Distance Labeling Schemes for Cube-Free Median Graphs

Victor Chepoi
Aix Marseille Université, Université de Toulon, CNRS, LIS, Marseille, France
victor.chepoi@lis-lab.fr

Arnaud Labourel
Aix Marseille Université, Université de Toulon, CNRS, LIS, Marseille, France
arnaud.labourel@lis-lab.fr

Sébastien Ratel
Aix Marseille Université, Université de Toulon, CNRS, LIS, Marseille, France
sebastien.ratel@lis-lab.fr

Abstract

Distance labeling schemes are schemes that label the vertices of a graph with short labels in such a way that the distance between any two vertices u and v can be determined efficiently by merely inspecting the labels of u and v, without using any other information. One of the important problems is finding natural classes of graphs admitting distance labeling schemes with labels of polylogarithmic size. In this paper, we show that the class of cube-free median graphs on n nodes enjoys distance labeling scheme with labels of $O(\log^3 n)$ bits.

2012 ACM Subject Classification Mathematics of computing → Graph algorithms

Keywords and phrases median graphs, labeling schemes, distributed distance computation

Digital Object Identifier 10.4230/LIPIcs.MFCS.2019.15

Related Version A full version of the paper is available at https://arxiv.org/abs/1809.10508.

Funding The work on this paper was supported by ANR project DISTANCIA (ANR-17-CE40-0015).

1 Introduction

Classical network representations are usually global in nature. In order to derive a useful piece of information, one must access to a global data structure representing the entire network even if the needed information only concerns few nodes. Nowadays, with networks getting bigger and bigger, the need for locality is more important than ever. Indeed, in several cases, global representations are impractical and network representation must be distributed. The notion of (distributed) labeling scheme has been introduced [12, 32, 38, 39, 27] in order to meet this need. A *(distributed) labeling scheme* is a scheme maintaining global information on a network using local data structures (or labels) assigned to nodes of the network. Their goal is to locally store some useful information about the network in order to answer a specific query concerning a pair of nodes by only inspecting the labels of the two nodes. Motivation for such localized data structure in distributed computing is surveyed and widely discussed in [38]. The predefined queries can be of various types such as distance, adjacency, or routing. The quality of a labeling scheme is measured by the size of the labels of nodes and the time required to answer queries. Trees with n vertices admit adjacency and routing labeling schemes with size of labels and query time $O(\log n)$[1] and distance labeling schemes with size of labels and query time $O(\log^2 n)$, and this is asymptotically optimal. Finding natural classes of graphs admitting distance labeling schemes with labels of polylogarithmic size is an important and challenging problem.

[1] All logarithms in this paper are in base 2.

A connected graph G is *median* if any triplet of vertices x, y, z contains a unique vertex simultaneously lying on shortest (x, y)-, (y, z)-, and (z, x)-paths. Median graphs constitute the most important class in metric graph theory [5]. This importance is explained by the bijections between median graphs and discrete structures arising and playing important roles in completely different areas of research in mathematics and theoretical computer science: in fact, median graphs, 1-skeletons of CAT(0) cube complexes from geometric group theory [30, 41], domains of event structures from concurrency [44], median algebras from universal algebra [7], and solution sets of 2-SAT formulae from complexity theory [36, 42] are all the same. In this paper, we design a distance labeling scheme for median graphs containing no cubes. In our scheme, the labels have $O(\log^3 n)$ bits and $O(1)$ query time. Our constant query time assumes the standard word-RAM model with word size $\Omega(\log n)$.

We continue with the idea of the labeling scheme. Let $G = (V, E)$ be a cube-free median graph with n vertices. First, the algorithm computes a median (centroid) vertex m of G. and the star $\mathrm{St}(m)$ of m (the union of all edges and squares of G incident to m). The star $\mathrm{St}(m)$ is gated, i.e., each vertex of G has an unique projection (nearest vertex) in $\mathrm{St}(m)$. Therefore, with respect to the projection function, the vertex-set of G is partitioned into fibers: the fiber $F(x)$ of $x \in \mathrm{St}(m)$ consists of all vertices $v \in V$ having x as the projection in $\mathrm{St}(m)$. Since m is a median of G, each fiber contains at most $\frac{n}{2}$ vertices. The fibers are also gated and are classified into panels and cones depending to the distance between their projections and m (one for panels and two for cones). Each cone has at most two neighboring panels however a panel may have an unbounded number of neighboring cones. Given two arbitrary vertices u and v of G, we show that $d_G(u, v) = d_G(u, m) + d_G(m, v)$ for all locations of u and v in the fibers of $\mathrm{St}(m)$ except the cases when u and v belong to neighboring cones and panels, or u and v belong to two cones neighboring the same panel, or u and v belong to the same fiber. If $d_G(u, v) = d_G(u, m) + d_G(m, v)$, then $d_G(u, v)$ can be retrieved by keeping $d_G(u, m)$ in the label of u and $d_G(v, m)$ in the label of v. If u and v belong to the same fiber $F(x)$, the computation of $d_G(u, v)$ is done by recursively partitioning the cube-free median graph $F(x)$ at a later stage of the recursion. In the two other cases, we show that $d_G(u, v)$ can be retrieved by keeping in the labels of vertices in all cones the distances to their projections on the two neighboring panels. It turns out (and this is the main technical contribution of the paper), that for each panel $F(x)$, the union of all projections of vertices from neighboring cones on $F(x)$ is included in an isometric tree of G and that the vertices of the panel $F(x)$ contain one or two projections in this tree. All such outward and inward projections are kept in the labels of respective vertices. Therefore, one can use distance labeling schemes for trees to deal with vertices u and v lying in neighboring fibers or in cones having a common neighboring panel. Consequently, the size of the label of a vertex u on each recursion level is $O(\log^2 n)$. Since the recursion depth is $O(\log n)$, the vertices of G have labels of size $O(\log^3 n)$. The distance $d_G(u, v)$ can be retrieved by finding the first time in the recursion when vertices u and v belong to different fibers of the partition. Consequently, the main result of the paper is the following theorem:

► **Theorem 1.** *There exists a distance labeling scheme that constructs in $O(n^2 \log n)$ time labels of size $O(\log^3 n)$ of the vertices of a cube-free median graph $G = (V, E)$. Given the labels of u and v of G, it computes in constant time the distance $d_G(u, v)$ between u and v.*

The remaining part of this note is organized in the following way. Section 2 introduces the notions used in this paper. In Section 3 we review the main results on distance labeling schemes and on median graphs. In Section 4 we recall or establish some properties of general median graphs used in our scheme. Section 5 presents the most important geometric and structural properties of cube-free median graphs, which are the essence of our distance scheme and which do not hold for general median graphs. Section 6 describes our distance labeling scheme for cube-free median graphs and proves Theorem 1. Due to page limits, the missing proofs and the pseudocodes are provided in the full version [22]. In the full version, we also describe a routing labeling scheme with similar performances.

2 Preliminaries

2.1 Basic notions

All graphs $G = (V, E)$ in this note are finite, undirected, simple, and connected. We will write $u \sim v$ if two vertices u and v are adjacent. The *distance* $d_G(u, v)$ between two vertices u and v is the length of a shortest (u, v)-path, and the *interval* $I(u, v) := \{x \in V : d_G(u, x) + d_G(x, v) = d_G(u, v)\}$ consists of all the vertices on shortest (u, v)–paths. A connected subgraph H of G is called *isometric* if $d_H(u, v) = d_G(u, v)$ for any two vertices u, v of H. A subgraph H of G is *gated* if for every vertex $v \notin V(H)$, there exists a vertex $v' \in V(H)$ such that for all $u \in V(H)$, $d_G(v, u) = d_G(v, v') + d_G(v', u)$ (v' is called the *gate* of v in H). For a vertex x of a gated subgraph H of G, the set $F(x) = \{v \in V : x \text{ is the gate of } v \text{ in } H\}$ is called the *fiber* of x with respect to H. The fibers $\{F(x) : x \in H\}$ define a partition of G. The *m-dimensional hypercube* Q_m has all subsets of $\{1, \ldots, m\}$ as the vertex-set and $A \sim B$ iff $|A \triangle B| = 1$.

A graph G is called *median* if the intersection $I(x, y) \cap I(y, z) \cap I(z, x)$ is a singleton for each triplet x, y, z of vertices; this unique intersection vertex is called the *median* of x, y, z. Median graphs are bipartite. Basic examples of median graphs are trees, hypercubes, rectangular grids, and Hasse diagrams of distributive lattices and of median semilattices [5]. The *star* $\mathrm{St}(z)$ of a vertex z of a median graph G is the union of all hypercubes of G containing z. The *dimension* $\dim(G)$ of a median graph G is the largest dimension of an hypercube subgraph of G. A *cube-free median graph* is a median graph G of dimension 2, see Figure 1 for illustrations. Even if cube-free median graphs are the skeletons of 2-dimensional CAT(0) cube complexes, their combinatorial structure is rather intricate. As an example, for $n, m \geq 5$, the Cartesian product $K_{1,n} \times K_{1,m}$ is a non-planar cube-free median graph. Moreover, for any n, one can construct a cube-free median graph containing K_n as a minor by gluing together $\binom{n}{2}$ grids of size $n \times n$ along a common horizontal side. Hence, this class is not a subset of any minor-closed graph family.

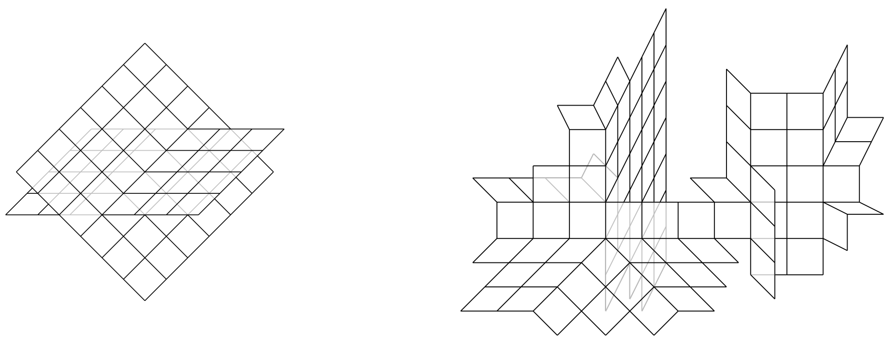

Figure 1 Cube-free median graphs.

2.2 Distance labeling schemes

A *labeling scheme* for a graph family $\mathcal{G}$ consists of an encoding function and a decoding function. These functions depend on the family $\mathcal{G}$ and on the type of queries: adjacency, distance, or routing queries. More formally, a *distance labeling scheme* on a graph family $\mathcal{G}$ consists of an *encoding function* $C_G : V(G) \to \{0,1\}^*$ that gives to every vertex of a graph G of $\mathcal{G}$ a label, and of a *decoding function* $D_G : \{0,1\}^* \times \{0,1\}^* \to \mathbb{N}$ that, given the labels of two vertices u and v of G, can compute efficiently the distance $d_G(u,v)$ between them.

3 Related work

3.1 Distance labeling schemes

Distance Labeling Schemes (DLS) have been introduced in a series of papers by Peleg et al. [38, 39, 27]. Before these works, some closely related notions already existed such as embeddings in a squashed cube [43] (equivalent to distance labeling schemes with labels of size $\log n$ times the dimension of the cube) or labeling schemes for adjacency requests [32]. One of the main results for DLS is that general graphs support distance labeling schemes with labels of size $O(n)$ bits [43, 27, 2]. This scheme is asymptotically optimal since $\Omega(n)$ bits labels are needed for general graphs. Another important result is that there exists a distance labeling scheme for the class of trees with $O(\log^2 n)$ bits labels [38, 3, 24]. Several classes of graphs containing trees also enjoy a distance labeling scheme with $O(\log^2 n)$ bit labels such as bounded tree-width graphs [27], distance-hereditary graphs [25], bounded clique-width graphs [23], and non-positively curved plane graphs [19]. A lower bound of $\Omega(\log^2 n)$ bits on the label length is known for trees [27, 3], implying that all the results mentioned above are optimal as well. Other families of graphs have been considered such as interval graphs, permutation graphs, and their generalizations [9, 26] for which an optimal bound of $\Theta(\log n)$ bits was given, and planar graphs for which there is a lower bound of $\Omega(n^{\frac{1}{3}})$ bits [27] and an upper bound of $O(\sqrt{n})$ bits [28].

3.2 Median graphs

Median graphs and related structures have an extensive literature; several surveys exist listing their numerous characterizations and properties [5, 33, 34]. These structures have been investigated in different contexts by quite a number of authors for more than half a century. In this subsection we briefly review the links between median graphs and CAT(0) cube complexes. We also recall some results, related to the subject of this paper, about the distance and shortest path problems in median graphs and CAT(0) cube complexes. For a survey of results on median graphs and their bijections with median algebras, median semilattices, CAT(0) cube complexes, and solution spaces of 2-SAT formulae, see [5]. For a comprehensive presentation of median graphs and CAT(0) cube complexes as domains of event structures, see the long version of [14].

It is not immediately clear from the definition, but median graphs are intimately related to hypercubes: median graphs can be obtained from hypercubes by amalgams and median graphs are themselves isometric subgraphs of hypercubes [8, 35]. Even more, median graphs are exactly the retracts of hypercubes [4]. Due to the abundance of hypercubes, to each median graph G one can associate a cube complex $X(G)$ obtained by replacing every hypercube of G by a solid unit cube. Then G can be recovered as the 1-skeleton of $X(G)$. The cube complex $X(G)$ can be endowed with several intrinsic metrics, in particular with the ℓ_2-metric. An important class of cube complexes studied in geometric group theory and combinatorics is the class of CAT(0) cube complexes. CAT(0) geodesic metric spaces are usually defined via the nonpositive curvature comparison axiom of Cartan–Alexandrov–Toponogov [13]. For cube complexes (and more generally for cell complexes) the CAT(0) property can be defined in a very simple and intuitive way by the property that *ℓ_2-geodesics between any two points are unique*. Gromov [30] gave a nice combinatorial characterization of CAT(0) cube complexes as *simply connected cube complexes with flag links*. It was also shown in [18, 40] that *median graphs are exactly the 1-skeletons of CAT(0) cube complexes*.

Previous characterizations can be used to show that several cube complexes arising in applications are CAT(0). Billera et al. [10] proved that the space of trees (encoding all tree topologies with a given set of leaves) is a CAT(0) cube complex. Abrams et al. [1, 29] considered the space of all possible positions of a reconfigurable system and showed that in many cases this state complex is CAT(0). Billera et al. [10] formulated the problem of computing the geodesic between two points in the space of trees. In the robotics literature, geodesics in state complexes correspond to the motion planning to get the robot from one position to another one with minimal power consumption. A polynomial-time algorithm for geodesic problem in the space of trees was provided in [37] and, very recently, [31] designed such an algorithm for all CAT(0) cube complexes.

Returning to median graphs, the following is known about the labeling schemes for them. First, the arboricity of any median graph G on n vertices is at most $\log n$, leading to adjacency schemes of $O(\log^2 n)$ bits per vertex. As noted in [21], one $\log n$ factor can be replaced by the dimension of G. Compact distance labeling schemes can be obtained for some subclasses of cube-free median graphs. One particular class is that of *squaregraphs*, i.e., plane graphs in which all inner vertices have degree ≥ 4. For squaregraphs, distance schemes with labels of size $O(\log^2 n)$ follow from a more general result of [19] for plane graphs of nonpositive curvature. Another such class of graphs is that of partial double trees [6]. Those are the median graphs which isometrically embed into a Cartesian product of two trees. The isometric embedding of partial double trees into a product of two trees immediately leads to distance schemes with labels of $O(\log^2 n)$ bits. Finally, with a technically involved proof, it was shown in [20] that there exists a constant M such that any cube-free median graph G with maximum degree Δ can be isometrically embedded into a Cartesian product of at most $\epsilon(\Delta) := M\Delta^{26}$ trees. This immediately shows that cube-free median graph admit distance labeling schemes with labels of length $O(\epsilon(\Delta)\log^2 n)$. Compared with the $O(\log^3 n)$-labeling scheme obtained in the current paper, the disadvantage of the $O(\epsilon(\Delta)\log^2 n)$-labeling scheme is the dependence from the maximum degree Δ of G. The situation is even worse for high dimensional median graphs: [20] presents an example of a 5-dimensional median graph/CAT(0) cube complex with constant degree which cannot be embedded into a Cartesian product of a finite number of trees. Therefore, for general finite median graphs the function $\epsilon(\Delta)$ does not exist. This in some sense explains the difficulty of designing polylogarithmic distance labeling schemes for general median graphs. Nevertheless, we do not have any indication to believe that such schemes do not exist.

4 Fibers in median graphs

In this section, we recall several useful properties of fibers of gated subgraphs of median graphs. From the definition, one can deduce that median graphs satisfy the following quadrangle condition: *For any vertices u, v, w, z such that $d_G(u, z) = k + 1$, $v, w \sim z$, and $d_G(u, v) = d_G(u, w) = k$, there is a unique vertex $x \sim v, w$ such that $d_G(u, x) = k - 1$.*

▶ **Lemma 1.** *[17]: A subgraph H of a median graph G is gated if and only if any vertex $v \notin V(H)$ is adjacent to at most one vertex of H.*

Combinatorially, the stars of median graphs may have quite an arbitrary structure: by a result of [8], there is a bijection (via the simplex graph operation) between the stars of median graphs and arbitrary graphs. However, from the metric point of view, stars $\mathrm{St}(z)$ have interesting properties:

▶ **Proposition 2.** *The stars $St(z)$ and their fibers $F(x), x \in St(z)$, are gated.*

Both properties of Proposition 2 are known for more general graphs: for gatedness of stars, see [15, Theorem 6.17]) and for gatedness of fibers of gated sets, see [16].

Let H be a gated subgraph of G and let $\mathcal{F}(H) = \{F(x) : x \in V(H)\}$ be the partition of V into fibers. We call two fibers $F(x)$ and $F(y)$ *neighboring* (notation $F(x) \sim F(y)$) if there exists an edge $x'y'$ of G with x' in $F(x)$ and y' in $F(y)$. If $F(x)$ and $F(y)$ are neighboring fibers of H, then denote by $\partial_y F(x)$ the set of all vertices $x' \in F(x)$ having a neighbor y' in $F(y)$ and call $\partial_y F(x)$ the *boundary of* $F(x)$ *relative to* $F(y)$. The following three results can be easily proved.

▶ **Lemma 2.** *Two fibers $F(x)$ and $F(y)$ of H are neighboring if and only if $x \sim y$. Moreover, if $F(x) \sim F(y)$, then $\partial_y F(x)$ induces a gated subgraph of G of dimension $\leq \dim(G) - 1$.*

For a vertex x of H and its fiber $F(x)$, the union of all boundaries $\partial_y F(x)$ over all $F(y) \sim F(x)$, $y \in V(H)$, is called the *total boundary* of the fiber $F(x)$ and is denoted by $\partial^* F(x)$. The boundaries $\partial_y F(x)$ constituting $\partial^* F(x)$ are called *branches* of $\partial^* F(x)$.

▶ **Lemma 3.** *The total boundary of any fiber of H is an isometric subgraph of G not containing $\dim(G)$-cubes.*

We conclude this section with an additional property of fibers of stars of *median vertices* of G, i.e., vertices minimizing the function $M(x) = \sum_{v \in V} d_G(x, v)$.

▶ **Lemma 4.** *Let m be a median vertex of a median graph G with n vertices. Then any fiber $F(x)$ of the star $St(m)$ of m has at most $n/2$ vertices.*

Unfortunately, the total boundary of a fiber does not always induce a median subgraph. Therefore, one cannot recursively apply the algorithm to the subgraphs induced by the total boundaries $\partial^* F(x)$. However, if G is cube-free, then the total boundaries of fibers are isometric subtrees of G and one can use for them distance schemes for trees. Even in this case, we still need an additional property of $\partial^* F(x)$. We establish it in the next section.

5 Fibers in cube-free median graphs

In this section, we establish additional properties of fibers and of their total boundaries in cube-free median graphs (for other properties of such graphs, see [11]). Using them we can show that for any pair u, v of vertices of G, the following trichotomy holds: the distance $d_G(u, v)$ either can be computed as $d_G(u, m) + d_G(m, v)$, or as the sum of distances from u, v to appropriate vertices u', v' of $\partial^* F(x)$ plus the distance between u', v' in $\partial^* F(x)$, or via a recursive call to the fiber containing u and v.

5.1 Classification of fibers

Let z be an arbitrary vertex of G and let $\mathcal{F}_z = \{F(x) : x \in \mathrm{St}(z)\}$ denote the partition of V into the fibers of $\mathrm{St}(z)$. We distinguish two types of fibers: the fiber $F(x)$ is called a *panel* if x is adjacent to z and $F(x)$ is called a *cone* if x has distance two to z. The interval $I(x, z)$ is the edge xz if $F(x)$ is a panel and is a square $Q_x := (x, y', z, y'')$ if $F(x)$ is a cone. In the second case, since y' and y'' are the only neighbors of x in $\mathrm{St}(z)$, by Lemma 2 we deduce that the cone $F(x)$ is adjacent to the panels $F(y')$ and $F(y'')$ and that $F(x)$ is not adjacent to any other panel or cone. By the same lemma, a panel $F(y)$ is not adjacent to any other panel, but $F(y)$ is adjacent to all cones $F(x)$ such that the square Q_x contains the edge yz.

5.2 Total boundaries of fibers are quasigated

For a set A, an *imprint* of a vertex $u \notin A$ on A is a vertex $a \in A$ such that $I(u, a) \cap A = \{a\}$. Denote by $\Upsilon(u, A)$ the set of all imprints of u on A. The most important property of imprints is that for any vertex $z \in A$, there exists a shortest (u, z)-path passing via an imprint. Therefore, if the set $\Upsilon(u, A)$ has constant size, one can store in the label of u the distances to the vertices of $\Upsilon(u, A)$. Using this, for any $z \in A$, one can compute $d_G(u, z)$ as $\min\{d_G(u, a) + d_G(a, z) : a \in \Upsilon(u, A)\}$. Note that A is gated iff any $u \notin A$ has a unique imprint on A. We will say that a set A is *quasigated* if $|\Upsilon(u, A)| \leq 2$ for any vertex $u \notin A$. The main goal of this subsection is to show that the total boundaries of fibers are quasigated.

Let T be a tree with a distinguished vertex r in G, called the *root* of T. We will say that a rooted tree T has *gated branches* if for any vertex x of T the unique path $P(x, r)$ of T connecting x to the root r is a gated subgraph of G. Lemma 3 implies:

▶ **Lemma 5.** *The total boundary of any fiber is an isometric tree with gated branches.*

By Lemma 5, $\partial^* F(x)$ has gated branches, however $\partial^* F(x)$ is not necessarily gated itself. Since a panel $F(x)$ may be adjacent to an arbitrary number of cones, one can think that the imprint-set $\Upsilon(u, \partial^* F(x))$ of a vertex u of $F(x)$ may have an arbitrarily large size. The following lemma shows that this is not the case, namely that $|\Upsilon(u, \partial^* F(x))| \leq 2$. This is one of the key ingredients in the design of the distance labeling scheme presented in Section 6. This property is no longer true for median graphs of dimension > 2.

▶ **Lemma 6.** *Any rooted (at r) tree T with gated branches of G is quasigated.*

Proof. Pick any $u \in V \setminus V(T)$ and suppose by way of contradiction that $\Upsilon(u, T)$ contains three distinct imprints x_1, x_2, and x_3. Since T has gated branches, none of the vertices x_1, x_2, x_3 belong to the path of T between r and another vertex from this triplet. In particular, r is different from x_1, x_2, x_3. Suppose additionally that among all rooted trees T' with gated branches of G and such that $|\Upsilon(u, T')| \geq 3$, the tree T has the minimal number of vertices. This minimality choice (and the fact that any subtree of T containing r is also a rooted tree with gated branches) implies that T is exactly the union of the three gated paths $P(r, x_1), P(r, x_2)$, and $P(r, x_3)$. Therefore, x_1, x_2 and x_3 are the leaves of T.

Let y_i be the neighbor of x_i in the path $P(r, x_i)$, $i = 1, 2, 3$. Since G is bipartite, either $x_i \in I(y_i, u)$ or $y_i \in I(x_i, u)$. Since $x_i \in \Upsilon(u, T)$, necessarily $x_i \in I(y_i, u)$. Let T'_i be the subtree of T obtained by removing the leaf x_i. From the minimality choice of T, we cannot replace T by the subtree T'_i. This means that $|\Upsilon(u, T'_i)| \leq 2$. Since $x_j, x_k \in \Upsilon(u, T'_i)$ for $\{i, j, k\} = \{1, 2, 3\}$, necessarily $I(y_i, u) \cap \{x_j, x_k\} \neq \varnothing$ holds.

First, notice that $x_1, x_2, x_3 \in I(u, r)$. Indeed, let z_i denote the median of the triplet x_i, u, r. If $z_i \neq x_i$, since $z_i \in I(x_i, r) = P(x_i, r) \subset T$ and $z_i \in I(u, x_i)$, we obtain a contradiction with the inclusion of x_i in $\Upsilon(u, T)$. Thus $z_i = x_i$, yielding $x_i \in I(u, z_i)$.

Now, suppose without loss of generality that $d_G(r,x_3) = \max\{d_G(r,x_i) : i = 1,2,3\} := k$. Since $I(y_3,u) \cap \{x_1,x_2\} \neq \varnothing$ as shown above, we can suppose that $x_2 \in I(y_3,u)$. Since $x_3 \in I(y_3,u)$, from these inclusions we obtain that $d_G(x_3,u)+1 = d_G(y_3,x_2)+d_G(x_2,u)$. Then $d_G(x_3,u) \geq d_G(x_2,u)$, and we conclude that $d_G(x_3,u) = d_G(x_2,u)$ and $d_G(y_3,x_2) = 1$. Since $x_2,x_3 \in I(r,u)$, $d_G(x_3,r) = d_G(x_2,r)$. We distinguish two cases:

Case 1. $d_G(x_1,r) = k$.

Since x_1,x_2,x_3 have the same distance k to r, we can apply to x_1 the same analysis as to x_3 and deduce that the neighbor y_1 of x_1 in T coincides with one of the vertices y_2 or y_3. Since $y_2 = y_3 = y$, we conclude that the vertices x_1,x_2,x_3 have the same neighbor y in T. Since y is closer to r than each of the vertices x_1,x_2,x_3 and since $x_1,x_2,x_3 \in I(r,u)$, we conclude that $x_1,x_2,x_3 \in I(y,u)$. Applying the quadrangle condition three times, we can find three vertices $x_{i,j}, i,j \in \{1,2,3\}, i \neq j$, such that $x_{i,j} \sim x_i,x_j$ and $d_G(x_{i,j},u) = k-1$. If two of the vertices $x_{1,2}, x_{2,3}$, and $x_{3,1}$ coincide, then we will get a forbidden $K_{2,3}$. Thus $x_{1,2}, x_{2,3}$, and $x_{3,1}$ are pairwise distinct. Since G is bipartite, this implies that $d_G(x_i,x_{j,k}) = 3$ for $\{i,j,k\} = \{1,2,3\}$. Since $x_{1,2},x_{2,3} \in I(x_2,u)$, by quadrangle condition there exists a vertex w such that $w \sim x_{1,2},x_{2,3}$ and $d_G(w,u) = k-2$. Since G is bipartite, $d_G(w,x_{3,1})$ equals to 3 or to 1. If $d_G(w,x_{3,1}) = 3 = d(y,w)$, then the triplet $y,w,x_{3,1}$ has two medians x_1 and x_3, which is impossible, because G is median. Thus $d_G(w,x_{3,1}) = 1$, i.e., $w \sim x_{3,1}$. Then one can easily see that the vertices $y,x_1,x_2,x_3,x_{1,2},x_{2,3},x_{3,1},w$ define an isometric 3-cube of G, contrary to the assumption that G is cube-free. This finishes the analysis of Case 1.

Case 2. $d_G(x_1,r) < k$.

This implies that $d_G(r,x_1) \leq k-1 = d_G(r,y)$. Let r' be the neighbor of r in the (r,y)-path of T. Note that $r' \notin I(r,x_1) = P(r,x_1)$. Otherwise, $r' \in P(r,x_1) \cap P(r,x_2) \cap P(r,x_3)$ and we can replace T by the subtree T' rooted at r' and consisting of the subpaths of $P(r,x_i)$ between r' and x_i, $i = 1,2,3$. Clearly T' is a rooted tree with gated branches and $x_1,x_2,x_3 \in \Upsilon(u,T')$, contrary to the minimality choice of T. Thus $r' \notin P(r,x_1)$.

Let also $P(r,x_1) = (r,v_1,\ldots,v_{m-1},v_m =: x_1)$. Note that r may coincide with y_1 and x_1 may coincide with v_1. Since $v_1,r' \in I(r,u)$, by quadrangle condition we will find $v'_2 \sim v_1,r'$ at distance $d_G(r,u)-2$ from u. Since $r' \notin I(r,x_1)$, $v'_2 \neq v_2$. Since $v_2,v'_2 \in I(v_1,u)$, by quadrangle condition we will find $v'_3 \sim v_2,v'_2$ at distance $d_G(r,u)-3$ from u. Again, since $r' \notin I(r,x_1)$, $v'_3 \neq v_3$. Continuing this way, we will find the vertices $v'_2,v'_3,\ldots,v'_m,v'_{m+1} =: x'_1$ forming an (r',x'_1)-path $P(r',x'_1)$ and such that $v'_{i+1} \sim v_i,v'_i$, $v'_{i+1} \neq v_{i+1}$, and v'_{i+1} is one step closer to u than v_i and v'_i. From its construction, $P(r',x'_1)$ is a shortest path. We assert that $P(r',x'_1)$ is gated. Otherwise, by Lemma 1, we can find two vertices v'_{i-1} and v'_{i+1} having a common neighbor z' different from v'_i. Let z be the median of the triplet z',v_{i-1},v_{i+1}. Then z is a common neighbor of z',v_{i-1},v_{i+1} and z is different from v_i (otherwise, we obtain a forbidden $K_{2,3}$). But then the vertices $v_{i-1},v_i,v_{i+1},v'_{i-1},v'_i,v'_{i+1},z,z'$ induce in G an isometric 3-cube, contrary to the assumption that G is cube-free. Consequently, $P(r',x'_1)$ is a gated path of G.

Let T'' be the tree rooted at r' and consisting of the gated path $P(r',x'_1)$ and the gated subpaths of $P(r,x_2)$ and $P(r,x_3)$ between r' and x_2,x_3, respectively. Clearly, T'' is a rooted tree with gated branches. Notice that $x'_1,x_2,x_3 \in \Upsilon(u,T'')$. Indeed, if x_2 or x_3 belonged to $I(x'_1,u)$, then $x'_1 \in I(x_1,u)$ and we would conclude that x_2 or x_3 belongs to $I(x_1,u)$, which is impossible because $x_1 \in \Upsilon(u,T)$. On the other hand, x'_1 cannot belong to $I(x_2,u)$ or to $I(x_3,u)$ because $d_G(x'_1,u) = d_G(x_1,u)-1 \leq d_G(x_2,u) = d_G(x_3,u)$. Consequently, $|\Upsilon(u,T'')| \geq 3$. Since T'' contains less vertices than T, we obtain a contradiction with the minimality choice of T. This concludes the analysis of Case 2, thus T is quasigated. ◀

Applying Lemmas 5 and 6 to the subgraph of G induced by the fiber $F(x)$, we obtain:

▶ **Corollary 3.** *The total boundary $\partial^* F(x)$ of any fiber $F(x)$ is quasigated.*

5.3 Classification of pairs of vertices

In Subsection 5.1, we classified the fibers of $\mathrm{St}(z)$ into panels and cones. In this subsection, we use it to provide a classification of pairs of vertices of G with respect to the partition into fibers, which extends the one done in [19] for planar median graphs.

Let z be an arbitrary fixed vertex of G. Let $\mathcal{F}_z = \{F(x) : x \in \mathrm{St}(z)\}$ be the partition of V into the fibers of $\mathrm{St}(z)$. Let u, v be two arbitrary vertices of G and suppose that u belongs to the fiber $F(x)$ and v belongs to the fiber $F(y)$ of $\mathcal{F}_z$. We say that u and v are *roommates* if they belong to the same fiber, i.e., $x = y$. We say that u and v are *1-neighboring* if $F(x)$ and $F(y)$ are two neighboring fibers (then one of them is a panel and another is a cone). We say that u and v are *2-neighboring* if $F(x)$ and $F(y)$ are distinct cones neighboring with a common panel, i.e., there exists a panel $F(w) \sim F(x), F(y)$. Finally, we say that u and v are *separated* if the fibers $F(x)$ and $F(y)$ are distinct, are not neighboring, and if both $F(x)$ and $F(y)$ are cones, then they are not 2-neighboring. From the definition it follows that any two vertices u, v of G are either roommates, or separated, or 1-neighboring, or 2-neighboring.

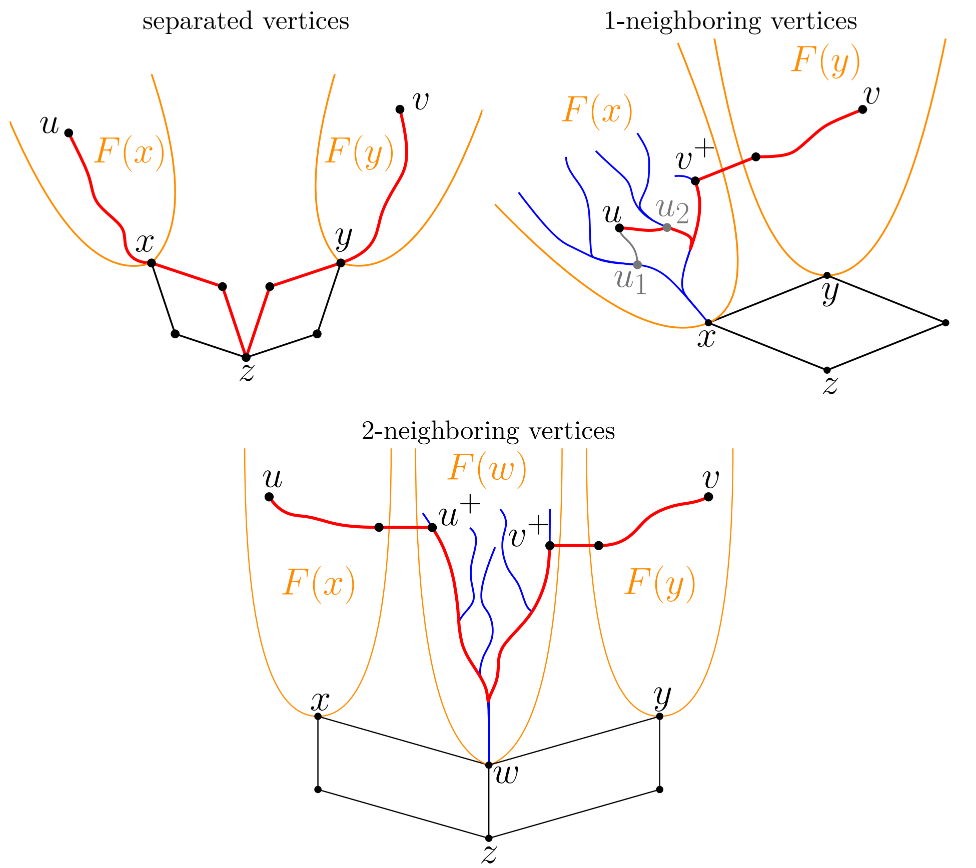

Figure 2 To Lemmas 7, 8 and 9: in red, shortest paths between separated, 1-neighboring, and 2-neighboring vertices u and v. The total boundaries of the panels appear in blue.

We continue with distance formulae for separated, 2-neighboring, and 1-neighboring vertices. The illustration of each of the formulae is provided in Figure 2.

▶ **Lemma 7.** *Two vertices u and v are separated if and only if $d_G(u,v) = d_G(u,z)+d_G(z,v)$.*

▶ **Lemma 8.** *Let u and v be two 1-neighboring vertices such that u belongs to the panel $F(x)$ and v belongs to the cone $F(y)$. Let u_1 and u_2 be the two imprints of u on the total boundary $\partial^* F(x)$ and let v^+ be the gate of v in $F(x)$. Then, $d_G(u,v) = \min\{d_G(u,u_1) + d_{\partial^* F(x)}(u_1,v^+), d_G(u,u_2) + d_{\partial^* F(x)}(u_2,v^+)\} + d_G(v^+,v)$.*

▶ **Lemma 9.** *Let u and v be two 2-neighboring vertices belonging to the cones $F(x)$ and $F(y)$, respectively, and let $F(w)$ be the panel neighboring $F(x)$ and $F(y)$. Let u^+ and v^+ be the gates of u and v in $F(w)$. Then $d_G(u,v) = d_G(u,u^+) + d_{\partial^* F(w)}(u^+,v^+) + d_G(v^+,v)$.*

6 Distance labeling scheme for cube-free median graphs

Let $G = (V,E)$ be a cube-free median graph with n vertices and let m be a median vertex of G. Let u,v be any pair of vertices of G for which we have to compute the distance $d_G(u,v)$. Applying Lemmas 7, 8, and 9 with m instead of z, the distance $d_G(u,v)$ can be computed once u and v are separated, 1-neighboring, or 2-neighboring and once u and v keep in their labels the distances to m, to the respective gates u^+ and v^+, and to the imprints u_1 and u_2 if u belongs to a panel. It also requires keeping in the labels of u and v the information necessary to compute each of the distances $d_{\partial^* F(x)}(u_1,v^+), d_{\partial^* F(x)}(u_2,v^+), d_{\partial^* F(w)}(u^+,v^+)$. Since the total boundaries are isometric trees, this can be done by keeping in the label of u the labels of u_1, u_2, and u^+ in a distance labeling scheme for trees, as well as keeping in the label of v such a label of v^+. This shows that $d_G(u,v)$ can be computed in all cases except when u and v are roommates. Since $F(x)$ is median, we can apply the same recursive procedure to each fiber $F(x)$ instead of G. Therefore, $d_G(u,v)$ is computed in the first recursive call when u and v will no longer belong to the same fiber of the current median vertex (we will sometimes refer at this median vertex as the *separator* of u and v). Since at each step the division into fibers is performed with respect to a median, $|F(x)| \leq n/2$ by Lemma 4, thus the tree of recursive calls has logarithmic depth.

In this section, we present the distance labeling scheme. The encoding scheme is described by the algorithm DIST_ENC presented in Subsection 6.1. Subsection 6.2 presents the algorithm DIST for answering distance queries. In Subsection 6.3, we briefly explain how a constant query time can be achieved by adding $O(\log^2 n)$ bits in head of each label.

6.1 Encoding

We describe now how DIST_ENC constructs for every vertex u of G a distance label $\mathrm{LD}(u)$. This is done recursively and every depth of the recursion is called a *step*. Initially, we suppose that every vertex u of G is given a unique identifier $\mathrm{id}(u)$. We define this naming step as Step 0 and denote the corresponding part of $\mathrm{LD}(u)$ by $\mathrm{LD}_0(u)$, i.e., $\mathrm{LD}_0(u) := \mathrm{id}(u)$. At Step 1, DIST_ENC computes a median vertex m of G, the star $\mathrm{St}(m)$ of m, and the partition $\mathcal{F}_m := \{F(x) : x \in \mathrm{St}(m)\}$ of V into fibers. Every vertex u of G receives the identifier $\mathrm{id}(m)$ of m and its distance $d_G(u,m)$ to m. After that, every vertex x of $\mathrm{St}(m)$ receives a special identifier $\mathrm{L}_{\mathrm{St}(m)}(x)$ of size $O(\log|V|)$ given by a distance labeling for the star $\mathrm{St}(m)$. Then, DIST_ENC computes the gate $u^{\downarrow}$ in $\mathrm{St}(m)$ of every vertex u of G and adds its identifier $\mathrm{L}_{\mathrm{St}(m)}(u^{\downarrow})$ to $\mathrm{LD}(u)$. The identifiers $\mathrm{L}_{\mathrm{St}(m)}(x)$ of the vertices of $\mathrm{St}(m)$ can also be used to distinguish the fibers of $\mathrm{St}(m)$. This triplet $(\mathrm{id}(m), d_G(u,m), \mathrm{L}_{\mathrm{St}(m)}(u^{\downarrow}))$ contains the necessary information relative to $\mathrm{St}(m)$ and is thus referred as the *part "star"* of the information $\mathrm{LD}_1(u)$ given to u at Step 1. We denote this part by $\mathrm{LD}_1^{\mathrm{St}}(u)$. We also set $\mathrm{LD}_1^{\mathrm{St[Med]}}(u) := \mathrm{id}(m)$, $\mathrm{LD}_1^{\mathrm{St[Dist]}}(u) := d_G(u,m)$ and $\mathrm{LD}_1^{\mathrm{St[gate]}}(u) := \mathrm{L}_{\mathrm{St}(m)}(u^{\downarrow})$ for the three components of the label $\mathrm{LD}_1^{\mathrm{St}}(u)$.

Afterwards, at Step 1, the algorithm considers each fiber $F(x)$ of $\mathcal{F}_m$. If $F(x)$ is a panel, then the algorithm computes the total boundary $\partial^* F(x)$ of $F(x)$. The vertices v of the quasigated tree $\partial^* F(x)$ are given special identifiers $\mathrm{LD}_{\partial^* F(x)}(v)$ of size $O(\log^2 |V|)$ consisting of a distance labeling scheme for trees (see [24]). For each vertex u of the panel $F(x)$, the algorithm computes the two imprints u_1 and u_2 of u in $\partial^* F(x)$ (it may happen that $u_1 = u_2$) and stores $(\mathrm{LD}_{\partial^* F(x)}(u_1), d_G(u, u_1))$ and $(\mathrm{LD}_{\partial^* F(x)}(u_2), d_G(u, u_2))$ in $\mathrm{LD}_1^{\mathrm{1st}}(u)$ and $\mathrm{LD}_1^{\mathrm{2nd}}(u)$. If $F(x)$ is a cone and $F(w_1)$ and $F(w_2)$ are the two panels neighboring $F(x)$, then for each vertex u of $F(x)$, the algorithm computes the gates u_1^+ and u_2^+ of u in $F(w_1)$ and $F(w_2)$, respectively. Since $u_i^+ \in \partial_x F(w_i) \subset \partial^* F(x), i = 1, 2$, the labels $\mathrm{LD}_{\partial^* F(w_1)}(u_1^+)$ and $\mathrm{LD}_{\partial^* F(w_2)}(u_2^+)$ in the distance labelings of trees $\partial^* F(w_1)$ and $\partial^* F(w_2)$ are well-defined. Therefore, the algorithm stores $(\mathrm{LD}_{\partial^* F(w_1)}(u_1^+), d_G(u, u_1^+))$ and $(\mathrm{LD}_{\partial^* F(w_2)}(u_2^+), d_G(u, u_2^+))$ in $\mathrm{LD}_1^{\mathrm{1st}}(u)$ and $\mathrm{LD}_1^{\mathrm{2nd}}(u)$. This ends Step 1.

Since $\mathcal{F}_m$ partitions V into gated median subgraphs, the label $\mathrm{LD}_2(u)$ added to $\mathrm{LD}(u)$ at Step 2 is constructed as $\mathrm{LD}_1(u)$ replacing G by the fiber $F(u^\downarrow)$ containing u, and so on. Since each fiber contains no more than half of the vertices of the current graph, at Step $\lceil \log |V| \rceil$, the fiber containing any vertex consists solely of this vertex, and the algorithm stops. Therefore, for each pair of vertices u and v of G, there exists a step of the recursion after which u and v are no longer roommates.

6.2 Distance queries

Let u and v be two vertices of G and let $\mathrm{LD}(u)$ and $\mathrm{LD}(v)$ be their labels returned by DIST_ENC. Here we describe how the algorithm DIST computes the information about the relative positions of u and v with respect to each other and how, using it, computes $d_G(u, v)$. First, the algorithm has to detect if u and v coincide or not. If $u \neq v$, then DIST finds the largest integer i such that $\mathrm{LD}_i^{\mathrm{St[Med]}}(u) = \mathrm{LD}_i^{\mathrm{St[Med]}}(v)$. This corresponds to the first time the vertices u and v belong to different fibers in a partition. Let m be the median vertex of the current median graph that is the separator of u and v. Then, the algorithm DIST retrieves the distances $d := d_G(u^\downarrow, v^\downarrow)$, $d_u := d_G(u^\downarrow, m)$ and $d_v := d_G(v^\downarrow, m)$. This is done by using the identifiers $\mathrm{LD}_i^{\mathrm{St[gate]}}(u)$ and $\mathrm{LD}_i^{\mathrm{St[gate]}}(v)$ and the distance decoder for distance labeling in stars. With this information at hand, one can easily decide for each of the vertices u and v if it belongs to a cone or to a panel, and moreover decide if the vertices u and v are 1-neighboring, 2-neighboring, or separated. In each of these cases, a call to an appropriate function is done.

First suppose that the vertices u and v are 1-neighboring ($d = 1$ and one of d_u, d_v is 1 and the other is 2), i.e., one of the vertices u, v belongs to a cone, the other one belongs to a panel, and the cone and the panel are neighboring. The function `Dist_1-Neighboring` returns the distance $d_G(u, v)$ in the assumption that u belongs to a panel and v belongs to a cone (if v belongs to a panel and u to a cone, it suffices to swap the names of the vertices u and v before using `Dist_1-Neighboring`). The function finds the gate v^+ of v in the panel of u by looking at $\mathrm{LD}_i^{\mathrm{St[gate]}}(v)$ (it also retrieves the distance $d_G(v, v^+)$). It then retrieves the imprint u^* of u (and the distance $d_G(u, u^*)$) on the total boundary of the panel that minimizes the distance of u to one of the two imprints plus the distance from this imprint to the gate v^+ using their tree distance labeling scheme. Finally, `Dist_1-Neighboring` returns $d_G(u, u^*) + d_G(u^*, v^+) + d_G(v^+, v)$ as $d_G(v, u)$.

Now suppose that the vertices u and v are 2-neighboring (i.e., $d = d_u = d_v = 2$). Then both u and v belong to cones. By inspecting $\mathrm{LD}_i^{\mathrm{St[gate]}}(u)$ and $\mathrm{LD}_i^{\mathrm{St[gate]}}(v)$, the function `Dist_2-Neighboring` determines the panel $F(w)$ sharing a border with the cones $F(u^\downarrow)$ and $F(v^\downarrow)$. Then, the function retrieves the respective gates u^+ and v^+ of u and v in this panel $F(w)$ and the distances $d_G(u, u^+)$ and $d_G(v, v^+)$. The distance between the gates u^+ and v^+ is retrieved using the distance decoder for trees. The algorithm returns $d_G(u, u^+) + d_G(u^+, v^+) + d_G(v^+, v)$ as $d_G(u, v)$.

In the remaining cases, the vertices u and v are separated. By Lemma 7, $d_G(u, v) = d_G(u, m) + d_G(m, v)$. Both u and v have stored the median vertex m and their distances to m. Therefore, `Dist_Separated` simply returns the sum of those two distances.

6.3 Complexity analysis and improved query time

The correctness of Dist results from the following properties of G: stars and fibers are gated (Proposition 2); total boundaries of fibers are quasigated (Corollary 3) isometric trees with gated branches (Lemma 5); the formulae for computing the distance between separated, 1-neighboring, and 2-neighboring vertices (Lemmas 7, 8, and 9). At each step of the encoding, $O(\log^2 n)$ bits are added to the label of every vertex (due to the tree-distance labeling scheme they contain). Since there are $\lceil \log n \rceil$ steps, the total length of each label is $O(\log^3 n)$. For decoding the labels, it suffices to read them once to find when the vertices are no longer roommates. This is done in time $O(\log^2 n)$ assuming the *word-RAM model*. Then it might be necessary to decode the distance labels for trees. This can be done in constant time [24].

To sum up, the most costly part of decoding the labels $\mathrm{LD}(u)$ and $\mathrm{LD}(v)$ is to read them up to find the (median) separator of u and v. But with an appropriate $O(\log^2 n)$ bits information concatenated to $\mathrm{LD}(u)$ and $\mathrm{LD}(v)$, one can find this median vertex in $O(1)$ time and then directly jump to the corresponding part of $\mathrm{LD}(u)$ and $\mathrm{LD}(v)$. For that, consider the tree T (of recursive calls) in which vertices at depth i are the median vertices chosen at step i and in which the children of a vertex x are the medians chosen at step $i + 1$ in the fibers generated by x at step i. We can observe that every vertex of G appears in this tree, that the separator m of any two vertices u and v of G is their nearest common ancestor in the tree T, and that its depth j in this tree corresponds to its position in $\mathrm{LD}(u)$ and $\mathrm{LD}(v)$, i.e., $\mathrm{LD}_j^{\mathrm{St[Med]}}(u) = \mathrm{LD}_j^{\mathrm{St[Med]}}(v) = \mathrm{id}(m)$. As noticed in [39], any distance labeling for trees T can be modified to support *nearest common ancestor's depth (NCAD)* queries by adding the depth $\mathrm{depth}(u)$ of u in T to the label $L(u)$ given to each vertex $u \in V(T)$ by the distance labeling. Given two vertices u and v of T, the NCAD decoder then returns $\frac{1}{2}(\mathrm{depth}(u) + \mathrm{depth}(v) - d_T(u, v))$. So, during the execution of Dist_Enc, we can also construct the tree T of recursive calls and then give an NCAD label $L'(u)$ in T to every vertex of G. Now, the first step of Dist will consist in decoding $L'(u)$ and $L'(v)$. Then the algorithm directly reads the parts of $\mathrm{LD}(u)$ and $\mathrm{LD}(v)$ corresponding to the last common median they stored. This establishes Theorem 1.

7 Conclusion

In this paper we presented a distance labeling scheme for cube-free median graphs G with labels of size $O(\log^3 n)$. For that, we considered the partitioning of G into fibers (of size $\leq n/2$) of the star $\mathrm{St}(m)$ of a median vertex m. Each fiber is further recursively partitioned using the same algorithm. We classified the fibers into panels and cones and the pairs of vertices u, v of G into roommates, separated, 1-neighboring, and 2-neighboring pairs. If u and v are roommates, then $d_G(u, v)$ is taken at a later step of the recursion. Otherwise, we showed how to retrieve $d_G(u, v)$ by keeping in the labels of u and v some distances from those vertices to some gates/imprints. Our main ingredient is the fact that the total boundaries of fibers of cube-free median graphs are isometric quasigated trees.

This last property of fibers is an obstacle in generalizing our approach to all median graphs, or even to median graphs of dimension 3. The main problem is that the total boundary is no longer a median graph. Therefore, one cannot apply to this total boundary the distance scheme for cube-free median graphs. Nevertheless, a more brute-force approach works for arbitrary median graphs G of constant maximum degree Δ. In this case, all hypercubes of G have constant size. Thus, the star $\mathrm{St}(m)$ cannot have more than $O(2^{\Delta})$ vertices, i.e., $\mathrm{St}(m)$ has a constant number of fibers. Since every fiber is gated, at every step of the encoding algorithm, every vertex v can store in its label the distance from v to its gates in all fibers of $\mathrm{St}(m)$. Consequently, this leads to distance labeling scheme with labels of (polylogarithmic) length $O(2^{\Delta}\log^3 n)$ for all median graphs with constant maximum degree Δ. We would like to finish this paper with the following question: *Does there exist a polylogarithmic distance labeling scheme for general median graphs or for median graphs of constant dimension?*

References

1. A. Abrams and R. Ghrist. State complexes for metamorphic robots. *Intl. J. Robotics Res.*, 23(7-8):811–826, 2004. `doi:10.1177/0278364904045468`.
2. S. Alstrup, C. Gavoille, E.B. Halvorsen, and H. Petersen. Simpler, faster and shorter labels for distances in graphs. In *SODA*, pages 338–350, 2016.
3. S. Alstrup, I.L. Gørtz, E.B. Halvorsen, and E. Porat. Distance Labeling Schemes for Trees. In *ICALP*, pages 132:1–132:16, 2016.
4. H.-J. Bandelt. Retracts of hypercubes. *J. Graph Theory*, 8(4):501–510, 1984. `doi:10.1002/jgt.3190080407`.
5. H.-J. Bandelt and V. Chepoi. Metric graph theory and geometry: a survey. *Contemporary Mathematics*, 453:49–86, 2008.
6. H.-J. Bandelt, V. Chepoi, and D. Eppstein. Ramified rectilinear polygons: coordinatization by dendrons. *Discr. Comput. Geom.*, 54(4):771–797, 2015. `doi:10.1007/s00454-015-9743-5`.
7. H.-J. Bandelt and J. Hedlíková. Median algebras. *Discr. Math.*, 45(1):1–30, 1983.
8. H.-J. Bandelt and M. van de Vel. Embedding topological median algebras in products of dendrons. *Proc. London Math. Soc.*, s3-58(3):439–453, 1989.
9. F. Bazzaro and C. Gavoille. Localized and compact data-structure for comparability graphs. *Discr. Math.*, 309(11):3465–3484, 2009.
10. L. J. Billera, S.P. Holmes, and K. Vogtmann. Geometry of the space of phylogenetic trees. *Adv. Appl. Math.*, 27:733–767, 2001.
11. B. Brešar, S. Klavžar, and R. Škrekovski. On cube-free median graphs. *Discr. Math.*, 307(3-5):345–351, 2007.
12. M. A. Breuer and J. Folkman. An unexpected result in coding the vertices of a graph. *J. Math. Anal. Appl.*, 20(3):583–600, 1967.
13. M.R. Bridson and A. Haefliger. *Metric Spaces of Non-Positive Curvature*, volume 319 of *Grundlehren der mathematischen Wissenschaften*. Springer-Verlag, Berlin, 1999.
14. J. Chalopin and V. Chepoi. A Counterexample to Thiagarajan's Conjecture on Regular Event Structures. In *ICALP*, pages 101:1–101:14. arXiv:1605.08288, 2017.
15. J. Chalopin, V. Chepoi, H. Hirai, and D. Osajda. Weakly modular graphs and nonpositive curvature. *Memoirs of AMS*, (to appear).
16. M. Chastand. Fiber-complemented graphs. I. Structure and invariant subgraphs. *Discr. Math.*, 226(1-3):107–141, 2001.
17. V. Chepoi. Classification of graphs by means of metric triangles. *Metody Diskret. Analiz.*, 49:75–93, 96, 1989.
18. V. Chepoi. Graphs of some CAT(0) complexes. *Adv. Appl. Math.*, 24(2):125–179, 2000.
19. V. Chepoi, F. F. Dragan, and Y. Vaxès. Distance and routing labeling schemes for non-positively curved plane graphs. *J. Algorithms*, 61(2):60–88, 2006.

20 V. Chepoi and M. F. Hagen. On embeddings of CAT(0) cube complexes into products of trees via colouring their hyperplanes. *J. Comb. Theory, Ser. B*, 103(4):428–467, 2013.

21 V. Chepoi, A. Labourel, and S. Ratel. On density of subgraphs of Cartesian products. *CoRR*, abs/1711.11485, 2017. arXiv:1711.11485.

22 Victor Chepoi, Arnaud Labourel, and Sébastien Ratel. Distance and routing labeling schemes for cube-free median graphs. *CoRR*, abs/1809.10508, 2018. arXiv:1809.10508.

23 B. Courcelle and R. Vanicat. Query efficient implementation of graphs of bounded clique-width. *Discr. Appl. Math.*, 131(1):129–150, 2003.

24 O. Freedman, P. Gawrychowski, P. K. Nicholson, and O. Weimann. Optimal distance labeling schemes for trees. In *Proceedings of the ACM Symposium on Principles of Distributed Computing*, pages 185–194. ACM, 2017.

25 C. Gavoille and C. Paul. Distance labeling scheme and split decomposition. *Discr. Math.*, 273(1-3):115–130, 2003.

26 C. Gavoille and C. Paul. Optimal distance labeling for interval graphs and related graph families. *SIAM J. Discr. Math.*, 22(3):1239–1258, 2008.

27 C. Gavoille, D. Peleg, S. Pérennes, and R. Raz. Distance labeling in graphs. *J. Algorithms*, 53(1):85–112, 2004.

28 P. Gawrychowski and P. Uznanski. A note on distance labeling in planar graphs. *CoRR*, abs/1611.06529, 2016. arXiv:1611.06529.

29 R. Ghirst and Peterson V. The geometry and topology of reconfiguration. *Adv. Appl. Math.*, 38:302–323, 2007.

30 M. Gromov. Hyperbolic groups. In S. M. Gersten, editor, *Essays in group theory*, volume 8 of *Math. Sci. Res. Inst. Publ.*, pages 75–263. Springer, New York, 1987.

31 K. Hayashi. A Polynomial Time Algorithm to Compute Geodesics in CAT(0) Cubical Complexes. In *ICALP*, pages 78:1–78:14, 2018.

32 S. Kannan, M. Naor, and S. Rudich. Implicit representation of graphs. *SIAM J. Discr. Math.*, 5(4):596–603, 1992.

33 S. Klavžar and H.M. Mulder. Median Graphs: characterizations, Location Theory and Related Structures. *J. Combin. Math. Combin. Comput.*, 30:103–127, 1999.

34 D.E. Knuth. *The Art of Computer Programming : Vol. 4. Fascicle 0, Introduction to combinatorial algorithms and Boolean functions.* Boston, Mass. ; London : Addison-Wesley, 2008. "Newly available sections of the classic work" –cover.

35 H.M. Mulder. *The Interval Function of a Graph*, volume 132 of *Mathematical Centre Tracts*. Mathematisch Centrum, Amsterdam, 1980.

36 H.M. Mulder and A. Schrijver. Median graphs and Helly hypergraphs. *Discr. Math.*, 25(1):41–50, 1979. doi:10.1016/0012-365X(79)90151-1.

37 M. Owen and J.S. Provan. A Fast Algorithm for Computing Geodesic Distances in Tree Space. *IEEE/ACM Trans. Comput. Biol. Bioinform.*, 8(1):2–13, 2011. doi:10.1109/TCBB.2010.3.

38 D. Peleg. Proximity-preserving labeling schemes. *J. Graph Theory*, 33(3):167–176, 2000. doi:10.1002/(SICI)1097-0118(200003)33:3\%3C167::AID-JGT7\%3E3.0.CO;2-5.

39 D. Peleg. Informative labeling schemes for graphs. *Theor. Comput. Sci.*, 340(3):577–593, 2005.

40 M. Roller. Poc sets, median algebras and group actions. Technical report, Univ. of Southampton, 1998.

41 M. Sageev. CAT(0) cube complexes and groups. In M. Bestvina, M. Sageev, and K. Vogtmann, editors, *Geometric Group Theory*, volume 21 of *IAS/Park City Mathematics Series*, pages 6–53. AMS, Institute for Advanced Study, 2012.

42 T.J. Schaefer. The complexity of satisfiability problems. In *STOC*, pages 216–226, 1978. doi:10.1145/800133.804350.

43 P. M. Winkler. Proof of the squashed cube conjecture. *Combinatorica*, 3(1):135–139, 1983.

44 G. Winskel and M. Nielsen. Models for Concurrency. In S. Abramsky, Dov M. Gabbay, and T. S. E. Maibaum, editors, *Handbook of Logic in Computer Science (Vol. 4)*, pages 1–148. Oxford University Press, 1995.