bold paths between hub centers).

laminars ending in that hub), i.e $\exists! h \in H$ s.t. $u, v \in B(h, r)$; or, else, inside a unique laminar (possibly touching one hub extremity of that laminar), i.e, $\exists! P \in \mathcal{P}$ s.t. $u, v \in B(P, k)$.

Figure 1 illustrates this definition and the notion of quotient graph that we define next. This definition basically defines a decomposition into k-neighborhoods of internally far apart shortest paths. It may seem a bit involved, but we think it expresses in a minimalist way what we mean by "internally far apart" with Item 4. Items 1 and 2 indicate that the graph is decomposed into laminars which are k-neighborhoods of certain paths and hubs which are balls centered at the extremities of those paths. Item 3 requires path to be shortest in the induced graph (rather than in G), to allow laminars with different length.

2.2 Quotient graph and equivalence between decompositions

As previously mentioned, the hub-laminar decomposition gives naturally raise to a skeleton, which can be simplified into a *quotient graph*.

▶ **Definition 2** (quotient graph and reduced quotient)**.** Given a graph G and an (r, k)-hub-laminar decomposition $(H, \mathcal{P})$ of G, the *quotient* of this decomposition is an edge-labeled multigraph with vertex-set H and for each $P \in \mathcal{P}$ with endpoints h, h' there is an edge hh' whose label is the length of P.

The *degree of a hub* denotes the degree of the corresponding vertex in the quotient graph, or equivalently the number of laminar paths its center is the endpoint of.

The *reduced quotient graph* of a decomposition $(H, \mathcal{P})$ is the multigraph obtained from its quotient graph by repeatedly removing degree 2 nodes: for every vertex u of the quotient incident with exactly two edges uv and uw with respective labels a and b, u and both edges are removed and a new edge vw is added with label $a + b$. (It is a loop when $v = w$.)

When the quotient is not a cycle (a case specifically adressed by MEIC, see Section 3) the reduced quotient is well defined and unique (recall that graphs are supposed connected).

▶ **Definition 3** (equivalence between decompositions)**.** Two hub-laminar decomposition of a given graph G, possibly with different parameters r, k, are *D-equivalent* if they have the same reduced quotient graph, up to an isomorphism ϕ of vertex-sets such that $d(h, \phi(h)) \leq D$ (d is the distance between hub centers in G, not in the reduced quotient).

2.3 Isometric cycle, circle embedding and distance labeling

A cycle C in a graph G is *isometric* if it preserves distances, i.e. $d_C(u,v) = d(u,v)$ for all $u, v \in V(C)$. In other words, for any pair u, v of nodes on the cycle, one of the two path linking u and v in the cycle is a shortest path in the graph. Note that an isometric cycle is necessarily an induced cycle. The MEIC problem consists in finding an isometric cycle with minimum eccentricity. It can be shown to be NP-complete following a similar proof as [5] for the NP-completeness of MESP problem.

A *circle embedding* of a graph G is a mapping $f : V(G) \to C$ where C is a circle of given length c. It has distortion γ if $d(u,v) \leq d_C(f(u), f(v)) \leq \gamma d(u,v)$ for all u, v in $V(G)$. The *circle distortion* $cd(G)$ of G is the minimum distortion of a circle embedding of G.

A distance labeling of a graph G consists in assigning a label L_u to each node $u \in V(G)$ together with a distance estimation function f that outputs an estimation of $d(u,v)$ when given L_u and L_v as input. It has additive distortion α if $d(u,v) \leq f(L_u, L_v) \leq d(u,v) + \alpha$ for all u, v in G.

3 Main results

Obviously, the reduced quotient graph of a graph having a (r,k)-hub-laminar decomposition follows the following trichotomy: it is either a path, a cycle or has a degree three node. We treat separately the three cases.

In the first case, the graph has a shortest path with eccentricity $\max\{3k, 2r\}$ and can be recognized through an approximate MESP algorithm such as [4]. (The $\max\{3k, 2r\}$ bound is a consequence of Lemma 12 given in Section 4.) In the second case, the graph has an isometric cycle with eccentricity at most $\max\{3k, 2r\}$. To recognize such graphs, we propose an approximate MEIC algorithm:

▶ **Theorem 4.** *Given a graph containing a K-dominating isometric cycle with length ℓ, a $6K$-dominating isometric cycle can be computed in $O(n^{4.752}\log(n))$ time. Moreover, the computed cycle is indeed $3K$-dominating when $\ell \geq 12K + 2$.*

We obtain therefore an algorithm for approximating circle embedding with low distortion.

▶ **Proposition 5.** *If a graph has circle distortion γ, it is possible to embed it in a circle with distortion $O(\gamma^2)$ in polynomial time.*

Recognizing the general case of decomposition is not a well defined problem as several decompositions may yield different trade-offs of the parameters. However, when laminars are long enough, all (r,k)-hub-laminar decompositions are indeed $O(k)$ equivalent. This can be seen as a consequence of the following recognition result.

▶ **Theorem 6.** *Given a graph G having a (r,k)-hub-laminar decomposition $(H, \mathcal{P})$ of minimal laminar length $\ell \geq 8r + 60k + 4$ and integers K, R such that $K \geq 3k$, $R \geq 2K + 3r + 3k$ and $2R + 8K < \ell - 2r - 18k - 4$, it is possible to compute in $O(\min(n,\lambda)m)$ time a (K,R)-hub-laminar decomposition which is $(K + 2r)$-equivalent to $(H, \mathcal{P})$.*

From the graph metric point of view, we obtain then a compact representation of distances:

▶ **Proposition 7.** *Given a graph G having an (r,k)-hub-laminar decomposition with laminar size λ, it is possible to compute in polynomial time a $O(\max\{k,r\})$-additive distance labeling with $O(\lambda \log n)$ bit labels.*

Due to lack of space, the proofs of these theorems, and of the lemmas and propositions stated below, are put in Appendix.

4 Algorithms

4.1 Minimum Eccentricity Isometric Cycle

We propose to approximate the MEIC problem by computing a longest isometric cycle, that is an isometric cycle of G with maximum length. The following lemma shows that a longest isometric cycle $O(k)$-dominates any k-dominating isometric cycle.

▶ **Lemma 8.** *Let G be a graph with an isometric cycle $C = c_1, ...c_p$ k-dominating G, and let D be a longest isometric cycle of G. Every vertex of C is at distance at most $5k$ of D. Furthermore, if D has length more than $12k + 2$ then every vertex of C is at distance at most $2k$ of D.*

Consequently, a longest isometric cycle in a graph is a 6-approximation for the MEIC problem, and a 3-approximation when the graph has a diameter large enough. As shown in [11], a longest isometric cycle can be computed in $\mathcal{O}(n^{4.752}\log(n))$ time. Theorem 4 is thus a direct consequence of this and Lemma 8.

4.2 General case outline

Consider a graph G having a (r, k) hub-laminar decomposition $(H, \mathcal{P})$ of minimal laminar length ℓ and having at least one hub of degree at least 3. The underlying idea of the algorithm is to use BFS (Breadth-first search) to compute shortest paths and their K-neighborhoods, K being chosen large enough to dominate every laminar traversed by the considered shortest paths, but small enough compared to ℓ to detect all hubs of degree at least 3.

The first step, called $FindHubs$, consists in applying the procedure $NextHub$ described in section 4.4 until it discovers no new hubs. This step yields two sets of hub-centers A and B, respectively called *unmovable* and *movable* hub-centers, which will be used to determine the laminars. An unmovable hub center $a \in A$ corresponds to exactly one hub center $h \in H$ such that $d(a, h)$ is bounded. It will be shown that A contains exactly one such vertex for every hub-center of H which degree is not 2.

A movable hub center $b \in B$ will only be added by $NextHub$ in a configuration corresponding to a cycle in the quotient graph of $(H, \mathcal{P})$ containing only one hub of degree at least 3, like the three laminars on the left of Figure 1. This is called a *Problematic Configuration* We then know there exists at least a degree 2 hub $h \in H$ somewhere in that cycle, but if they are thin enough they may remain merged in the laminars and we can not bound $d(b, h)$, and b may be moved in the second step described bellow.

The laminars are determined in a second step by the $FindLaminars$ procedure, which links the hub-centers of the previous step by shortest paths. The only difficulty which has to be taken into account refers to hubs of degree 2 in $(H, \mathcal{P})$. Indeed, the BFS runs of the hub-detection step may have missed one of them because they K-dominated it, whereas the BFSs of second step don't. In that case, the set of hubs A is adapted by adding the new discovered hub, and if needed, the corresponding movable hub center is deleted from B.

Figure 2 gives a summary of the two steps by showing a possible outcome of the $FindHubs$ and $FindLaminars$ on an example. The $FindHubs$ procedure detects all hubs of degree different from 2 and some of those of degree 2. Moreover, it places a movable hub on each problematic configuration. $FindLaminars$ then computes the corresponding laminars, adding new hubs if a hub of degree 2 missed in the first step is detected. Some of them may however still be undetected, being replaced by a movable hub or just missing in the final

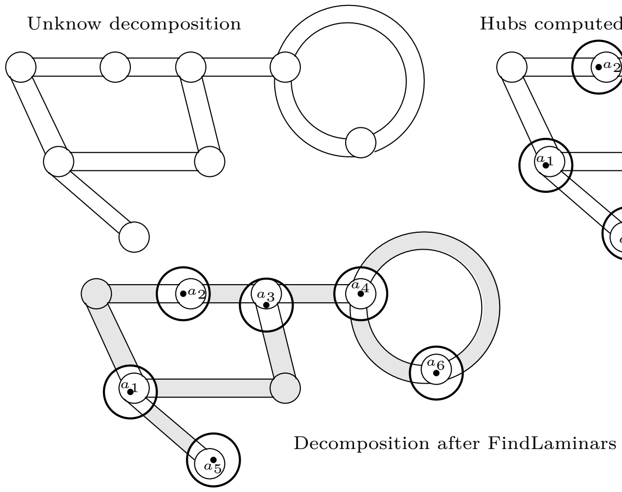

Figure 2 Illustration of the different steps of the algorithm. The $(H, \mathcal{P})$ decomposition is unknown (top left). Notice a Problematic Case on the right of the graph: a cycle of laminar with only one degree not 2 hub. First, hub centers are computed such that every hub $B(h,r), h \in H$ with degree different from 2 is covered by $B(a_i, R), a_i \in A$ (top right). Finally the laminars are computed (greyed, bottom) and some movable hubs may be moved (like b_1 moved into a_5). Some thin degree 2 hubs from H are not found but merged in the K-laminars. Fortunately the hub center a_5 was found by BFS in the second step, but we could also have output b_1 instead, or both a_5 and b_1, yielding in any case an equivalent reduced quotient.

decomposition. The quotient graphs of the decomposition supposed by Theorem 6 and that of the constructed decomposition may therefore be different, but their reduced quotients are equivalent.

4.3 Some rules to compute a decomposition

We rely on the following properties for running our algorithm. The first tool is used to identify hub centers. Fortunately there is a pattern that, when it occurs, signals that any hub-laminar decomposition must have a hub nearby:

▶ **Lemma 9** (Hub trigger). *Consider three numbers r, k and $K \geq 3k$. If there exists*

- *a shortest path Q from a to b*
- *a vertex $u \in V(Q)$ such that $d(a,u) > K + 6k$ and $d(b,u) > K + 6k$*
- *a vertex v such that $d(u,v) = K$*
- *a vertex w such that $vw \in E(G)$ and $d(Q,w) = K + 1$ and $d(u,w) = K + 1$,*

then any (r,k)-hub-laminar decomposition $(H, \mathcal{P})$ has a hub center $h \in H$ with $d(u,h) \leq K + r$.

Fig. 3 (right) illustrates this. This pattern, when found, allows to propose u as a hub. And it is very powerful, since every hub h of degree at least three of any (r,k)-hub-laminar decomposition shall trigger this pattern, for any shortest path Q passing close to h and long enough, as stated by the following lemma:

▶ **Lemma 10** (Degree ≥ 3 Hub Detection). *Consider a graph admitting an (r,k)-hub-laminar decomposition $(H, \mathcal{P})$, and having a hub $h \in H$ whose degree is at least 3, and consider $K \geq 3k$. For any shortest path Q and vertex $u \in V(Q)$ such that*

- *$d(u,h) \leq K$,*
- *u is at distance at most $r + 4K + 9k + 2$ of both endpoints of Q,*

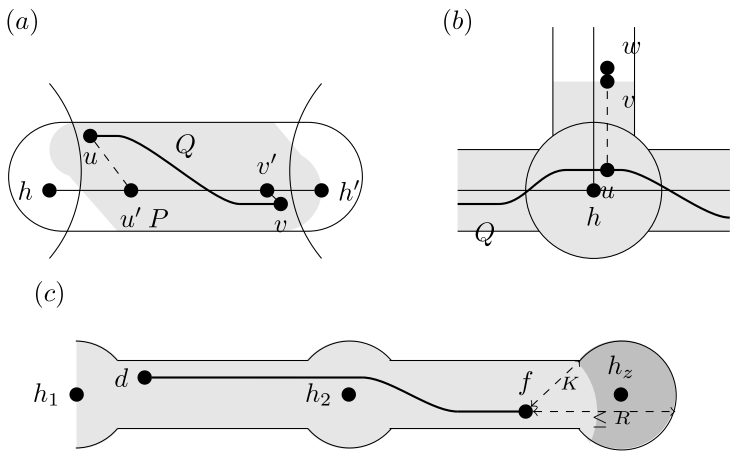

Figure 3 Illustration of the structural properties. (a) Lemma 12: The part of the laminar k-covered by $P_{u'v'}$ is K-covered by Q. (b) Lemma 9: If Q goes through a hub of degree ≥ 3, a triplet u, v, w can be found, and it is the only such case far apart the extremities of Q. (c) Lemma 11.

there exists

- $x \in V(Q)$ *such that x is at distance at least $K + 6k$ from both endpoints of Q, and*
- $vw \in E(G)$ *such that* $d(x, v) = K$ *and* $d(Q, w) = K + 1$

Notice that if a graph admits an (r, k)-hub-laminar decomposition $(H, \mathcal{P})$ where all hubs have degree at least three, then the pattern is enough to find all its hubs, or more exactly to compute a set of hubs H' which is in bijection ϕ with H, i.e. for all $h \in H$ $d(h, \phi(h)) \leq K + r$. Of course to do so in polynomial time we shall use a clever collection of paths Q to trigger all hubs. This is the idea developed in the algorithm. But before we shall explain how to deal with degree 1 hubs, the "dead end" hubs.

▶ **Lemma 11** (Dead-end hub). *Consider the graph G' induced by a sequence of incident hubs and laminar $H_1, L_1, H_2, ...H_z$, such that h_1 and h_z are at distance at least $2R + r + 2$. Suppose moreover that H_z is a hub of degree 1 and all other hubs but H_1 are of degree 2.*

Let d in L_1 be at distance at most $R + r$ of h_1 and f a vertex of G' the furthest from d. f is then at distance at most $2r + 2k$ from h_z.

This lemma allow to approximate h with f. As we have just seen, hubs with degree different from 2 are, in some sense, uniquely defined (up to a certain distance) in any hub-laminar decomposition of given parameters. Degree 2 hubs however may be added at discretion on any hub-laminar decomposition, in the middle of long laminar, so we cannot imagine a sufficient condition for detecting them. However, they are necessary in very few case, namely

- to dominate a vertex at distance more than k, but less than r, inside of a r-laminar (not k-laminar)
- for the Problematic Configuration, since a laminar must have two distinct extremities

The last property, proved in [4], deals no more with computing hub centers but with computing laminars. While it is is NP-hard to find a shortest path that k-dominates a k-laminar graph, any path $3k$-dominates a section of the laminar between its extremities. Stated more formally:

▶ **Lemma 12** (Path local dominating). *Consider a shortest path P (say, from h to h'). Let Q be a path from u to v contained in $B(P,k)$.*

Assume there exists $u' \in \mathcal{P}$ and $v' \in P$ such that $d(u,u') \leq k$ and $d(v,v') \leq k$.

Then every vertex of $P_{u'v'}$ is at distance at most $2k$ from Q.

Furthermore, every vertex of $B(P_{u'v'},k)$ is at distance at most $3k$ of Q.

Fig. 3 (left) illustrates this. We extensively use this lemma for designing an approximation algorithm: P is any laminar path, and Q is chosen to $3k$ dominate the middle of the laminar of P, i.e. all vertices far enough from P extremities (Lemma13 and 14 define "far enough" as $2R + 8K + 2r + 18k + 4$) and we therefore get a decomposition into $3k$-laminar graphs.

4.4 Finding hubs

In the following section, graphs are vertex-colored, with possibly some uncolored vertices.

The general idea of the algorithm is that starting with an uncolored graph, we end with a fully colored one, such that :

- Every vertex at distance less than $K+1$ of a computed hub a is colored with color a.
- Other vertices of the graph are colored with color *lam*.

4.4.1 The StopBFS function

The **StopBFS** procedure, provided a vertex d and a color c, consists in running an usual Breadth-first search, starting at vertex d, with the following additional rules:

- only uncolored vertices are put in the BFS queue
- the BFS stops immediately if a vertex f is visited (i.e. extracted from BFS queue) and f has a colored neighbor whose color is not c.
- otherwise, if the BFS stops because its queue is empty, let f be the last visited vertex
- function $StopBFS(d,c)$ returns the BFS path P from d to f (which is a shortest path in the graph induced by G after removing c-colored vertices).

4.4.2 Finding a new hub: NextHub

Given a vertex s, typically corresponding to an already selected hub center, the **NextHub** procedure (see pseudo-code in Algorithm 1) detects new hubs: it colors $B(s,R)$ with a new color and runs a StopBFS procedure from its border. In the case of a not deep-enough tree, the discovered vertices are colored to not be reused during the hub discovery. Otherwise, it may either find a new hub of degree at least 3 by Lemma 10, find a new hub of degree 1 by Lemma 11, meet another hub and dominate a laminar by Lemma 12 or cycle and come back to hit $B(s,R)$. The later case indicates that the algorithm encountered the problematic configuration and induces the creation of a movable hub.

Given a path P, $r3K(P)$ denotes the subpath of P obtained by removing the $3K$ first and $3K$ last vertices of P. In the sequel, sets A and B respectively denote the unmovable and movable hub centers.

4.4.3 Finding all hubs : FindHub

The $FindHub$ simply consists in considering the initial uncolored graph G and to construct the sets A and B of unmovable and movable hub-centers by repeatedly applying $NextHub$ (see pseudo-code in Appendix).

Algorithm 1: NextHub

NextHub
Input: A graph G with possibly colored vertices, integers R and K, hub-center sets A and B, and a vertex s
Output: Updated sets A, B and vertex coloring
Color every vertex in $B(s, R)$ with a new color $col(s)$
Choose an uncolored vertex d at distance $R + 1$ from s
Let $P = stopBFS(d, col(s))$ and f the last vertex of P
If *P is of length less that $2R + 4K + 2$* **then**
  `/* Not deep enough tree: no laminar is crossed */`
  Color all vertices visited by $stopBFS(d, col(s))$ with color lam
else if *$\exists w, a$ s.t. $col(w) \neq col(s)$ and $h \in r3K(P)$ and $d(w, a) = K+1$ and $d(w, P) = K+1$* **then**
  `/* A hub has been detected by Lemma 9 configuration */`
  Add to A the first vertex a of $r3K(P)$ satisfying the above
else if *f is at distance less than $2K$ of $B(s, R)$* **then**
  `/* P is deep, found no hub and came back near the root: problematic configuration */`
  Add to B the vertex b in the middle of P
  Color uncolored vertices in $B_{G \setminus \{B(d,R) \cup B(f,R)\}}(P, K)$ with color lam
else if *f is not adjacent to a colored vertex* **then**
  `/* P is long, found no hub and doesn't come back: dead end */`
  Add f to A
else
  `/* P links B(s,R) to a vertex of a color different from col(s). The dominated vertices correspond to a laminar. */`
  Color uncolored vertices in $B_{G \setminus \{B(d,R) \cup B(f,R)\}}(P, K)$ with color lam

For the first call, we first compute a long path Q using a double BFS. More precisely, starting at any s_0, we compute a furthest node s and then repeatedly apply $NextHub$ until a vertex a is added to A. If there is a unique hub of degree at least three, the fact that $\ell > 2R + 8K + 2r + 18k + 4$ ensures that the deepest vertex of any BFS is at distance greater than $R + (r + 4K + 9k + 2)$ of the hub center. If there are at least two hubs of degree at least three, $\ell > 2R + 8K + 2r + 18k + 4$ implies that any vertex is at distance greater than $\frac{\ell}{2} > R + (r + 4K + 9k + 2)$ of one of the two hub-centers. In any case, the $Next\_Hub$ function applied to s has to find the configuration from Lemma 9 at some point, ensuring that a first hub center $a \in A$ is found. We set $A = \{a\}$ and $B = \emptyset$, and uncolor the whole graph.

Once this first vertex of A has been found, $NextHub$ is run while there exists a hub center $a \in A$ having an uncolored vertex in its $R + 1$-neighborhood. If ℓ is large, the $FindHubs$ procedure finds all hubs of $(H, \mathcal{P})$ up to those of degree 2, as stated in the following lemma.

▶ **Lemma 13.** *Suppose that $(H, \mathcal{P})$ has at least a hub of degree 3, and $\ell(H, \mathcal{P}) > 2R + 8K + 2r + 18k + 4$. Then, for every vertex $a \in A$, there exists a vertex $h \in H$ such that their distance is at most $K + 2r$. Conversely, for every $h \in H$ of degree different from 2, such a vertex a is selected in A.*

4.5 Finding laminars

At this step we have a set of unmovable hubs including all hubs of degree 1 or at least 3, and potentially those of degree 2. Moreover, the set B of movable hub-centers indicates the places where problematic configuration occur. We have to identify the laminars and their paths, keeping in mind that some new hubs of degree 2 may be detected. Each path is found by a BFS starting at an hub center and ending at the first other hub center encountered.Then we remove from the graph the vertices from the laminar, but not the hubs. For each path P linking two hub centers h and h', the vertices from $B(P,k) - (B(h,R) \cup B(h',R)$ are removed from the graph. Hub center h is no more used when $B(h,R)$ becomes disconnected and the whole process ends when the graph consists in disconnected hubs only.

To prevent any difficulty arising from ending a shortest path with a movable hub $B(b,R)$, we start by those hubs to run BFSs. Indeed, such hubs correspond to a configuration where the quotient of the decomposition (the one supposed by Theorem 6 and the computed one, since they have the same reduced quotient) contains a cycle . If a movable hub has been used, it means that only one hub center $a \in A$ corresponding to hub-center $h \in H$ of degree at least 3 has been found, and that all other hubs are of degree 2 on the cycle and have been missed. Starting from b, the first element of $A \cup B$ which is hit is then a, whatever direction was followed from $B(b,R)$. Thus, two BFS from b to a are run and follow the ring in opposite directions. Either the two obtained paths K-dominate all vertices of the ring, in which case b is transferred to A and the two paths added to $\mathcal{Q}$; Or there exist a vertex in the ring which is not K-dominated. This vertex is then at distance at most $K + 2r$ of some $h \in H$ (cf Appendix for a proof). It is thus added to A and b is deleted from B.

Once the movable centers have been considered, no other places with problematic configurations are left. One therefore just has to draw shortest paths between vertices of A, and Lemma 12 ensures that they cover the laminars of $(H,\mathcal{P})$. The only difficulty is again that a hub of degree 2 that had not been discovered by $FindHubs$ may this time be discovered by $FindLaminars$ because Lemma 9 configuration is encountered. In that case, this degree 2 hub center is added to A and a new BFS is run from it. See pseudo-code of $FindLaminars$ in Appendix.

▶ **Lemma 14.** *Suppose that $(H,\mathcal{P})$ has at least a hub of degree 3, and $\ell(H,\mathcal{P}) > 2R + 8K + 2r + 18k + 4$. Suppose that $FindLaminars$ is run on sets A and B returned by $FindHubs$. Then it ends with every vertex deleted or marked as undeletable.*

As shown in the appendix, Lemmas 13 and 14 imply Theorem 6.

5 Embedding and distance labeling

5.1 Circle embedding with bounded distortion

Proposition 5, stated in Section 3, is a consequence of Theorem 4 and the two following propositions.

▶ **Proposition 15.** *Any graph G having a circle embedding with distortion γ has a shortest path or an isometric cycle with eccentricity $\lfloor \gamma/2 \rfloor$ at most.*

▶ **Proposition 16.** *Given a graph G and an isometric cycle with eccentricity k in G, an embedding of G in a circle with distortion $O(k \cdot cd(G))$ can be computed in polynomial time.*

Proof sketch. The construction of the embedding is similar to that of [5] with Euler tours of trees of depth k rooted in the cycle (see [5]). We then obtain an embedding of the graph in a cycle of length $2n$ at most that can be easily embedded in a circle with same length. The distortion of an edge uv of G is then at most twice the size of the union S of trees rooted on the shortest path of the cycle from the root u' of the tree of u to the root v' of the tree of v. As we have $d(u', v') \leq 2k + 1$, the diameter of S is at most $4k + 1$. Now consider an embedding of G in a circle C with distortion $\gamma = cd(G)$. Two nodes of S are embedded at distance at most $\gamma(4k + 1)$ in the circle and different nodes are at distance 1 at least. We thus have $|S| \leq \gamma(8k + 2)$, and our embedding has distortion $O(\gamma k)$. ◀

5.2 Distance labeling for general hub-laminar decomposition

A hub-laminar decomposition of a graph G allows to compute a compact representation of distances in G with additive distortion. A distance labeling is said to be c-additive and have s bit labels when the label L_u assigned to a node u contains at most s bits and for all pairs of nodes u, v, a distance estimation $\widehat{d_{uv}}$ can be computed from L_u and L_v such that $d(u, v) \leq \widehat{d_{uv}} \leq d(u, v) + c$. Proposition 7 is a consequence of Theorem 6 and the following proposition.

▶ **Proposition 17.** *Given a (r, k)-hub-laminar decomposition $(H, \mathcal{P})$ with λ laminars of a graph G, a $\max(4k, 2r)$-additive distance labeling with $O(\lambda \log n)$ bit labels can be computed in polynomial time.*

Acknowledgments. The authors thank Michel Habib for inspiring discussions about k-laminar graphs, and Eric Bapteste, Philippe Lopez and Chloé Vigliotti for raising the problem of identifying complex laminar structures in biological graphs.

References

1 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*, 288(15):A731–A734, 1979.

2 Mihai Badoiu, Julia Chuzhoy, Piotr Indyk, and Anastasios Sidiropoulos. Low-distortion embeddings of general metrics into the line. In *SToC 2005*, pages 225–233. ACM, 2005. doi:10.1145/1060590.1060624.

3 Mihai Badoiu, Kedar Dhamdhere, Anupam Gupta, Yuri Rabinovich, Harald Räcke, R. Ravi, and Anastasios Sidiropoulos. Approximation algorithms for low-distortion embeddings into low-dimensional spaces. In *SODA 2005*, pages 119–128. SIAM, 2005. URL: http://dl.acm.org/citation.cfm?id=1070432.1070449.

4 Etienne Birmelé, Fabien de Montgolfier, and 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. doi:10.1007/978-3-319-48749-6_16.

5 Feodor F. Dragan and Arne Leitert. On the minimum eccentricity shortest path problem. In *Algorithms and Data Structures - 14th International Symposium, WADS 2015, Victoria, BC, Canada, August 5-7, 2015. Proceedings*, pages 276–288, 2015. doi:10.1007/978-3-319-21840-3_23.

6 Michael R. Fellows, Fedor V. Fomin, Daniel Lokshtanov, Elena Losievskaja, Frances A. Rosamond, and Saket Saurabh. Distortion is fixed parameter tractable. *TOCT*, 5(4):16:1–16:20, 2013. doi:10.1145/2489789.

7 Cyril Gavoille and Olivier Ly. Distance labeling in hyperbolic graphs. In *ISAAC 2005*, volume 3827 of *Lecture Notes in Computer Science*, pages 1071–1079. Springer, 2005. doi:10.1007/11602613_106.

8 Cyril Gavoille, David Peleg, Stéphane Pérennes, and Ran Raz. Distance labeling in graphs. *J. Algorithms*, 53(1):85–112, 2004. doi:10.1016/j.jalgor.2004.05.002.

9 Piotr Indyk. Algorithmic applications of low-distortion geometric embeddings. In *FOCS 2001*, pages 10–33. IEEE Computer Society, 2001. doi:10.1109/SFCS.2001.959878.

10 Piotr Indyk and Jiří Matoušek. Low-distortion embeddings of finite metric spaces. In Jacob E. Goodman and Joseph O'Rourke, editors, *Handbook of Discrete and Computational Geometry, Second Edition.*, pages 177–196. Chapman and Hall/CRC, 2004. doi:10.1201/9781420035315.ch8.

11 Daniel Lokshtanov. Finding the longest isometric cycle in a graph. *Discrete Applied Mathematics*, 12(157):2670–2674, 2009.

12 David Peleg. Proximity-preserving labeling schemes. *Journal of Graph Theory*, 33(3):167–176, 2000. doi:10.1002/(SICI)1097-0118(200003)33:3<167::AID-JGT7>3.0.CO;2-5.

13 Jimmy H et al. Saw. Exploring microbial dark matter to resolve the deep archaeal ancestry of eukaryotes. *Phil. Trans. R. Soc. B*, 370(1678):20140328, 2015.

14 Torsten Thomas, Jack Gilbert, and Folker Meyer. Metagenomics-a guide from sampling to data analysis. *Microbial informatics and experimentation*, 2(1):3, 2012.

15 Mikkel Thorup and Uri Zwick. Approximate distance oracles. *J. ACM*, 52(1):1–24, 2005. doi:10.1145/1044731.1044732.

16 Finn Völkel, Eric Bapteste, Michel Habib, Philippe Lopez, and Chloe Vigliotti. Read networks and k-laminar graphs. arXiv:1603.01179, Mar 2016. URL: https://hal.inria.fr/hal-01282715.