Distance and routing labeling schemes for non-positively curved plane graphs[1]

Victor Chepoi,[†] Feodor Dragan,[‡] and Yann Vaxès[†]

[†]Laboratoire d'Informatique Fondamentale de Marseille, Université de la Méditerranée, Faculté des Sciences de Luminy, F-13288 Marseille Cedex 9, France
{chepoi,vaxes}@lidil.univ-mrs.fr
and
[‡]Department of Computer Science, Kent State University, Kent, Ohio 44242, USA
dragan@cs.kent.edu

In memory of Chiril Prisăcaru

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 (e.g., in constant or logarithmic time) by merely inspecting the labels of u and v, without using any other information. Similarly, routing labeling schemes are schemes that label the vertices of a graph with short labels in such a way that given the label of a source vertex and the label of a destination, it is possible to compute efficiently (e.g., in constant or logarithmic time) the port number of the edge from the source that heads in the direction of the destination. In this note we show that the three major classes of non-positively curved plane graphs enjoy such distance and routing labeling schemes using $O(\log^2 n)$ bit labels on n-vertex graphs. In constructing these labeling schemes interesting metric properties of those graphs are employed.

Keywords: distance labeling scheme, routing labeling scheme, planar graphs, efficient algorithms.

1 Introduction

Traditional graph representations are global in nature, and require users to have access to data on the entire graph topology in order to derive useful information, even if the sought piece of information is very local, and pertains to only few vertices.

In contrast, the notion of *adjacency labeling scheme*, introduced in [12, 13] involves using more localized labeling schemes for graphs. The idea is to assign each vertex v a label $L(v)$ in a way that will allow one to infer the adjacency of two vertices directly from their labels, without using any additional information. Obviously, labels of unrestricted size can be used to encode any desired information. However, efficiency considerations dictate the use of relatively *short* labels (say, of length polylogarithmic in n), which nevertheless allow us to deduce adjacencies efficiently (say, in constant time). The feasibility of such efficient adjacency labeling schemes is explored in [36, 52].

Interest in this natural idea was recently revived by the observation that in addition to adjacency labeling schemes, it may be possible to device similar schemes for capturing *distance, connectivity, flow* and other information [46, 47, 38, 24, 33, 30, 55, 56, 37].

[1]Part of this work was done while the second author was visiting the Laboratoire d'Informatique Fondamentale de Marseille. This author thanks the Université de la Méditerranée for support.

1.1 Distance labeling schemes

The notion of *distance labeling schemes* was first introduced in [47], where also the relevance of distance labeling schemes in the context of communication networks was pointed out. Let us define this notion more formally. A graph family $\mathcal{F}$ is said to have an $l(n)$ distance labeling scheme if there is a function L labeling the vertices of each n-vertex graph in $\mathcal{F}$ with distinct labels of up to $l(n)$ bits, and there exists an algorithm, called *distance decoder*, that given two labels $L(v), L(u)$ of two vertices v, u in a graph from $\mathcal{F}$, decides the distance between v and u in time polynomial in the length of the given labels. Note that the algorithm is not given any additional information, other that the two labels, regarding the graph from which the vertices were taken.

As observed in [36], a class of $2^{\Omega(n^{1+\epsilon})}$ n-vertex graphs, must use adjacency labels (and thus distance labels) whose total combined length is $\Omega(n^{1+\epsilon})$. Hence, at least one label must be of $\Omega(n^{\epsilon})$ bits. Specifically, for the class of all unweighted graphs, any distance labeling scheme must label some n-vertex graphs with labels of size $\Omega(n)$. This raises the natural question of whether more efficient labeling schemes can be constructed for special graph classes.

A distance labeling scheme for trees that uses only $O(\log^2 n)$ bit labels and a constant time distance decoder has been given in [46][2] . This result is complemented by a lower bound proven in [33], showing that $\Omega(\log^2 n)$ bit labels are necessary for the class of all trees. The scheme developed for trees was later extended in [38, 33, 31] to other graph classes with "well-behaved" separators; $O(\log^2 n)$ distance labeling schemes were presented for interval graphs, permutation graphs, distance-hereditary graphs and all graphs of bounded treewidth, while an $O(\sqrt{n}\log n)$ distance labeling scheme was presented for all planar graphs. Recently, authors of [32] improved the bound on the label size given in [33] for interval graphs by a $\log n$ factor. They designed a distance labeling scheme with $O(\log n)$ bit labels and a constant time distance decoder for interval graphs and circular arc graphs. Note that for the class of planar graphs only a lower bound of $\Omega(n^{1/3})$ on the label size is known. This leaves an intriguing polynomial gap between upper and lower bounds on the label size.

Note also that another interesting question of whether more efficient labeling schemes can be constructed if one abandons the ambitions goal of capturing *exact* information and settles for obtaining *approximate* estimates is addressed in [30]. Here we will mention only a recent result for planar graphs obtained in [54]: any planar graph admits an approximate distance labeling scheme with stretch $1+\epsilon$ and $O(\log\Delta\log n/\epsilon)$ bit labels, where Δ is the graph's diameter.

Adjacency and distance labeling schemes motivated the general question of developing label-based network representations that allow retrieving useful information about arbitrary functions or substructures in a graph in a *localized* manner, i.e., using only the local information available to the vertices under inspection, and not having to search for additional *global* information. Such kind of *informative labeling schemes* are considered in [47].

[2] [46] claims to have only $O(\log n)$ time distance decoder, but in fact it is not hard to make that decoder to run in constant time.

1.2 Routing labeling schemes

Routing is one of the basic tasks that a distributed network of processors must be able to perform. A *routing scheme* is a mechanism that can deliver packets of information from any vertex of the network to any other vertex. One aims at routing along short paths. More specifically, a routing scheme is a distributed algorithm. Each processor in the network has a routing daemon running on it. This daemon receives packets of information and has to decide whether these packets have already reached their destination, and if not, how to forward them towards their destination. Each packet of information has a *header* attached to it. This header contains the address of the destination of the packet, and in some cases, some additional information that can be used to guide the routing of this message towards its destination. Each routing daemon has a local *routing table* at its disposal. It has to decide, based on this table and on the packet header only, whether to pass the packet to its host, or whether to forward the packet to one of its neighbors in the network. The *stretch* of a routing scheme is the worst ratio between the length of a path on which a message is routed and the length of the shortest path in the network from the source to the destination.

A straightforward approach for achieving the goal of guaranteeing optimal routes is to store a complete routing table in each vertex v in the network, specifying for each destination u the first edge (or an identifier of that edge, indicating the output port) along some shortest path from v to u. However, this approach may be too expensive for large systems since it requires a total of $O(n^2 \log d)$ memory bits in an n-processor network with maximum degree d. Thus, an important problem in large scale communication networks is the design of routing schemes that produce efficient routes and have relatively low memory requirements.

This problem can be approached via localized techniques based on labeling schemes [48]. Informally speaking, the routing problem can be presented as requiring us to assign a label to every vertex of a graph. This label can contain the address of the vertex as well as the local routing table. The labels are assigned in such a way that at every source vertex v and given the address of any destination vertex u, one can decide the output port of an outgoing edge of v that leads to u. The decision must be taken locally in v, based solely on the label of v and the address of u.

Following [48], one can give the following formal definition. A family $\Re$ of graphs is said to have an $l(n)$ routing labeling scheme if there is a function L labeling the vertices of each n-vertex graph in $\Re$ with distinct labels of up to $l(n)$ bits, and there exists an efficient algorithm, called the *routing decision*, that given the label of a source vertex v and the label of the destination vertex (the header of the packet), decides in time polynomial in the length of the given labels and using only those two labels, whether this packet has already reached its destination, and if not, to which neighbor of v to forward the packet.

Thus, the goal is, for a family of graphs, to find routing labeling schemes with small stretch factor, relatively short labels and fast routing decision.

To obtain routing schemes for general graphs that use $o(n)$-bit label for each vertex, one has to abandon the requirement that packets are always routed on shortest paths, and settle instead for the requirement that packets are routed on paths with relatively small stretch [49, 1, 2, 18, 21, 56]. A stretch 3 scheme that uses labels of size $\tilde{O}(n^{2/3})$ was obtained in [18], and a stretch 5 scheme that uses labels of size $\tilde{O}(n^{1/2})$ was obtained in [21].[3] Recently, authors of [56] further improved these results. They presented a routing scheme that uses

[3] Here, $\tilde{O}(f)$ means $O(f \text{ polylog } n)$.

only $\tilde{O}(n^{1/2})$ bits of memory at each vertex of an n-vertex graph and has stretch 3. Note that, each routing decision takes constant time in their scheme, and the space is optimal, up to logarithmic factors, in the sense that every routing scheme with stretch < 3 must use, on some graphs, routing tables of total size $\Omega(n^2)$, and hence $\Omega(n)$ at some vertex (see [22, 34, 28]).

There are many results on optimal (with stretch factor 1) routing schemes for particular graph classes, including complete graphs, grids (alias meshes), hypercubes, complete bipartite graphs, unit interval and interval graphs, trees and 2-trees, rings, tori, unit circular-arc graphs, outerplanar graphs, and squaregraphs (see [3, 17, 23, 26, 39, 40, 44, 45, 51]). All those graph families admit optimal routing schemes with $O(d \log n)$ labels and $O(\log d)$ routing decision. These results follow from the existence of special so called *interval routing* schemes for those graphs. In this special routing method, each vertex is assigned a distinct label from the set $\{1, \ldots, n\}$. And, for each vertex v, the incident edges are labeled with one or several subintervals of the (linear or circular) interval $[1..n]$ so that intervals associated with edges incident to v are pairwise disjoint and their union covers $[1..n]$. When a message with destination u arrives at vertex $v \neq u$, the message is forwarded on the unique outgoing edge labeled with an interval containing the label of u. We will not discuss details of this scheme here; for precise definitions and an overview of this area, we refer the reader to [27].

Observe that in interval routing schemes the local memory requirement increases with the degree of the vertex. Routing labeling schemes aim at overcoming the problem of large degree vertices. In [56], a shortest path routing labeling scheme for trees of arbitrary degree and diameter is described that assigns each vertex of an n-vertex tree a $(1 + o(1)) \log_2 n$-bit label. Given the label of a source vertex and the label of a destination it is possible to compute, in constant time, the neighbor of the source that heads in the direction of the destination. A similar result was independently obtained also in [24]. Recently, the result for trees was used in [19] in designing an interesting additive 2 stretched routing scheme for chordal graphs with labels of $O(\log^3 n / \log \log n)$ bits per vertex and $O(1)$ routing decision. For planar graphs, a routing labeling scheme with optimal stretch 1 which uses $8n + o(n)$ bits per vertex is developed in [29].

1.3 Our contribution

In this note we design efficient distance and routing labeling schemes for three natural classes of planar graphs introduced in [42, 43] and further investigated in [9, 10, 8, 50, 57] and the references cited therein. These are the basic classes of planar graphs of non-positive combinatorial curvature: (i) the plane graphs with all inner faces of length at least 4 and with all inner vertices of degree at least 4 (called the (4,4)-*graphs*), (ii) the plane graphs with all inner faces of length at least 3 and all inner vertices of degree at least 6 (called the (3,6)-*graphs*), and (iii) the plane graphs with all inner faces of length at least 6 and all inner vertices of degree at least 3 (called the (6,3)-*graphs*). Particular instances of (4,4)-graphs are the *squaregraphs* (the (4,4)-graphs in which all inner faces have length 4) [4, 11, 15, 17, 53], the planar *cellular graphs* (the bipartite (4,4)-graphs) [5], and particular instances of (3,6)-graphs are the *trigraphs* (the (3,6)-graphs in which all inner faces have length 3) investigated in [6, 15, 17]. Based on geometric properties of these graph classes, we design for them labeling schemes with labels of size $O(\log^2 n)$ bits and a constant time distance decoder and routing decision.

The paper is organized as follows. The next section presents the main definitions and notions. Section 3 describes the general lines of the method used for distance queries and routing in all three classes of graphs. In Section 4 we establish the principal distance properties of (4,4)-, (3,6)-, and (6,3)-graphs used in these schemes. We conclude with a detailed presentation of the labeling schemes from Section 3.

2 Preliminaries

All graphs $G = (V, E)$ occurring in this paper are undirected, unweighted, connected, n-vertex planar graphs embedded on the plane. The *distance* $d(u, v) := d_G(u, v)$ between two vertices u and v is the length of a shortest (u, v)-path, and the *interval* $I(u, v)$ between u and v consists of all vertices on shortest (u, v)-paths, that is, $I(u, v) = \{x \in V : d(u, x) + d(x, v) = d(u, v)\}$. An induced subgraph of G (or the corresponding vertex set A) is called *convex* if it includes the interval between any of its vertices. For a set $S \subseteq V$ and a vertex x of G, the *projection* $Pr(x, S)$ of x on S consists of all vertices $v \in S$ such that $I(v, x) \cap S = \{v\}$. Notice that $I(s, x) \cap Pr(x, S) \neq \emptyset$ for any vertex $s \in S$. A set $S \subseteq V$ is called *gated* [20] if for every vertex $x \notin S$ the projection $Pr(x, S)$ consists of a single vertex (the *gate* of x) and is called *quasigated* if for every vertex $x \notin S$ the projection $Pr(x, S)$ consists of one or two adjacent vertices of S (also called the gates of x).

For a plane graph G, let ∂G be the cycle (actually, closed walk) bounding the outer face of G oriented counterclockwise and let G^* be the *geometric dual* of G (in which vertices are defined only for inner faces of G). Notice that the class of (4,4)-graphs is self-dual in the sense that the dual of a (4,4)-graph is again a (4,4)-graph, while the classes of (3,6)- and (6,3)-graphs are mutually dual. Two neighbors x, y of a vertex v of G are called *consecutive* if u, x, y belong to a common inner face of G. Following [41, 43, 9, 10] and the references therein, we introduce now the curvature function of a plane graph G. Assume that each inner face with k sides of G is replaced by a regular k-gon in Euclidean plane with side length 1, thus yielding a cell complex $||G||$. For a vertex v of G, let $\alpha(v)$ denote the sum of the corner angles of the polygons of $||G||$ containing the vertex v. If v is an inner vertex of G, denote the *curvature* at v to be $\kappa(v) = 2\pi - \alpha(v)$, i.e., it is defined as the 2π-angle-defect of the flat polygons meeting at v. When v is a vertex in the boundary ∂G, define the *turning angle* at v to be $\tau(v) = \pi - \alpha(v)$. A vertex $v \in \partial G$ with $\tau(v) > 0$ is called a *corner* of G. The following Lyndon's curvature theorem [43] is a discrete version of the Gauss-Bonnet theorem and holds for all plane graphs:

$$\sum_{v \in V - \partial G} \kappa(v) + \sum_{v \in \partial G} \tau(v) = 2\pi.$$

A plane graph G has *non-positive curvature* if $\kappa(v) \leq 0$ for every inner vertex v of G. It can be easily shown that the plane graphs of each of the types (4,4), (3,6), and (6,3) have non-positive curvature, and from this perspective they have been investigated in a number of papers; cf. for example [9, 10, 42, 43, 57]. From the Gauss-Bonnet formula it follows that a plane graph of non-positive curvature has at least 3 corners. In Section 4, we will further specify this property for each type of those graphs.

For an edge uv of a graph G, define the following partition of the vertex set V:

$$W(u, v) = \{x \in V : d(x, u) < d(x, v)\},$$

$$W(v,u) = \{x \in V : d(x,v) < d(x,u)\},$$
$$W_=(uv) = \{x \in V : d(x,v) = d(x,u)\}.$$

If G is bipartite, then the set $W_=(uv)$ is empty. A *cut* $\{A,B\}$ of G is a partition of the vertex-set V into two parts, and a *convex cut* is a cut in which the halves A and B are convex. Denote by $E(A,B)$ the set of all edges of G having one end in A and another one in B, and say that those edges are crossed (or cut) by $\{A,B\}$. The *zone* $Z(A,B)$ of the cut $\{A,B\}$ is the subgraph induced by the union of all inner faces of G sharing edges with $E(A,B)$ and call the subgraphs induced by $\partial A = Z(A,B) \cap A$ and $\partial B = Z(A,B) \cap B$ the *borders* of the cut $\{A,B\}$. A zone $Z(A,B)$ is called a *strip* if $||Z(A,B)||$ is a simply connected region of the plane (in particular, $Z(A,B)$ induces a path in the dual graph G^*).

We continue with the definition of alternating cuts introduced and investigated in [14, 50]. Two edges $e' = (u',v')$ and $e'' = (u'',v'')$ on a common inner face F of G are called *opposite* in F if $d_F(u',u'') = d_F(v',v'')$ and equals the diameter of the cycle F. If F is an even face, then any its edge has an unique opposite edge, otherwise, if F is an odd face, then every edge $e \in F$ has two opposite edges e^+ and e^- sharing a common vertex. In the latter case, if F is oriented clockwise, for e we distinguish the *left opposite edge* e^+ and the *right opposite edge* e^-. If every face of $Z(A,B)$ is crossed by a cut $\{A,B\}$ in two opposite edges, then we say that $\{A,B\}$ is an *opposite cut* of G. We say that an opposite cut $\{A,B\}$ is *straight* on an even face $F \in Z(A,B)$ and that it *makes a turn* on an odd face $F \in Z(A,B)$. The turn is *left* or *right* depending which of the pairs $\{e,e^+\}$ or $\{e,e^-\}$ it crosses. An opposite cut $\{A,B\}$ of a plane graph G is *alternating* if the turns on it alternate. For a graph G of one of the three types (4,4), (3,6), or (6,3), denote by $\mathcal{AC}(G)$ the collection of alternating cuts of G (if G is not bipartite, then every alternating cut whose zone consists of even faces only is considered in $\mathcal{AC}(G)$ twice).

Finally, for a graph $G = (V,E)$ and a vertex x, let $F(x) = \sum_{v \in V} d(x,v)$. Any vertex minimizing the function F is called a *median vertex* of the graph G. Notice the following simple but important property of the function F : *if uv is an edge of G, then* $F(v) - F(u) = |W(u,v)| - |W(v,u)|$. From this we immediately conclude that if v is a median vertex of G, then $|W(u,v)| \leq |V|/2$ for any neighbor u of v.

3 General method

Our distance and routing labeling schemes are based on geometric properties of alternating cuts of (4,4)-, (3,6)-, and (6,3)-graphs G (some of them have been already proven in [8, 50] in order to establish the scale 2 embedding of these graphs into hypercubes). First we prove that any alternating cut $\{A,B\}$ of G is convex, moreover its borders are convex paths (thus the zone of every such cut is a strip sharing two edges with ∂G). Then, in case of (4,4)-graphs, we show that the zones of alternating cuts are quasigated (in cellular graphs, in particular, in squaregraphs, the zones are gated). For (3,6)-graphs, we show that the projections on zones of alternating cuts are convex paths whose vertices have the same distance to x. Finally, for (6,3)-graphs, we show that if $x \in A$, then $Pr(x, Z(A,B))$ either consists of one or two adjacent vertices or of a sequence of vertices of ∂A such that any two consecutive vertices are at distance 2 from each other and all vertices of $Pr(x, Z(A,B))$ except the leftmost and the rightmost have the same distance to x while those two end vertices may be one step further from x. In all cases, the projection $Pr(x, Z(A,B))$ can be compactly represented by the end vertices and the type of this projection. For example, for (4,4)-graphs, at vertex x we will keep the gate(s) on $Z(A,B)$ and the distance from x to the projection. For (3,6)-graphs, we will keep the end vertices of the convex path $Pr(x, Z(A,B))$ and the distance from x to the projection. For (6,3)-graphs, additionally to the end vertices of $P(x, Z(A,B))$, we will keep the information about the second leftmost and/or the second rightmost vertices of this projection provided there are two values for distances from x to the vertices of $P(x, Z(A,B))$. For routing messages from x the following property of $Pr(x, Z(A,B))$ is crucial: we show that either there is a neighbor of x one step closer to $Z(A,B)$ whose projection on this zone coincides with that of x or there exist two neighbors of x one step closer to the zone and whose projections on $Z(A,B)$ cover the projection of x. We will keep at x the information about such neighbors and use it in the routing decision. So, in all cases we need only $O(\log n)$ bits to store at x the entire information about the relative position of x with respect to the zone $Z(A,B)$. Therefore, we can report in constant time the distance between two vertices $x \in A$ and $y \in B$ using only the information related to $Z(A,B)$ stored at x and y (for this we design also an $O(1)$-time algorithm for computing the distance between two projections on $Z(A,B)$). However, we need more information in order to compute the distances between two vertices of A or two vertices of B.

To compute the distances or a routing shortest path between arbitrary two vertices of G, we describe a distributed data structure which at each vertex $x \in V$ keeps the projections of x on the zones of only $O(\log_2 n)$ alternating cuts of G. For this, let v be a median vertex of G and let $u_0, u_1, \ldots, u_{k-1}$ be its neighbors in counterclockwise order around v, according to the embedding of G in the plane. (We may assume without loss of generality that v is an inner vertex of G, otherwise we can add a constant number of vertices and faces around v to transform it into an inner vertex and obtain a graph of the same type.) Every edge vu_i is crossed by two alternating cuts $\{A'_i, B'_i\}$ and $\{A''_i, B''_i\}$ such that $v \in A'_i \cap A''_i$ and $u_i \in B'_i \cap B''_i$. Let us orient the cuts $\{A'_i, B'_i\}$ and $\{A''_i, B''_i\}$ such that v is on the left borderline. In this case, we will denote by $\{A'_i, B'_i\}$ that cut from the two alternating cuts separating u_i and v, such that the last turn before $u_i v$ is on the right (if it exists) and the next turn after $u_i v$ is on the left (if it exists). If none of these two turns exists, then $\{A'_i, B'_i\}$ and $\{A''_i, B''_i\}$ coincide. For each vertex u_i, set $C_v(u_i) = B'_i \cap A'_{i+1 (mod\ k)}$ and call $C_v(u_i)$ a *cone with apex* u_i. Every cone is convex as the intersection of two convex sets. We show that $C_v(u_i) \subseteq W(u_i, v)$, yielding $|C_v(u_i)| \leq n/2$ because v is a median vertex. Furthermore, we establish that the cones $C_v(u_0), \ldots, C_v(u_{k-1})$ together with the vertex v form a partition of the vertex-set of G. To report the distance or a routing path between two query vertices x and y efficiently, yet another property of the partition $C_v(u_0) \cup \ldots \cup C_v(u_{k-1}) \cup \{v\}$ is significant. We call two neighbors u_i, u_j of v *p-consecutive* and their cones $C_v(u_i), C_v(u_j)$ *p-neighboring* if $\min\{|i-j|, k-|i-j|\} = p$. Let $x \in C_v(u_i)$ and $y \in C_v(u_j)$. We show that if $C_v(u_i)$ and $C_v(u_j)$ are not p-neighboring for $p \leq 2$, then $d(x,y) = d(x,v) + d(v,y)$ (for squaregraphs a similar property with $p \leq 1$ holds). Therefore, in order to report distance between two vertices x and y in different cones, we have to keep their distances to the median vertex v, the projections on and the distances to the 1-neighboring and 2-neighboring cones, more precisely on/to the zones separating the respective cones. Finally, if x and y belong to the same cone $C_v(u_i)$, then the distance $d(x,y)$ can be retrieved by recursively decomposing the (convex) subgraph G_i induced by $C_v(u_i)$. Routing between x and y can be performed by converting the distance labeling scheme in the following way. To route a message from x to y lying in different cones, instead of distances, we have to store in the label of x the output port number of the first edge on a shortest path from x to v and the output port number of the first edge on a shortest path from x to a certain vertex (which will be further specified) of the projection of x on the 1-neighboring and 2-neighboring (for $(3,6)$ and $(6,3)$-graphs) cones, or more precisely on the zones separating the respective cones. If x is its own projection on the zone between x and y, we consider the relative position of x and the projection of y on the zone to decide in constant time via which edge the message should be sent. Finally, if x and y belong to the same cone $C_i(v)$, then, again, the routing can be done by recursively decomposing the subgraph G_i induced by this cone.

All these facts suggest the necessity of building a decomposition tree $T(G)$ of G, which can be constructed in the following way. Find a median vertex v of G and the cones $C_v(u_0),\ldots,C_v(u_{k-1})$ with apices at the neighbors of v. Let G_i be the subgraph of G induced by the vertices of $C_v(u_i)$. For each G_i construct a decomposition tree $T(G_i)$ recursively and build $T(G)$ by taking pair (G,v) to be the root and connecting the root of each tree $T(G_i)$ as a child of (G,v). It is easy to see that a decomposition tree $T(G)$ of a graph G with n vertices has depth at most $\log_2 n$ and can be constructed in $O(n^2\log n)$ time (for squaregraphs and trigraphs we need only $O(n\log n)$ time because a median vertex in these graphs can be found in linear time). Indeed, in each level of recursion we need to find median vertices of current subgraphs and to construct the corresponding cones. Also, since the graph sizes are reduced by a factor $1/2$, the recursion depth is $O(\log n)$.

For tree $T(G)$ we need a labeling scheme for depths of nearest common ancestors (*NCA-depth labeling scheme*). In [47] such a scheme with $O(\log^2 n)$ bit labels but with $O(\log n)$ query time was presented for any tree with n nodes. One can use here the fact that $T(G)$ has the $O(\log n)$ depth and get constant query time in this case. To do this one can simply translate the technique of Harel and Tarjan [35] to a labeling scheme. Note that whenever they access global information, it is associated with an ancestor in a tree. Since the depth of our tree is $O(\log n)$, one can copy this ancestor information down to each descendant and get the desired label of $O(\log^2 n)$ bits. Thus, tree $T(G)$ can be preprocessed in $O(n\log n)$ time for depths of nearest common ancestors. This preprocessing step creates for $T(G)$ an NCA-depth labeling scheme with $O(\log^2 n)$ bit labels and constant query time. For each vertex x of a graph G, let $S(x)$ be the deepest node of $T(G)$ containing x and A_x be the label of $S(x)$ in the NCA-depth labeling scheme. Let also $S_0,S_1,\ldots,S_h$ be the nodes of the path of $T(G)$ from the root (G,v) (which is S_0) to the node $S(x)=S_h$. Clearly, $h\le\log_2 n$.

In our distance (or routing) labeling scheme, vertex x will keep in its label $L(x)$ the string A_x and $O(\log_2 n)$ strings of $O(\log n)$ bits, one for each node S_i $(i\in\{0,\ldots,h\})$. The string for $S_i=(G_i,v_i)$ will contain the distance (or routing) and projection information obtained during the decomposition of a subgraph G_i using its median vertex v_i. To report the distance between vertices x and y of G (or to route a message from x to y), we can do the following. First, using strings A_x and A_y, find the depth in $T(G)$ of the nearest common ancestor $S_k=(G_k,v_k)$ of $S(x)$ and $S(y)$. Clearly, vertices x and y belong to different cones defined by v_k in G_k. Therefore, one can apply the method described above to compute $d_G(x,y)$ (or the port number of an edge incident to x which heads in the direction of y) in constant time using only the strings in $L(x)$ and $L(y)$ which correspond to the node S_k of $T(G)$.

In Section 5, this general method will be described in details for each of the three graph classes.

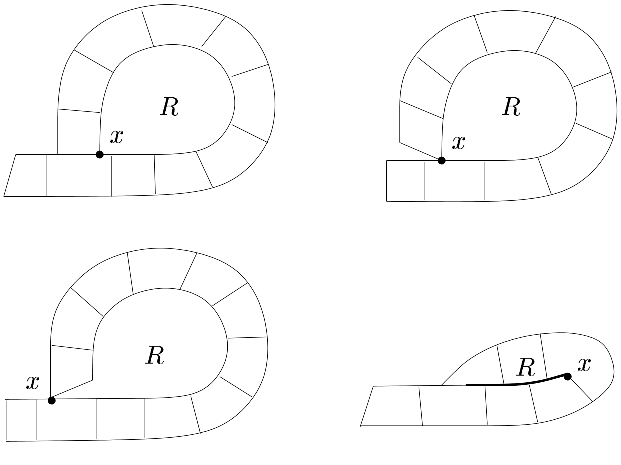

Figure 1: To the proof of Lemma 4.2.

4 Geometry of (4,4)-, (3,6)-, and (6,3)-graphs

Here we establish the metric and structural properties of (4,4)-, (3,6)-, and (6,3)-graphs used in the distance and routing labeling schemes described in Section 3 and detailized in Section 5. In the following results, unless specified, G is a plane graph of one of those types.

4.1 Alternating cuts

We start with a result first established by Lyndon [42] for the three classes of plane graphs in question and later extended to all plane graphs of non-positive curvature by Baues and Peyerimhoff [9].

Lemma 4.1 [9, 42] *For each vertex x of a plane graph G of non-positive curvature, all vertices at maximum distance from x are located on the outer face ∂G.*

As we noticed in Section 2, every plane graph G of non-positive curvature has at least 3 corners. If G is a (4,4)-graph, then every corner of G is a vertex of degree 2, and from the Gauss-Bonnet formula we conclude that in fact such a graph G must contain at least 4 corners. In a similar way, one concludes that a (3,6)-graph either contains exactly 3 corners which are vertices of degree 2 incident to inner faces of length 3 or at least 4 corners. In the latter case, the corners are either vertices of degree 2 or vertices of degree 3 incident to two inner faces, one of length 3 and another of length at most 5. Finally, in a (6,3)-graph all corners are vertices of degree 2 and G contains at least 6 corners. The following sharper version of this result established in [8] will be of more use: *∂G contains at least 6 edges whose end vertices are corners;* we call them *corner edges* (this again follows easily from the Gauss-Bonnet formula).

We continue with the properties of alternating cuts of G. In the following results, $\{A, B\}$ is an alternating cut of the graph G. Several lemmata have been proven in [8] for (3,6)- and (6,3)-graphs. For (4,4)-graphs, the analogies of some of those results have been established in [50]. For the sake of completeness we provide all results with proof.

Lemma 4.2 [8, 50] *$Z(A,B)$ is a strip.*

PROOF. Suppose, by way of contradiction, that $Z(A,B)$ is self-intersecting. Then we will obtain one of the four configurations depicted in Figure 1. In the first three cases, consider the subgraph H of G induced by all vertices lying in the bounded region R. Obviously H has the same type as G. If G is a (4,4)-graph, then all vertices of ∂H except one must have degree larger that two, otherwise we get an inner vertex of degree 2 or 3 in G. However this contradicts the fact that H must have at least 4 corners. Analogously, if G is a (6,3)-graph, H cannot contain any corner edge. If G is a (3,6)-graph, any corner of H different from x will be an inner vertex of G of degree at most 5, again leading to a contradiction. In the fourth case, the region R degenerates into a path, and one of the end vertices of this path (namely, the vertex x) is an inner vertex of G having degree 2. □

Lemma 4.3 [8] *The border lines of an alternating cut are convex paths. In particular, the alternating cuts of G and their zones are convex.*

PROOF. From Lemma 4.2 we conclude that ∂A and ∂B are paths. Therefore, it suffices to establish that they are convex. Assume the contrary and among alternating cuts with nonconvex border lines pick an alternating cut $\{A,B\}$ such that ∂A contains a closest pair of vertices x and y which can be connected by a shortest (x,y)-path P such that $P \cap \partial A = \{x,y\}$. Since the lengths of the subpaths of ∂A and ∂B, comprised between the end vertices of two edges of $E(A,B)$, differ by at most 1 (because the cut $\{A,B\}$ is alternating), necessarily the whole path P must belong to the set A. Otherwise, we can replace $x,y \in \partial A$ with a pair of vertices of $\partial B \cap P$, contrary to our choice. Let z be a neighbor of x on the path P. Consider the alternating cuts $\{A',B'\}$ and $\{A'',B''\}$ which cross the edge xz. If one of these cuts, say $\{A',B'\}$, crosses another edge $x'y'$ of P, where $z,y' \in A'$, then by replacing $x,y \in \partial A$ with the pair $z,y' \in \partial A'$, we will get a contradiction with the choice of x,y. Thus, we may assume that both these alternating cuts separate some adjacent vertices u,v of the path ∂A, say $x,u \in \partial A'$ and $z,v \in \partial B'$. We will obtain one of the situations depicted in Figure 2. In the first case, let H be the subgraph of G comprised in the region R. Let t be the closest to u common vertex of ∂A and $\partial A'$. If G is a (4,4)-graph, then H may contain only two corners t and u. If G is a (6,3)-graph, then H may contain at most four corner edges which are all incident to t or to u. Finally, if G is a (3,6)-graph, then any corner w of H other than t and u will be an inner vertex of G having degree at most 5: if, say $w \in \partial A'$, then w has maximum 4 neighbors in the zone $Z(A',B')$ and yet another neighbor located in the interior of the region R. This contradicts the fact that H must contain at least three corners. Now consider the second possibility from Figure 2. If G is a (4,4)- or a (6,3)-graph, then from the definition of an alternating cut one concludes that u cannot have other neighbors in $\partial B'$ except v. Since u can have at most one neighbor in ∂B, u is an inner vertex of G of degree 2 or 3, which is impossible if G is of type (4,4). But, if u has degree 3, then necessarily the inner face of $Z(A,B)$ containing the edge uv is either of length 4 or 5, yielding a contradiction if G is of type (6,3). Finally, if G is of type (3,6), since ∂A and $\partial A'$ share the subpath between u and x, one can easily deduce that u is an inner vertex of G of degree at most 5, a contradiction. This shows that the border lines of alternating cuts of G are convex paths, from which we infer that the alternating cuts of G and their zones are convex, too. □

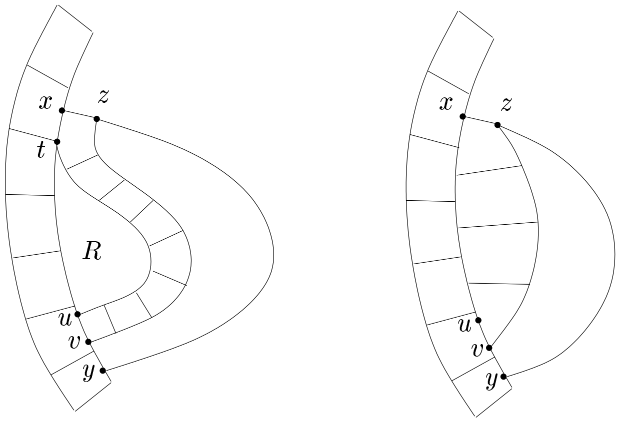

Figure 2: To the proof of Lemma 4.3.

Since every edge of G is crossed by exactly two alternating cuts from $\mathcal{AC}(G)$, previous lemma implies that the graphs of types (4,4), (3,6), and (6,3) have a scale 2 embedding into a hypercube (i.e., for any pair of vertices x, y of G, $2d(x,y)$ equals the number of cuts of $\mathcal{AC}(G)$ separating the vertices x and y), a result established in [8] for (3,6)- and (6,3)-graphs and in [50] for (4,4)-graphs.

Let xy be an edge of G and let $\{A', B'\}$ and $\{A'', B''\}$ be the (not necessarily distinct) alternating cuts crossing xy, where $x \in A' \cap A''$ and $y \in B' \cap B''$. We will establish now a relation between A', B', A'', B'' and the sets $W(x,y), W(y,x), W_=(xy)$ (the third set here may be empty). By removing the edges of $E(A', B') \cup E(A'', B'')$ from G but leaving their end vertices, we get a graph whose connected components are induced by the pairwise intersections $A' \cap A'', B' \cap B'', A' \cap B''$, and $A'' \cap B'$. We assert that these convex sets coincide with $W(x,y), W(y,x)$ and the connected components of $W_=(xy)$. First notice that from the definition of alternating cuts and convexity of their border lines it follows that $Z = Z(A', B') \cap Z(A'', B'')$ consists of one or several faces constituting a strip. Notice that each of the end faces of Z either shares an edge with the outer face of G or is an odd face. Denote by F and D these odd faces if they exist. Notice that all other faces of Z have even length. Let uv and wz be the first edges of F and D cut by $\{A', B'\}$ and $\{A'', B''\}$ while moving from xy towards these faces, and assume that $u, w \in A' \cap A''$ and $v, z \in B' \cap B''$. Let pr' and pr'' be the opposite to uv edges of F and let qt' and qt'' be the opposite to wz edges of D. Assume that the cut $\{A', B'\}$ makes turns at the edges pr'' and qt' such that $p, t' \in \partial A'$ and $r'', q \in \partial B'$. Then, the cut $\{A'', B''\}$ makes turns at the edges pr' and qt'' such that $r', q \in \partial A''$ and $p, t'' \in \partial B''$; see Figure 3 for an illustration. Since the zones $Z(A', B'), Z(A'', B'')$ and their border lines are convex, we conclude that all vertices of $\partial A'$ are closer to x than to y, all vertices of $\partial B''$ are closer to y than to x, while the vertices of the subpaths L', L'' of $\partial B'$ and $\partial A''$, comprised between q and their end vertices of ∂G, are equidistant from x and y as well as the vertices of the subpaths P', P'' of $\partial A'$ and $\partial B''$, comprised between p and their end vertices of ∂G. Now, if we pick a vertex z in $B' \cap A''$, then any shortest path between z and x or y must cross one of the paths L' or L'', therefore z is equidistant from x and y. Analogously, all vertices of $A' \cap B''$ are equidistant from x and y, while all vertices of $A' \cap A''$ are closer to x than to y and all vertices of $B' \cap B''$ are closer to y than to x. Since any path connecting vertices from different convex sets $A' \cap A'', A' \cap B'', B' \cap A'', B' \cap A''$ necessarily employs an edge of $E(A', B') \cup E(A'', B'')$, these sets are the connected components of the graph obtained from G by removing the edges of $E(A', B') \cup E(A'', B'')$. If both F and D do not exist, then the cuts $\{A', B'\}$ and $\{A'', B''\}$ coincide, and if only one of the faces F, D exists, then $W_=(xy)$ consists of a single convex component. Summarizing, we obtain the following result:

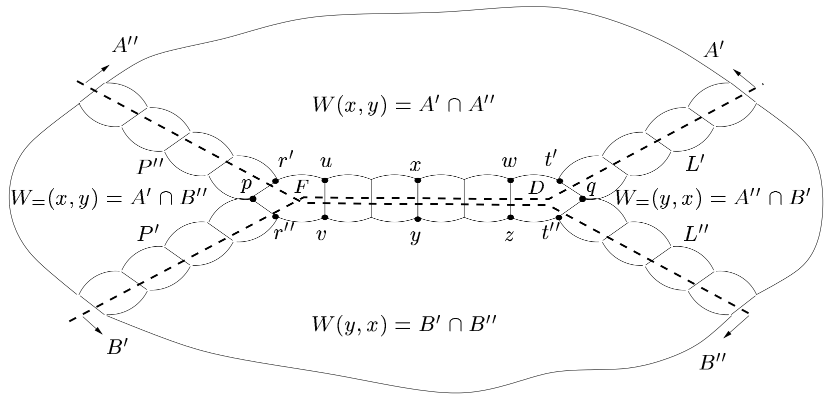

Figure 3: Relationship between the sets A', B', A'', B'' and the sets $W(x,y), W(y,x), W_=(xy)$.

Lemma 4.4 *$W(x,y) = A' \cap A'', W(y,x) = B' \cap B''$, while $W_=(x,y) := B' \cap A''$ and $W_=(y,x) := A' \cap B''$ constitute a partition of $W_=(xy)$ into two (maybe empty) convex subsets.*

From the previous discussion we also conclude that every vertex $z' \in W_=(x,y)$ can be connected to x and y by shortests path going via p. Moreover, p is the furthest from z' vertex of $I(z', x) \cap I(z', y)$. Following [7], we call p the *apex* of z' with respect to x and y. Analogously, one can define the apex of every vertex $z'' \in W_=(y,x)$ and see that it coincides with the vertex q (see Figure 3).

4.2 Faces

We continue with some properties of inner faces of G.

Lemma 4.5 *The intersection of any two inner faces of G is either empty, or a vertex, or an edge.*

Proof. Let F' and F'' be two intersecting inner faces. Then $F' \cap F''$ cannot contain paths of length 2, because all inner vertices must have degree at least 3. Thus, $F' \cap F''$ consists of a sequence of isolated vertices and edges. Let R be the region bounded by two paths of F' and F'', respectively, comprised between two consecutive non-adjacent common vertices. The subgraph H of G comprised in R has the same type as G but does not contain vertices of degree 2 at all. This leads to a contradiction for the case of (4,4)- and (6,3)-graphs. If G is a (3,6)-graph, any corner of degree 3 of H is an inner vertex of degree 3 or 4 of G, which is impossible. This establishes our assertion. □

Lemma 4.6 [8] *Every inner face F of G is convex.*

PROOF. Assume, by way of contradiction, that two (necessarily non adjacent) vertices u, v of F can be connected by a shortest path P outside F, i.e., $P \cap F = \{u, v\}$. From Lemma 4.3 we conclude that u and v will be separated by any alternating cut $\{A, B\}$ which crosses the face F. This cut crosses two edges $u'v'$ and $u''v''$ of F, where $u', u'' \in A$, $v', v'' \in B$ and the edge $u'v'$ is located in the interior of the region bounded by P and the subpath of F comprised between u and v. Moreover, $\{A, B\}$ cuts the path P in some edge xy, where $x \in A$ and $y \in B$. Since $v'', v', y \in \partial B$, from Lemma 4.3 we know that there exists a unique shortest path between v'' and y and this path must pass via the vertices v and v', yielding $v = v'$. Analogously, one concludes that $u = u'$, contrary to the choice of u and v. □

Lemma 4.7 *Every inner face F of G is quasigated.*

PROOF. Suppose there exist vertices of G whose projections on F contain non-adjacent vertices and among such triplets select the vertices $x \notin F$, $u, v \in Pr(x, F)$ minimizing the distance sum $d(x, u) + d(u, v) + d(v, x)$. Pick shortest (u, x)- and (v, x)-paths P' and P''. Let Q' and Q'' be the subpaths of F comprised between u and v, and assume, without loss of generality, that Q' is located in the interior of the region bounded by the closed walk $P' \cup P'' \cup Q''$. Let w be the neighbor of u in Q'. Then $d(x, w) \geq d(x, u)$, because $u \in Pr(x, F)$. Since every alternating cut separating the vertices w and x either separates u from x or u from w, and there exists exactly $2d(w, x) (\geq 2d(u, x))$ such cuts, we conclude that there exists a cut $\{A, B\} \in \mathcal{AC}(G)$ such that $u, x \in A$ and $w, v \in B$. This cut separates two adjacent vertices $p, q \in P''$ and two adjacent vertices $u', w' \in Q''$, where $u', p \in A$ and $v', q \in B$. Notice that $u', u, p \in \partial A$ and $v', w, q \in \partial B$. Since ∂B is convex by Lemma 4.3, the vertices v and w lie on this path. Therefore, $w \in I(v, q) \subset I(v, x)$, contrary to the assumption that $I(x, v) \cap F = \{v\}$. This establishes that the vertices of $Pr(x, F)$ are pairwise adjacent. For graphs of type (4,4) or (6,3) we immediately obtain that $Pr(x, F)$ consists of one or two vertices. If G is of type (3,6), however $|Pr(x, F)| > 2$, then F is 3-cycle (u, v, w) whose vertices have the same distance to x. One of these vertices, say u, is located in the region R bounded by the edge vw and two shortest paths connecting v, w with x. Consider the subgraph (of type (3,6)) of G induced by the vertices lying inside R or on the boundary of R. By Lemma 4.1, all inner vertices of this graph must be closer to x than v and w, which is not the case for u. □

Lemma 4.8 *Given three vertices x, y, z such that $d(x, z) = d(y, z)$ and $d(x, y) = 1$, either x, y, z belong to a common odd face or there exists an even face F such that the edges xy and $z'z''$ are opposite in F, where $Pr(z, F) = \{z', z''\}$.*

PROOF. Let $z \in W_{=}(x, y)$; see Figure 3. Then the required face is F if the edge xy belongs to F or the face of Z incident to xy and comprised between xy and F. □

4.3 Intervals

The following property of intervals will be of much use.

Lemma 4.9 *The vertex x has at most two (consecutive) neighbors in the interval $I(x, y)$. If G is a (6,3)-graph, then $I(x, y)$ contains at most two vertices at distance 2 from x. Moreover,* *if x has two neighbors and two vertices at distance 2 in $I(x,y)$, then these five vertices belong to a common inner face of G.*

PROOF. We proceed by induction on $d(x,y)$. Pick two neighbors u,w of x in $I(x,y)$ and let P' and P'' be two shortest (x,y)-paths passing via u and w. From Lemma 4.1 applied to the subgraph of G induced by all vertices lying on $P'\cup P''$ or inside the region R bounded by the closed walk $P'\cup P''$, we conclude that any neighbor $v\in R$ of x also belongs to $I(x,y)$. So, further assume that $I(x,y)$ contains three vertices u,v,w such that each of the triplets x,u,v and x,v,w lies on common inner faces F' and F'' of G. Then $F'\cap F''=\{x,v\}$. If F' and F'' have length ≥ 4, let v' and v'' be the (different) neighbors of v in the faces F' and F''. Since $v',v''\in I(v,y)$ in view of Lemma 4.7, by the induction assumption we conclude that v,v', and v'' belong to a common inner face F of G. As a consequence, we infer that v is an inner vertex of G of degree 3, leading to a contradiction if G is of type (4,4) or (3,6). If F',F'' have length 3 each (i.e., G has the type (6,3)), then the edges uv and vw belong to two other inner faces D' and D''. Let v' and v'' be the neighbors of v in D' and D''. Since $v',v''\in I(v,y)$ by Lemma 4.8, the vertices v,v',v'' belong to a common face D, and we conclude that v has degree 5 if $v'\neq v''$ and degree 4 if $v'=v''$. The case when G is of type (3,6) and only one of the faces F' and F'' has length 3 is analogous. On the other hand, if G is of type (6,3), then v has in $I(v,y)$ four vertices at distance 2, in contradiction to the induction hypothesis. Thus, in all cases, x may have maximum two consecutive neighbors in $I(x,y)$. To complete the proof, it remains to establish that if G is of type (6,3), then $I(x,y)$ contains at most two vertices at distance 2 from x. Let u and v be the neighbors of x in $I(x,y)$, and let F be the inner face of G passing via x,u, and v. Let u' and v' be the neighbors of u and v in F, and suppose, by way of contradiction, that v has yet another neighbor v'' in $I(v,y)\subset I(x,y)$. The vertices v',v,v'' belong to a common inner face F' of G. As $F\cap F'=\{v,v'\}$ and the faces F and F' are quasigated and have at least 6 vertices each, we conclude that the neighbors of v' in F and F' and the neighbor of v'' in F' are all different and belong to the interval $I(v,y)$, contrary to the induction hypothesis. □

4.4 Projections on zones

We specify the structure of projections of vertices on zones for each type of graphs.

Lemma 4.10 *The zone $Z(A,B)$ of any alternating cut $\{A,B\}$ of a (4,4)-graph G is quasigated, i.e., $Pr(x,Z(A,B))$ consists of one or two adjacent vertices. If additionally G is bipartite, then $Z(A,B)$ is gated.*

PROOF. Let $x\in A$ and denote by a,b the end vertices of the border line ∂A. Let u and v be the vertices of $Pr(x,Z(A,B))$ closest to a and b, respectively. Then $u\in I(a,x)$ and $v\in I(b,x)$. Suppose that $d(x,v)\leq d(x,u)$ and call $k=d(x,v)$ the *distance* of x to $Z(A,B)$. We proceed by induction on k. Suppose, by way of contradiction, that u and v are not adjacent. Then, the neighbor w of v in the convex path $I(u,v)\subseteq\partial A$ is different from u. Let y be a neighbor of x in $I(x,v)$. Since the distance of y to $Z(A,B)$ is smaller than k, by the induction hypothesis y has a quasigate in $Z(A,B)$. Hence, either $Pr(y,Z(A,B))=\{v\}$ or $Pr(y,Z(A,B))=\{v,w\}$. Notice that $d(y,w)\geq d(y,v)$, otherwise $w\in I(y,v)\subset I(x,v)$, contrary to the fact that $v\in Pr(x,Z(A,B))$. We will assume that $Pr(y,Z(A,B))=\{v,w\}$, the other case being similar. Then $w \in I(u, y) \subseteq I(a, y)$. Let z be the neighbor of u in the convex path $I(u, w)$. Then $v, w, y \in W(z, u)$, while $x \notin W(z, u)$ from the choice of u. From Lemma 4.4 we infer that one of the alternating cuts (say, $\{A', B'\}$), crossing the edge uz, also crosses the edge xy, say $u, x \in \partial A'$ and $z, y \in \partial B'$. Lemma 4.3 yields $I(u, x) \subseteq \partial A'$ and $I(z, w) \subset I(z, y) \subseteq \partial B'$. Hence uz is an inner edge of $E(A', B')$. Let F' and F'' be the faces of $Z(A', B')$ sharing uz, where $F' \in Z(A, B)$. If $z \neq w$, then F'' contains the edge zt of the path $I(z, w)$ incident to z. Since $F'' \cap F' = \{u, z\}$, the edge zt belongs to a face $F \neq F'$ of the zone $Z(A, B)$. Since the faces of G have at least 4 edges and the cut $\{A', B'\}$ is alternating, we deduce that z is an inner vertex of degree 3, a contradiction. Now suppose that $z = w$, i.e., $d(u, v) = 2$. In this case, $d(x, u) = d(x, z) = d(x, v)$. Also $d(y, v) = d(y, w)$ because $v, w \in Pr(y, Z(A, B))$. Let y_0 be the apex of y with respect to v, w, and let y' be the neighbor of y_0 in the convex path $I(y_0, z) \subseteq \partial B'$. Then $y' \in W(z, v)$ and $y_0 \notin W(z, v)$, whence, by Lemma 4.4, there is an alternating cut $\{A'', B''\}$ crossing y_0y' and vz, say $y', z \in \partial A''$ and $y_0, v \in \partial B''$. Since $y' \in \partial B'$ and G has type (4,4), one can easily see that y' is an inner vertex of degree at most 3: except its two neighbors in $I(x, z)$, y' may have only one other neighbor in $\partial A'$. □

For a vertex $x \in A$, we call two vertices u, v of $Pr(x, Z(A, B)) \subseteq \partial A$ *consecutive* if the subpath P of ∂A comprised between u and v does not contain other vertices of $Pr(x, Z(A, B))$.

Lemma 4.11 *If $\{A, B\}$ is an alternating cut of a (3,6)-graph G and $x \in A$, then $Pr(x, Z(A, B))$ induces a (convex) subpath of ∂A all vertices of which have the same distance to x.*

Proof. $Pr(x, Z(A, B))$ is a subset of the convex path ∂A. To establish the assertion, it suffices to show that two consecutive vertices u, v of $Pr(x, Z(A, B))$ are adjacent in G. Suppose, by way of contradiction, that u and v are not adjacent, and let P be the subpath of ∂A between u and v. Then, for any vertex $w \in P \setminus \{u, v\}$, at least one of the vertices u, v belongs to $I(w, x)$. Therefore, there exist two adjacent or coinciding vertices $w', w'' \in P$ such that $u \in I(w', x)$ and $v \in I(w'', x)$. If $w' = w''$, then the neighbors t' and t'' of w' in ∂A belong to the interval $I(w', x)$, therefore w', t', and t'' belong to a common inner face F (by Lemma 4.9). Since F is not in $Z(A, B)$ and w' belongs to at most three inner faces of $Z(A, B)$, we deduce that w' is an inner vertex of degree at most 4: w' has two neighbors in ∂A, at most two neighbors in ∂B and no other neighbors, yielding a contradiction.

Now suppose that w' and w'' are adjacent and $w' \notin I(w'', x), w'' \notin I(w', x)$, whence $d(x, w') = d(x, w'')$. Notice also that $w' \neq u$ or $w'' \neq v$, say the second. Let x_0 be the apex of x with respect to w', w'', and let $x' \in I(x_0, w')$ and $x'' \in I(x_0, w'')$ are adjacent to x_0. Consider the two alternating cuts $\{A', B'\}$ and $\{A'', B''\}$ crossing the edge $w'w''$. Then one of these cuts, say $\{A', B'\}$, will cross the edge x_0x' and the second one $\{A'', B''\}$ will cross the edge x_0x''. Obviously, the cut $\{A', B'\}$ will also cross an edge of every shortest (x, u)-path and the cut $\{A'', B''\}$ will cross an edge $z'z''$ of every shortest (x, v)-path. Let $x, z', w' \in A''$ and $z'', w'' \in B''$, more precisely $z', w' \in \partial A''$ and $z'', w'' \in \partial B''$. Let also F be the face of $Z(A, B)$ containing the edge $w'w''$. Since $v \in I(w'', x)$ and $z'' \in I(v, x)$, we conclude that $v \in I(w'', z'') \subseteq \partial B'$. Therefore, the vertex w'' belongs to a face $F \in Z(A, B) \cap Z(A'', B'')$, to another face D of $Z(A'', B'')$ and maybe to two other faces of $Z(A, B)$. Since w'' may have only one neighbor in $D \cap \partial A''$, except its neighbor in $I(w'', v)$ (this will happen if D has length 3), we conclude that w'' is an inner vertex of G but its degree is 4 or 5, a contradiction. □

We call a sequence of vertices $u_1, u_2, \ldots, u_k$ of ∂A a *2-path* if $d(u_i, u_{i+1}) = 2$ for all $i = 1, \ldots, k-1$.

Lemma 4.12 *If $\{A, B\}$ is an alternating cut of a (6,3)-graph G and $x \in A$, then $Pr(x, Z(A,B))$ is a vertex, two adjacent vertices, or a 2-path such that any its consecutive vertices belong to different faces of $Z(A,B)$. In the latter case, all vertices of $Pr(x, Z(A,B))$ except maybe the leftmost and/or the rightmost vertices have the same distance to x, while one or both end vertices may be one step further from x.*

PROOF. First we will establish that $d(u,v) \leq 2$ for two consecutive vertices $u, v \in Pr(x, Z(A,B))$. Suppose, by way of contradiction that, $d(u,v) \geq 3$. Then, as in the proof of Lemma 4.11, there exist two adjacent or coinciding vertices $w', w'' \in P$ such that $u \in I(w', x)$ and $v \in I(w'', x)$. If $w' = w''$, then the neighbors $t' \in I(w', u)$ and $t'' \in I(w', v)$ of w' in ∂A belong to the interval $I(w', x)$, therefore, w', t', and t'' belong to a common inner face F of G. This face intersects $Z(A,B)$ along the edges $w't'$ and $w't''$ (which must belong to different faces of this zone). As $d(u,v) \geq 3$, we may suppose, without loss of generality, that $t' \neq u$. Since G is a (6,3)-graph, we conclude that w' will have in $I(w', x)$ at least three distinct vertices at distance 2: the neighbors of t' and t'' in F and the neighbor of t' in $I(t', u)$, which is impossible by Lemma 4.9. If $w' \neq w''$, then $w' \neq u$ or $w'' \neq v$, say the second. We employ the same notations as in the similar case for (3,6)-graphs (see the proof of Lemma 4.11): x_0 is the apex of x with respect to w', w'', $x' \in I(x_0, w')$ and $x'' \in I(x_0, w'')$ are neighbors of x_0, $\{A', B'\}$ is the alternating cut crossing the edges $w'w''$ and x_0x' and $\{A'', B''\}$ is the alternating cut crossing the edges $w'w''$ and x_0x''. Finally, let $F \in Z(A,B)$ and D be the faces of $Z(A'', B'')$ sharing the edge $w'w''$. Notice that both F and D are also faces of the zone $Z(A', B')$. Since G is a (6,3)-graph and $I(w'', v) \cup I(v, z') \subseteq \partial B''$, the face D contains the neighbor t of w'' in $I(w'', v)$ (t exists because $w'' \neq v$) and the neighbor of t in $\partial B''$. Analogously, if $\{A', B'\}$ crosses the edge $y'y''$ of a shortest (x,u)-path and $w', y' \in \partial A'$, $w'' \in \partial B''$, since $I(w', u) \cup I(u, y') \subseteq \partial A'$, we conclude that the neighbor s of w' in $I(w', u)$ (if it exists) belongs to the face D. Since $F \cap D = \{w', w''\}$, the vertex w'' has a neighbor in ∂B. But now t will be an inner vertex of degree 2 unless $t = v$. Hence $w' \neq u$, and in a completely analogous way we can show that $s = u$, whence $u \in D$. But on the other hand, $u \in F$, because F has length ≥ 6 and therefore w' cannot have a neighbor in ∂B. This contradiction with Lemma 4.5 establishes that indeed $d(u,v) \leq 2$.

If $d(u,v) = 2$, then u and v must belong to different faces of $Z(A,B)$ because the faces of G are quasigated. We also assert that in this case the vertices u and v belong to a common inner face $F \notin Z(A,B)$. Let y be their common neighbor. If $d(u,x) = d(v,x)$, then $u, v \in I(y,x)$ and u, v, y lie on a common face by Lemma 4.9. So, assume $d(u,x) < d(v,x) = d(y,x)$. From last equality and Lemma 4.8 one concludes that y and v belong to a common face $F' \notin Z(A,B)$. If $u \in F'$, we are done. Otherwise, let y' be the neighbor of y in F', and let t' be the neighbor of y' in F'. Since $y', u \in I(y,x)$, there is a face F'' passing via u, y' and y by Lemma 4.9. Then, the neighbor t'' of y' in F'' is different from t'. Therefore, t'', t', and the neighbor of u in F'' are three vertices of $I(y,x)$ at distance 2 from y, yielding a contradiction with the second assertion of Lemma 4.9. Thus, u and v belong to a common face.

In order to further specify the structure of the projections, first suppose that $Pr(x, Z(A,B))$ contains two adjacent vertices u, v. We assert that in this case $Pr(x, Z(A,B))$ does not contain any other vertices. For this, we proceed by induction on $d(x,u)$. Suppose, by way of contradiction, that w is yet another vertex from the projection. In view of what has been proven above, w may be chosen so that $d(v,w) \leq 2$. First assume that v and w are adjacent. Then, the vertices u, v, and w have the same distance to x and the vertices u, w belong to different faces of $Z(A,B)$. Also one can easily see that v is an inner vertex of G. Let F' and F'' be the faces containing the edges uv and vw provided by Lemma 4.8, respectively (which do not belong to $Z(A,B)$). Let $v' \in F'$ and $v'' \in F''$ be the neighbors of v and let t' and t'' be the neighbors in F' and F'' of v' and v'' different from v. Notice that the vertices v', v'', t', t'' belong to the interval $I(v,x)$. If $v' \neq v''$, then let F be the face passing via v, v', v'' (this face exists by Lemma 4.9). The neighbors z' and z'' in F of v' and v'' also belong to $I(v,x)$ and are different from t' and t'', whence $I(v,x)$ contains at least 4 vertices at distance 2 from v, which contradicts Lemma 4.9. So, assume that $v' = v''$, and let F be the face passing via the vertices $t', t'' \in I(v',x)$ and v'. Let z' and z'' be the neighbors of t' and t'' in F. Denote also by s' and s'' the neighbors of t' and t'' in F' and F''. Since $z' \neq s'$ and $z'' \neq s''$ and $z', z'' \in I(v',x)$, from Lemma 4.9 we conclude that s' and s'' do not belong to $I(v',x)$. This is possible only if F' and F'' are faces of length 6. Let u' be the common neighbor of s' and u in F'. Consider an alternating cut $\{A', B'\}$ which crosses the edges $s'u'$ and $v'v$. Then, $s', t', t'', s'' \in Pr(x, Z(A', B'))$, which is impossible by the induction hypothesis. Now consider the case when $d(v,w) = 2$ and $d(u,w) = 3$. Let y be the common neighbor of v and w. Let the face F' play the same role as in previous case. From the first part of the proof we know that v, y, w belong to a common face F''. Let, as above, v' and v'' be the neighbors of v in F' and F'', and let t' and t'' be the neighbors of v' and v'' in F' and F''. Notice that $t', t'' \in I(v,x)$, therefore, if $v' \neq v''$, then v, v', v'', t', t'' belong to a common face by Lemma 4.9, yielding that v' and v'' are inner vertices of degree 2, a contradiction. So, let $v' = v''$, and denote by F the face containing v', t', and t'' and denote by z' and z'' the neighbors of t' and t'' in F. Since $z', z'' \in I(v',x)$, again by Lemma 4.9, we infer that the neighbors s' and s'' of t' and t'' in F' and F'', respectively, do not belong to $I(v',x)$, whence t' and t'' belong to the projections of x on the faces F' and F''. This implies that F' has length 6 and F'' has length 6 or 7. Again consider an alternating cut $\{A', B'\}$ crossing the edges $s'u'$ and $v'v$, where u' is the common neighbor of s' and u in F'. Then $s', t', t'' \in Pr(x, Z(A', B'))$, and we get a contradiction with the induction hypothesis. This shows that if the projection of x on $Z(A,B)$ contains two adjacent vertices, then it does not contain any other vertices.

Now suppose that $Pr(x, Z(A,B)) = \{u_1, u_2, \ldots, u_k\}$ is a 2-path. To prove the lemma in this case, it suffices to establish that $d(u_i, x) \leq \min\{d(u_{i-1}, x), d(u_{i+1}, x)\}$ for any $i = 2, \ldots, k-1$. Suppose, by way of contradiction, that $d(u_i, x) > d(u_{i-1}, x)$. Let F' be the face containing the vertices u_{i-1} and u_i, and let F'' be the face containing the vertices u_i and u_{i+1}. Then F' and F'' must intersect along an edge $u_i y$, otherwise we can show as before that $I(u_i, x)$ contains at least four vertices at distance 2 from u_i. Let t' and t'' be the neighbors of y in F' and F''. Since $t', t'' \in I(y,x)$, they belong to a common face F. Let z' and z'' be the neighbors of t' and t'' in F. Since F is quasigated and $d(t',x) = d(t'',x)$ one can easily conclude that z' and z'' belong to $I(y,x)$. On the other hand, since $d(u_i, x) = d(u_{i-1}, x) + 1$ and F' is quasigated, either the vertex t' does belong to the projection of x on F' or F' has length 6. In the either case, the neighbor s' of t' in F' belongs to the interval $I(t',x) \subset I(y,x)$. Since s' is different from z' and z'', the interval $I(y,x)$ contains three vertices at distance 2 from y, a contradiction with Lemma 4.9. If F' has length 6, let s be the common neighbor of t' and u_{i-1}. Consider an alternating cut $\{A', B'\}$ crossing the edges su_{i-1} and yu_i. Then we easily see that the adjacent vertices s, t', and a vertex of the face F'' belong to $Pr(x, Z(A', B'))$, contrary to what has been proven above. This completes the proof of the lemma. □

Lemma 4.13 *Given a vertex $x \in A \setminus \partial A$ and an alternating cut $\{A, B\}$ of G, there exist two (not necessarily distinct) neighbors u_x and v_x of x such that $I(x, w) \cap \{u_x, v_x\} \neq \emptyset$ for any vertex $w \in Pr(x, Z(A, B))$. In particular, $I(x, y) \cap \{u_x, v_x\} \neq \emptyset$ for any vertex $y \in B$. Analogously, any vertex $x \in \partial A$ has two neighbors u_x, v_x in $Z(A, B)$ such that $I(x, y) \cap \{u_x, v_x\} \neq \emptyset$ for any vertex $y \in B$.*

PROOF. First suppose that $x \in A \setminus \partial A$. The result is obvious if $Pr(x, Z(A, B))$ consists of one or two vertices (in particular, for (4,4)-graphs in view of Lemma 4.10): as u_x and v_x it suffices to take any neighbors of x on shortest paths connecting x with the vertices from the projection. Analogously, if x contains a neighbor $u_x \in I(x, u) \cap I(x, v)$ (where u and v are the end vertices of $Pr(x, Z(A, B))$), then from the properties of projections one concludes that $Pr(u_x, Z(A, B)) = Pr(x, Z(A, B))$ and that $u_x \in I(x, w)$ for any $w \in Pr(x, Z(A, B))$. So, further assume that $I(x, u) \cap I(x, v) = \{x\}$, in particular, $d(x, v) \leq d(u_x, v)$ for any neighbor u_x of x in $I(x, u)$. Let $Pr(u_x, Z(A, B)) \subseteq I(u, v')$ and let t be a neighbor of u_x in $I(u_x, v') \subseteq I(u_x, v)$. If $d(x, v) < d(u_x, v)$, then denote by F the face containing the vertices u_x, t, x (see Lemma 4.9), otherwise if $d(x, v) = d(u_x, v)$, then denote by F the face containing the edge xu_x and provided by Lemma 4.8. Let v_x be the neighbor of x in F different from u_x.

First, suppose that G is a graph of type (3,6). By Lemma 4.11, $Pr(x, Z(A, B)) = I(u, v)$ and all its vertices have the same distance to x. Moreover $Pr(u_x, Z(A, B)) \subseteq Pr(x, Z(A, B))$ and every vertex of $Pr(u_x, Z(A, B))$ is closer to u_x than to x, because u_x is one step closer to $Z(A, B)$ than x. Suppose by way of contradiction that v' is not adjacent to v (otherwise we are done). Since $v' \in I(v, u_x)$ and $d(x, v) \leq d(u_x, v)$ we conclude that $d(v', v) = 2$ and $x \in I(u_x, v)$. The face F defined above and passing via u_x, t, x will have length ≥ 4, otherwise x and t are adjacent and $u_x \notin I(x, v')$. Since $x, t \in I(u_x, v)$ and F is quasigated, one can easily conclude that $v_x \in I(x, v)$. We also assert that $\{u_x, v_x\} \cap I(x, w) \neq \emptyset$, where w is the common neighbor of v' and v. Indeed, the vertex t is closer to w than the vertices u_x and x, therefore the distance from w to F is at most $d(w, x) - 1$, thus $x \notin I(x, w) \cap Pr(w, F)$ and a shortest path from x to w crossing this intersection will go via u_x or v_x.

Now, let G be a (6,3)-graph. By Lemma 4.12, we may assume that $d(u, v) > 2$ and that $Pr(x, Z(A, B))$ consists of each second vertex of the subpath of ∂A comprised between u and v. As in the case of (3,6)-graphs or applying Lemma 4.8 one concludes that x does not belong to the projections of v and v' on F. Moreover, v_x belongs to a shortest path P between v and x passing via $Pr(v, F)$. Now, one can easily see that for any other vertex $w \in Pr(x, Z(A, B))$, every shortest path connecting w with x and intersecting $Pr(w, F)$ will pass via u_x or via v_x. The vertices $w \in Pr(x, Z(A, B))$ for which $u_x \in I(w, x)$ form a subchain of $Pr(x, Z(A, B))$ because if $w', w'' \in Pr(x, Z(A, B))$ have this property then every vertex of $Pr(x, Z(A, B))$ comprised between w' and w'' also will have this property (the same is true for the vertices $w \in Pr(x, Z(A, B))$ for which $v_x \in I(w, x)$).

Finally notice that for any vertex $y \in B$ any shortest (x, y)-path crosses $Z(A, B)$, therefore there will be such shortest paths which will go via $Pr(x, Z(A, B))$. From what has been proven above we conclude that there exist shortest (x, y)-paths going via u_x or v_x.

Now suppose that $x \in \partial A$. Since any shortest path connecting x with a vertex of B crosses the path ∂B, to establish the second assertion it suffices to show that x has two neighbors

$u_x, v_x \in Z(A,B)$ such that $I(x,z) \cap \{u_x, v_x\} \neq \emptyset$ for any vertex $z \in \partial B$. If x does not have any neighbor in ∂B, then as u_x and v_x we will consider the two neighbors of x in ∂A. Now suppose that x has precisely one neighbor in ∂B, say y, and let $\{A', B'\}$ and $\{A'', B''\}$ be the alternating cuts crossing the edge xy (i.e. $\{A,B\}$ is one of these cuts); see Figure 3. Let x' and x'' be the neighbors of x in ∂A, where x' is between x and w and x'' is between u and x. For all vertices $z \in (\partial B' \setminus L') \cup (\partial B'' \setminus P'')$ there is a shortest (x,z)-path going via y; for all vertices $z \in L'$ there is a shortest (x,z)-path going via x' and for all vertices $z \in P''$ there is a shortest (x,z)-path going via x''. Therefore we can set $\{u_x, v_x\} := \{x', y\}$ if $\{A,B\} = \{A', B'\}$ and $\{u_x, v_x\} := \{x'', y\}$ if $\{A,B\} = \{A'', B''\}$. Finally, if x contains two neighbors in ∂B, then they form a triangular face and from the definition of alternating cut one concludes that there is a shortest path between x and any vertex of ∂B going via these neighbors. For example, if in Figure 3 x is adjacent to q, then $\{A,B\} = \{A', B'\}$ and for any vertex of L' there is a shortest path to x going via q (for any other vertex of $\partial B'$ there will be a shortest path to x going via y). $\square$

4.5 A distance property

Let v be an inner vertex of G and let $u_0, u_1, \ldots, u_{k-1}$ be the neighbors of v labeled counterclockwise. Notice that the sets $W(u_i, v)$ and $W(u_j, v)$ are disjoint unless u_i and u_j are consecutive or coincide. Indeed, if $z \in W(u_i, v) \cap W(u_j, v)$ and $i \neq j$, then $u_i, u_j \in I(v,z)$, therefore, by Lemma 4.9, x, u_i and u_j belong to a common face of G.

Lemma 4.14 *If G is a (4,4)- or a (6,3)-graphs, $x \in W(u_i, v)$, $y \in W(u_j, v)$, and u_i, u_j are not p-consecutive for $p \leq 2$, then $v \in I(x,y)$. Analogously, if G is a (3,6)-graph, $x \in W(u_i, v)$, $y \in W(u_j, v)$, and u_i, u_j are not p-consecutive for $p \leq 3$, then $v \in I(x,y)$.*

Proof. First, by induction on $d(y, u_j)$, we will show that if $y \in W(u_j, v)$ and u_i and u_j are not consecutive in a graph G of type (4,4) or (6,3), then $v \in I(u_i, y)$ (this covers the assertion in the case $x = u_i$). An analogous assertion holds for a (6,3)-graph provided u_i and u_j are neither consecutive nor 2-consecutive. The result is obvious if $y = u_j$. So, assume $y \neq u_j$ and let y' be a neighbor of y in $I(y, u_j)$. Since $y' \in W(u_j, v)$, the induction hypothesis yields $v \in I(u_i, y')$, therefore, $y' \in W(v, u_i)$. If $v \notin I(u_i, y)$, then $y \notin W(v, u_i)$. From Lemma 4.4 there exists an alternating cut $\{A,B\}$ such that $u_i, y \in \partial A$ and $v, y' \in \partial B$. Moreover $u_j \in I(v, y') \subseteq \partial B$. This immediately implies that the vertices u_i, v, u_j belong to a common face of $Z(A,B)$, which is impossible because u_i and u_j are not consecutive. For (3,6)-graphs either we get the same contradiction, or u_i, v, and the neighbor z of u_i in ∂A constitute a triangular face of $Z(A,B)$, while v, z, and u_j belong to another face of $Z(A,B)$, from which we infer that u_i and u_j are 2-consecutive. This contradiction establishes the required inclusion $v \in I(u_i, y)$.

Now we consider the general case $x \neq u_i$. We may suppose, without loss of generality, that $I(x,y) \cap W(u_i, v) = \{x\}$, otherwise we can use induction on $d(x, u_i)$. Let z be a neighbor of x in $I(x,y)$. Since $z \notin W(u_i, v)$, by Lemma 4.4, there exists an alternating cut $\{A,B\}$ crossing the edges $u_i v$ and xz. Let $u_i, x \in \partial A$ and $v, z \in \partial B$. Denote by u_l the neighbor of v in the convex path ∂B. Since u_l and u_j are not consecutive if G is a graph of type (4,4) or (6,3), and u_l and u_j are not consecutive or 2-consecutive if G is a graph of type (3,6), from the first part of this proof we obtain that $v \in I(u_l, y)$.

If the projection of y on $Z(A,B)$ does not intersect the subpath of ∂B starting at u_l, passing via z, and ending at a vertex of ∂G, then $v \in I(z,y) \subset I(x,y)$, and we are done. On the other hand, if the whole projection $Pr(y, Z(A,B))$ is contained in this subpath of ∂B, then necessarily $u_l \in I(v,y)$. Since $u_j \in I(v,y)$, from Lemma 4.9 we conclude that u_l and u_j are consecutive, i.e., u_i and u_j must be 2-consecutive. So, $Pr(y, Z(A,B))$ must have vertices of ∂B on both sides of u_l. This is impossible if G is a graph of type (4,4): by Lemma 4.10, $Pr(y, Z(A,B))$ will consist of two adjacent vertices v and u_l, however $v \in I(u_l, y)$ as noticed above. If G has type (6,3), then from Lemma 4.12 and the distance constraints between the vertices y, v, u_l we infer that v and the second neighbor t of u_l in ∂B belong to $Pr(y, Z(A,B))$. However this is not possible because v, u_l, and t belong to the same face of $Z(A,B)$ and this face must be quasigated by Lemma 4.7. Finally, if G is a (3,6)-graph, then Lemma 4.11 implies that both v and u_l belong to $Pr(y, Z(A,B))$, which is impossible because $v \in I(u_l, y)$ by what has been shown above. This contradiction completes the proof. $\square$

4.6 Partition into cones

Let v be an inner vertex of G and let $N(v) = \{u_0, \ldots, u_{k-1}\}$ be the set of neighbors of v ordered counterclockwise around v. Every edge vu_i is crossed by two alternating cuts $\{A'_i, B'_i\}$ and $\{A''_i, B''_i\}$. Recall that we have chosen an orientation of these cuts such that v is on the left border. Moreover, we suppose that $\{A'_i, B'_i\}$ is the cut for which the last turn before $u_i v$ (if it exists) is to the right and thus the next turn after $u_i v$ is to the left. For each neighbor u_i of v, define the *cone with apex* u_i as $C_v(u_i) := B'_i \cap A'_{i+1(mod\ k)}$. Let Γ_i be the closed walk which starts at u_i follows $\partial B'_i$ backward (with respect to the orientation of the cut $\{A'_i, B'_i\}$) until a boundary vertex $b_i \in \partial G \cap \partial B'_i$, goes on the boundary in counterclockwise order until it meets a vertex a_i in $\partial A'_{i+1} \cap \partial G$ and then comes back to u_i following the subpath of $\partial A'_{i+1}$ comprised between b_i and u_i. From the definition of $C_v(u_i)$ and Lemma 4.4 it follows that the cone $C_v(u_i)$ consists of all vertices of G lying on Γ_i or inside the region bounded by Γ_i (see Figure 4).

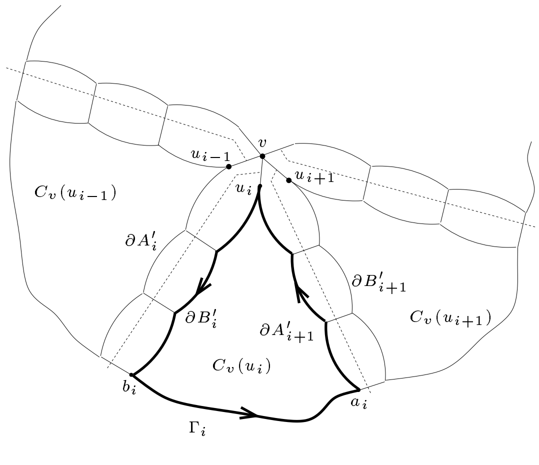

Figure 4: The cone $C_v(u_i)$

Lemma 4.15 *$C_v(u_i)$ consists of all vertices x such that $I(v,x) \cap N(v)$ equals $\{u_i\}$ or $\{u_{i-1}, u_i\}$. In particular, together with $\{v\}$ the cones $C_v(u_i)$ $(i = 0, \ldots, k-1)$ constitute a partition of the vertex set of G, each set containing at most $n/2$ vertices.*

PROOF. We first show that for a vertex $x \in C_v(u_i)$ any shortest (x,v)-path goes via u_i or u_{i-1}. Let yz be the first edge on this path such that $y \in C_v(u_i)$ and $z \notin C_v(u_i)$. Then, either $y \in \partial A'_{i+1}$ and $z \in \partial B'_{i+1}$ or $y \in \partial B'_i$ and $z \in \partial A'_i$. In the first case, since u_i is located on the subpath of the convex path $\partial A'_{i+1}$ between y and v, the unique shortest (y,v)-path is a subpath of $\partial A'_{i+1}$ which goes via u_i. In the second case, analogously, either $z = v$ or u_{i-1} is located on the subpath of the convex path $\partial A'_i$ between z and v (z and u_{i-1} may coincide). Therefore, the unique shortest (z,v)-path goes via u_{i-1}. We have shown that no shortest (x,v)-path goes via a neighbor of v different from u_i and u_{i-1}. Conversely, let x be a vertex of G such that $I(x,v) \cap N(v)$ equals $\{u_i\}$ or $\{u_{i-1}, u_i\}$. Pick a shortest (x,v)-path which goes via u_i. Pick the first edge yz on this path such that $z \notin C_v(u_i)$ and $y \in C_v(u_i)$. If $y \in \partial B'_i$ and $z \in \partial A'_i$, we obtain a contradiction with the convexity of $\partial A'_i$ because $z, v \in \partial A'_i$ while a shortest path joining them leaves $\partial A'_i$. Otherwise, we have $z \in \partial B'_{i+1}$ and $y \in \partial A'_{i+1}$. By Lemma 4.4, z belongs to $W(u_{i+1}, v)$, hence there is a shortest (z,v)-path which goes via u_{i+1}, in contradiction with the choice of x. □

Two cones $C_v(u_i)$ and $C_v(u_j)$ are called *p-neighboring* if $\min\{|i-j|, k-|i-j|\} = p$.

Lemma 4.16 *If $x \in C_v(u_i)$ and $y \in C_v(u_j)$ and the cones $C_v(u_i)$ and $C_v(u_j)$ are not p-neighboring for $p \leq 2$, then $d(x,y) = d(x,v) + d(v,y)$. For squaregraphs, a similar result holds with $p \leq 1$.*

PROOF. For graphs of type (4,4) or (6,3) the result directly follows from Lemma 4.14 because $C_v(u_i) \subseteq W(u_i, v)$ and $C_v(u_j) \subseteq W(u_j, v)$ by Lemma 4.15. The same argument can be applied for (3,6)-graphs except the case when u_i and u_j are 3-consecutive. Let u_l be as in the proof of Lemma 4.14. Following the same proof we will obtain the required property $d(x,y) = d(x,v) + d(x,y)$ except the case when $v, u_l \in Pr(y, Z(A,B))$. Then $d(y,v) = d(y,u_l)$. Consider the face $F \notin Z(A,B)$ containing the edge vu_l and provided by Lemma 4.8. Then the neighbor $u_r \neq u_l$ of v in F belongs to the interval $I(v,y)$. Since u_j also belongs to $I(v,y)$, from Lemma 4.9 we infer that u_r and u_j are consecutive. In the partition into cones, y will belong to the cone $C_v(u_r)$ which is 2-neighboring with $C_v(u_i)$ and not to $C_v(u_j)$.

Finally, if G is a squaregraph and $C_v(u_i)$ and $C_v(u_j)$ are 2-neighboring, however $v \notin I(x,y)$, then $I(v,x)$ and $I(v,y)$ will not intersect only in v. Let u_l be a neighbor of v in this intersection. By Lemma 4.9, we conclude that u_l is consecutive to u_i and u_j. From Lemma 4.15 we deduce that one of the vertices x or y must belong to the cone $C_v(u_l)$, in contradiction with our assumption. □

5 Distance queries and routing

In this section, we describe in details the distance and routing labeling schemes in a graph G of type (4,4),(3,6), and (6,3).

5.1 Distance between two paths of a zone

Here we describe a subroutine for reporting in constant time the distance between two paths $P \subseteq \partial A$ and $Q \subseteq \partial B$ of the zone $Z(A,B)$ of an alternating cut $\{A,B\}$. For this, first we present a labeling of the vertices of $Z(A,B)$ which allows to compute the distance between any two vertices $p \in \partial A$ and $q \in \partial B$ in $O(1)$ time. Pick an edge $ab \in E(A,B) \cap \partial G$, where $a \in \partial A$, $b \in \partial B$, and ∂A is the left border line of $Z(A,B)$. Suppose, without loss of generality, that the last turn (if it exists) of $E(A,B)$ before the edge ab is to the right (the other case being analogous). Then for every edge $a'b' \in Z(A,B)$ with $a' \in A$ and $b' \in B$ either $d(a,a') = d(b,b')$ or $d(a,a') = d(b,b') + 1$ holds. We say that the edge $a'b'$ is *horizontal* in the first case and *inclined* in the second case. For a vertex $b' \in \partial B$ define $\alpha_1(b') := \min\{d(a',a) - d(b',b) : a'b' \in E(A,B)\}$ and $\alpha_2(b') := \max\{d(a',a) - d(b',b) : a'b' \in E(A,B)\}$. If G is of type $(4,4)$ or $(6,3)$, then b' has a unique neighbor in ∂A and thus $\alpha_1(b') = \alpha_2(b') \in \{0,1\}$. On the other hand, if G is of type $(3,6)$ and b' belongs to a triangular face of the zone $Z(A,B)$, then we may have $\alpha_1(b') = 0$ and $\alpha_2(b') = 1$. We say that a vertex $a' \in \partial A$ is *above* vertex $p \in \partial A$ if it belongs to the subpath of ∂A comprised between p and a, and *below* p otherwise (we employ the same terminology for vertex $q \in \partial B$ and the vertices $b' \in \partial B$). By convention, p is above and below itself. Let $r(p)$ be the first vertex above p which is incident to an inclined edge of $E(A,B)$ and let $s(p)$ be the first vertex below p which is incident to a horizontal edge of $E(A,B)$ (if such vertices do not exist, then set $r(p) := a$ and let $s(p)$ be the second end vertex of ∂A). Let $above(p)$ and $below(p)$ be the first vertices above and below p which are incident to edges of $E(A,B)$ (notice that $above(p) = p = below(p)$ if p has a neighbor in ∂B). Let also $Above(p)$ (with capital A) be the first vertex strictly above p which is incident to an edge of $E(A,B)$ (if $p = a$ we set $Above(p) := p$). Clearly, unless $p = a$, $Above(p) \neq p$ holds. Analogously define $above(q)$, $Above(q)$ and $below(q)$ for the vertices $q \in \partial B$.

We say that a vertex $p \in \partial A$ is *above* a vertex $q \in \partial B$ if $d(a, below(p)) \leq d(b, above(q)) + \alpha_2(above(q))$ (in a similar, way we say q is above p if $d(b, below(q)) + \alpha_1(below(q)) \leq d(a, above(p))$).

Using the labels defined above, we can recognize in $O(1)$ time if two vertices $p', p'' \in \partial A$ belong to a common face of $Z(A,B)$: this holds if and only if

$$d(a, Above(p')) = d(a, Above(p'')) \text{ or } d(a, below(p')) = d(a,p') = d(a, Above(p''))$$

(a similar test can be applied to the vertices $q', q'' \in \partial B$). Analogously, one can test in $O(1)$ time if two vertices $p \in \partial A$ and $q \in \partial B$ belong to a common face of $Z(A,B)$. This happens if and only if at least one of the equalities

$$(p = a \text{ and } q = b) \text{ or } d(a, Above(p)) = d(b, Above(q)) + \alpha_2(Above(q)) \text{ or}$$

$$d(a, below(p)) = d(a,p) = d(b, Above(q)) + \alpha_2(Above(q)) \text{ or}$$

$$d(b, below(q)) = d(b,q) = d(a, Above(p)) - \alpha_2(above(q)) \text{ or}$$

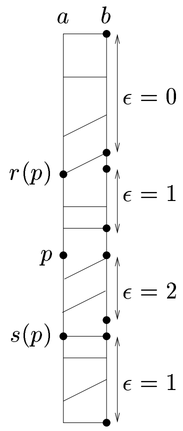

Figure 5: Illustration for formula (1)

$$d(a, Above(p)) = d(a, p) - 1 = d(b, q) = d(b, below(q)).$$

We call this the common_face_test. A formal description of this procedure is given below.

If two vertices $p \in \partial A$ and $q \in \partial B$ are such that p is above q or q is above p then, a simple case analysis shows that the distance between two vertices p and q can be computed via the formula (see Figure 5)

$$d(p, q) = |d(p, a) - d(q, b)| + \epsilon, \tag{1}$$

where

$$\epsilon = \begin{cases} 0 & \text{if } d(q, b) < d(r(p), a), \\ 1 & \text{if } d(r(p), a) \leq d(q, b) < d(p, a) \text{ or } d(s(p), a) \leq d(q, b), \\ 2 & \text{if } d(p, a) \leq d(q, b) < d(s(p), a). \end{cases}$$

Otherwise, one can see that p and q belong to a common face of $Z(A, B)$ and that $d(p, q)$ is the minimum of

$$d(a, p) - d(a, above(p)) + 1 + d(b, q) - d(b, above(q)) \tag{2}$$

and

$$d(a, below(p)) - d(a, p) + 1 + d(b, below(q)) - d(b, q), \tag{3}$$

Therefore we can retrieve $d(p, q)$ in $O(1)$ time from the label

$$\begin{array}{lccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ B_p := & (1, & d(p,a), & d(above(p),a), & d(below(p),a), & d(s(p),a), & d(r(p),a), & d(Above(p),a)) \end{array}$$

of $p \in \partial A$ and the label

$$\begin{array}{lccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ B_q := & (0, & d(q,b), & d(above(q),b), & d(below(q),b), & \alpha_2(above(q)), & \alpha_1(below(q)), & d(Above(q),b), \end{array}$$

$$\overset{8}{\alpha_2(Above(q)))}$$

of $q \in \partial B$ (see function distance_vertices(B_p, B_q) below). The first entry in B_p (and in B_q) is a bit that indicates that the last turn (if it exists) of $E(A, B)$, before the edge ab, is to the right. If the last turn is to the left, we need to interchange the roles of p and q in the consideration.

Notice also that $d(p', p'') = |d(a, p') - d(a, p'')|$ for any two vertices $p', p'' \in \partial A$ and $d(q', q'') = |d(b, q') - d(b, q'')|$ for any two vertices $q', q'' \in \partial B$.

function distance_vertices(B_p, B_q)

```
if B_p(1) = 0 then /* rename inputs */ set C := B_p, B_p := B_q, B_q := C
if B_p(4) ≤ B_q(3) + B_q(5) or B_q(4) + B_q(6) ≤ B_p(3) then
   if B_q(2) < B_p(6) then set ε := 0
   if B_p(6) ≤ B_q(2) < B_p(2) or B_p(5) ≤ B_q(2) then set ε := 1
   if B_p(2) ≤ B_q(2) < B_p(5) then set ε := 2
   return |B_p(2) − B_q(2)| + ε.
else
   return min{B_p(2) − B_p(3) + 1 + B_q(2) − B_q(3), B_p(4) − B_p(2) + 1 + B_q(4) − B_q(2)}
```

function common_face_test$(B_p, B_q,$ flag)

```
do case
   case flag = 0    /* i.e., p and q are on the same side of the zone */
      if B_p(7) = B_q(7) or B_p(4) = B_p(2) = B_q(7) then return true
      else return false
   case flag = 1    /* i.e., p and q are on different sides of the zone */
      if B_p(1) = 0 then /* rename inputs */ set C := B_p, B_p := B_q, B_q := C
      if B_p(2) = B_q(2) = 0 or B_p(7) = B_q(7) + B_q(8) or B_p(4) = B_p(2) = B_q(7) + B_q(8) or
         B_q(4) = B_q(2) = B_p(7) + B_q(5) or B_p(7) = B_p(2) − 1 = B_q(2) = B_q(4) then return true
      else return false
end case.
```

Now we will adjust this labeling scheme in order to compute the distance $d(P, Q) := \min\{d(p, q) : p \in P, q \in Q\}$ between two paths $P \subseteq \partial A$ and $Q \subseteq \partial B$. Let p', p'' be the end vertices of P and let q', q'' be the end vertices of Q, such that $d(p', a) \le d(p'', a)$ and $d(q', b) \le d(q'', b)$. We distinguish between four complementary cases depending of the reciprocal location of P and Q; see Figure 6 for an illustration: (i) one path is above another; (ii) one path is contained in a face of $Z(A, B)$ and the second path intersects this face; (iii) there is a unique face F of $Z(A, B)$ intersecting both paths P, Q and neither of these paths is contained in F; (iv) several faces of $Z(A, B)$ intersect both paths P and Q.

We say that path P is above Q if p'' is above q'. In this case, $d(P, Q) = d(p'', q')$. Analogously, Q is above P if q'' is above p'. Then $d(P, Q) = d(p', q'')$. This settles case (i), therefore further we may suppose that neither P is above Q nor Q is above P.

Applying the common_face_test, we can check in constant time if any pair of the vertices p', p'', q', q'' belong to a common face of $Z(A, B)$. Suppose this test returned that p' and p'' belong to a common face, say F. If q' and q'' also belong to a common face (which cannot be other than F), then

$$d(P, Q) = \min\{d(p', q'), d(p'', q'')\}. \tag{4}$$

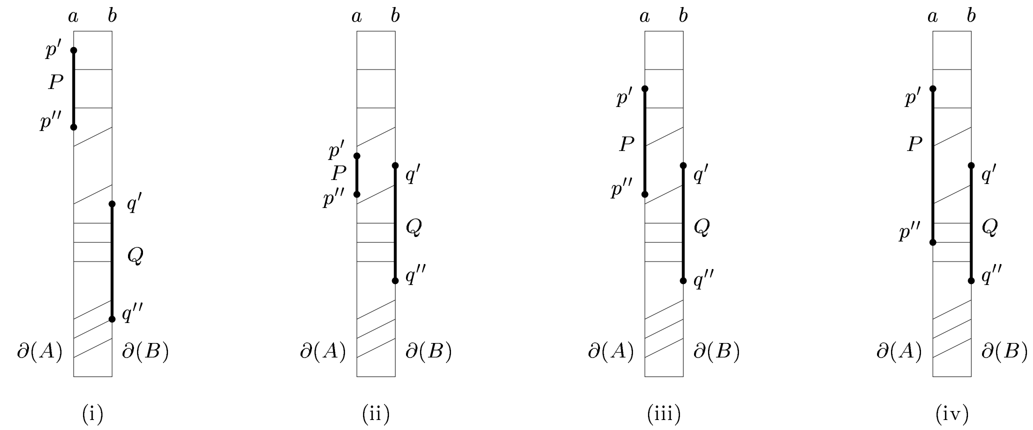

Figure 6: Cases (i)-(iv)

If $q' \in F$ and $q'' \notin F$ (employing the common_face_test for p', q' and p', q''), then

$$d(P,Q) = \min\{d(p',q'), d(p'', below(p'')) + 1\}, \tag{5}$$

because the neighbor of $below(p'')$ in ∂B will be a vertex of Q (a similar formula holds if $q'' \in F$ and q' does not belong to F). Finally, if both vertices q' and q'' are outside F, then

$$d(P,Q) = \min\{d(p', above(p')) + 1, d(p'', below(p'')) + 1\}. \tag{6}$$

This completes the analysis of case (ii), so further we may assume that neither of the paths P and Q is entirely contained in a face of $Z(A,B)$.

The case (iii) arises if and only if only one of the pair of vertices p', q'' and p'', q' belongs to a common face of $Z(A,B)$ and this again can be detected by the common_face_test. In the first case,

$$d(P,Q) = \min\{d(p', above(p')) + 1, d(q'', below(q'')) + 1\}. \tag{7}$$

In the second case,

$$d(P,Q) = \min\{d(q', above(q')) + 1, d(p'', below(p'')) + 1\}. \tag{8}$$

Finally, if no condition of cases (i)-(iii) is satisfied, then several faces of $Z(A,B)$ intersect both paths P and Q. Then obviously there is an edge of $E(A,B)$ with one end in P and another end in Q, yielding $d(P,Q) = 1$.

Summarizing, we conclude that from the labels $B_{p'}, B_{p''}, B_{q'}$, and $B_{q''}$ of the end vertices of the paths $P \subseteq \partial A$ and $Q \subseteq \partial B$ we can compute the distance $d(P,Q)$ in $O(1)$ time. We call this subroutine distance_paths.

```
function distance_paths(B_{p'}, B_{p''}, B_{q'}, B_{q''})
  if B_{p'}(1) = 0 then /* rename inputs */ set C := B_{p'}, B_{p'} := B_{q'}, B_{q'} := C
    and C := B_{p''}, B_{p''} := B_{q''}, B_{q''} := C
  if B_{p''}(4) ≤ B_{q'}(3) + B_{q'}(5) then return distance_vertices(B_{p''}, B_{q'})
  if B_{q''}(4) + B_{q''}(6) ≤ B_{p'}(3) then return distance_vertices(B_{p'}, B_{q''})
  if common_face_test(B_{p'}, B_{p''}, 0) then
    if common_face_test(B_{q'}, B_{q''}, 0) then
      return min{distance_vertices(B_{p'}, B_{q'}), distance_vertices(B_{p''}, B_{q''})}
    if common_face_test(B_{p'}, B_{q'}, 1) and not(common_face_test(B_{p'}, B_{q''}, 1)) then
      return min{distance_vertices(B_{p'}, B_{q'}), B_{p''}(4) − B_{p''}(2) + 1}
    if not(common_face_test(B_{p'}, B_{q'}, 1)) and common_face_test(B_{p'}, B_{q''}, 1) then
      return min{distance_vertices(B_{p''}, B_{q''}), B_{p'}(2) − B_{p'}(3) + 1}
    else return min{B_{p'}(2) − B_{p'}(3) + 1, B_{p''}(4) − B_{p''}(2) + 1}
  if common_face_test(B_{q'}, B_{q''}, 0) then
    if common_face_test(B_{p'}, B_{q'}, 1) then
      return min{distance_vertices(B_{p'}, B_{q'}), B_{q''}(4) − B_{q''}(2) + 1}
    if common_face_test(B_{p''}, B_{q'}, 1) then
      return min{distance_vertices(B_{p''}, B_{q''}), B_{q'}(2) − B_{q'}(3) + 1}
    else return min{B_{q'}(2) − B_{q'}(3) + 1, B_{q''}(4) − B_{q''}(2) + 1}
  if common_face_test(B_{p'}, B_{q''}, 1) then
    return min{B_{p'}(2) − B_{p'}(3) + 1, B_{q''}(4) − B_{q''}(2) + 1}
  if common_face_test(B_{p''}, B_{q'}, 1) then
    return min{B_{p''}(4) − B_{p''}(2) + 1, B_{q'}(2) − B_{q'}(3) + 1}
  else return 1.
```

5.2 Computing the distance between $x \in A$ and $y \in B$

In this subsection, we establish how to compute in constant time the distance $d(x, y)$ between two vertices $x \in A$ and $y \in B$, where $\{A, B\}$ is an alternating cut with the zone $Z(A, B)$. We will use the short-hands $P := Pr(x, Z(A, B))$ and $Q := Pr(y, Z(A, B))$. Recall also that for a vertex p and a subset S of G, the *distance* from p to S is $d(x, S) = \min\{d(x, s) : s \in S\}$.

If G is a (4,4)- or a (3,6)-graph, then P and Q are paths and function distance_paths applied to these paths will return $d(P, Q)$. Now, $d(x, y)$ can be computed using the formula $d(x, y) = d(x, Z(A, B)) + d(P, Q) + d(y, Z(A, B))$ whose correctness is provided by Lemmata 4.10 and 4.11. In view of previous results, in order to implement this formula in constant time it suffices to keep at x the distance $d(x, Z(A, B)) := d(x, P)$, the labels $B_{p'}, B_{p''}$ of the end vertices p', p'' of P and at y the distance $d(y, Z(A, B)) := d(y, Q)$ and the labels $B_{q'}, B_{q''}$ of the end vertices q', q'' of Q.

Now suppose that G is a (6,3)-graph, and let $\overline{P}$ and $\overline{Q}$ be the path-completions of the projections P and Q (which are the subpaths of ∂A and ∂B spanned by the end vertices p', p'' and q', q'' of P and Q). From Lemma 4.12 we know that $d(x, Z(A, B))$ is either equal to $\max\{d(x, p'), d(x, p'')\} - 1$ or $d(x, Z(A, B)) = d(x, p') = d(x, p'')$ (an analogous relationship holds for $d(y, Z(A, B)), d(y, q')$, and $d(y, q'')$).

We apply to $\overline{P}$ and $\overline{Q}$ considerations of the previous subsection in order to specify the reciprocal location of the paths $\overline{P}, \overline{Q}$. In case (i), $d(P, Q) = d(\overline{P}, \overline{Q})$ and is realized by one of the pairs p', q'' or p'', q'; consequently $d(x, y) = d(x, p') + d(p', q'') + d(q'', y)$ or $d(x, y) = d(x, p'') + d(p'', q') + d(q', y)$, respectively. In case (ii), we may assume that $\overline{P}$ belongs to a face F of $Z(A, B)$, i.e., P consists of one or two adjacent vertices. If $\overline{Q}$ belongs to the same face of $Z(A,B)$, then

$$d(x,y) = d(x,Z(A,B)) + \min\{d(p',q'), d(p'',q'')\} + d(y,Z(A,B)).$$

If $q' \in F$ and $q'' \notin F$, then, in view of Lemma 4.12, we obtain

$$d(x,y) = d(x,p') + \min\{d(p',q') + d(y,q'), d(p'', below(p'')) + 2 + d(y,Z(A,B))\}$$

(a similar formula holds if $q' \notin F$ and $q'' \in F$). Finally, if $q', q'' \notin F$, then

$$d(x,y) = \min\{d(x,p') + d(p', above(p')) + 2, d(x,p'') + d(p'', below(p'')) + 2\} + d(y,Z(A,B)).$$

In case (iii), exactly one of the pairs p', q'' and p'', q' belongs to a common face of $Z(A,B)$. Thus, P and Q consist of three or more vertices each; see Lemma 4.12. In the first case $d(x,y)$ is the minimum of

$$d(x,p') + d(p', above(p')) + 2 + d(y,Z(A,B))$$

and

$$d(x,Z(A,B)) + d(q'', below(q'')) + 2 + d(y,q''),$$

i.e., either there is a shortest (x,y)-path passing via p' and the next to q'' vertex of Q or there exists a shortest (x,y)-path passing via the next to p' vertex of P and q''. In the second case, $d(x,y)$ is the minimum of

$$d(y,q') + d(q', above(q')) + 2 + d(x,Z(A,B))$$

and

$$d(x,p'') + d(p'', below(p'')) + 2 + d(y,Z(A,B)).$$

Finally, in case (iv) we have $d(P,Q) = 3$, whence $d(x,y) = d(x,Z(A,B)) + 3 + d(y,Z(A,B))$. Therefore, in order to compute $d(x,y)$ we should keep at x the distances $d(x,p'), d(x,Z(A,B)), d(x,p'')$ (some of them can be equal) and the labels $B_{p'}, B_{p''}$ and at y we should keep the distances $d(y,q'), d(y,Z(A,B)), d(y,q'')$ and the labels $B_{q'}, B_{q''}$.

Summarizing all discussions of this subsection, for (4,4),(3,6) and (6,3)-graphs we can retrieve $d(x,y)$ in $O(1)$ time from the label

$$\begin{array}{cccccc} & 1 & 2 & 3 & 4 & 5 \\ D_x := & (d(x,p'), & d(x,Z(A,B)), & d(x,p''), & B_{p'}, & B_{p''}) \end{array}$$

of $x \in A$ and the label

$$\begin{array}{cccccc} & 1 & 2 & 3 & 4 & 5 \\ D_y := & (d(y,q'), & d(y,Z(A,B)), & d(y,q''), & B_{q'}, & B_{q''}) \end{array}$$

of $y \in B$ (see function distance_(3,6)$\vee$(4,4)-graphs(D_x, D_y) and distance_(6,3)-graphs(D_x, D_y) defined below).

function $\mathsf{distance\_(3,6)\vee(4,4)\text{-}graphs}(D_x, D_y)$
return $D_x(2)+$ $\mathsf{distance\_paths}(D_x(4), D_x(5), D_y(4), D_y(5)) + D_y(2)$

function $\mathsf{distance\_(6,3)\text{-}graphs}(D_x, D_y)$
if $B_{p'}(1) = 0$ then /* rename inputs */ $D_x \leftrightarrow D_y$,
extract B'_p and B''_p from D_x
extract B'_q and B''_q from D_y
if $B_{p''}(4) \leq B_{q'}(3) + B_{q'}(5)$ then return $D_x(3)+\mathsf{distance\_vertices}(B_{p''}, B_{q'})+D_y(1)$
if $B_{q''}(4) + B_{q''}(6) \leq B_{p'}(3)$ then return $D_x(1)+\mathsf{distance\_vertices}(B_{p'}, B_{q''})+D_y(3)$
if $\mathsf{common\_face\_test}(B_{p'}, B_{p''}, 0)$ then
if $\mathsf{common\_face\_test}(B_{q'}, B_{q''}, 0)$ then return
$D_x(2) + \min\{\mathsf{distance\_vertices}(B_{p'}, B_{q'}), \mathsf{distance\_vertices}(B_{p''}, B_{q''})\} + D_y(2)$
if $\mathsf{common\_face\_test}(B_{p'}, B_{q'}, 1)$ and $not(\mathsf{common\_face\_test}(B_{p'}, B_{q''}, 1))$ then return
$D_x(1) + \min\{\mathsf{distance\_vertices}(B_{p'}, B_{q'})+D_y(1),\ B_{p''}(4) - B_{p''}(2) + 2 + D_y(2)\}$
if $not(\mathsf{common\_face\_test}(B_{p'}, B_{q'}, 1))$ and $\mathsf{common\_face\_test}(B_{p'}, B_{q''}, 1)$ then return
$D_x(1) + \min\{\mathsf{distance\_vertices}(B_{p''}, B_{q''})+D_y(3),\ B_{p'}(2) - B_{p'}(3) + 2 + D_y(2)\}$
else return $\min\{D_x(1) + B_{p'}(2) - B_{p'}(3) + 2, D_x(3) + B_{p''}(4) - B_{p''}(2) + 2\} + D_y(2)$
if $\mathsf{common\_face\_test}(B_{q'}, B_{q''}, 0)$ then
if $\mathsf{common\_face\_test}(B_{p'}, B_{q'}, 1)$ then return
$D_y(1) + \min\{\mathsf{distance\_vertices}(B_{p'}, B_{q'})+D_x(1),\ B_{q''}(4) - B_{q''}(2) + 2 + D_x(2)\}$
if $\mathsf{common\_face\_test}(B_{p''}, B_{q'}, 1)$ then return
$D_y(1) + \min\{\mathsf{distance\_vertices}(B_{p''}, B_{q''})+D_x(3),\ B_{q'}(2) - B_{q'}(3) + 2 + D_x(2)\}$
else return $\min\{D_y(1) + B_{q'}(2) - B_{q'}(3) + 2, D_y(3) + B_{q''}(4) - B_{q''}(2) + 2\} + D_x(2)$
if $\mathsf{common\_face\_test}(B_{p'}, B_{q''}, 1)$ then return
$\min\{D_x(1) + B_{p'}(2) - B_{p'}(3) + D_y(2), D_x(2) + B_{q''}(4) - B_{q''}(2) + D_y(3)\} + 2$
if $\mathsf{common\_face\_test}(B_{p''}, B_{q'}, 1)$ then return
$\min\{D_x(3) + B_{p''}(4) - B_{p''}(2) + D_y(2), D_y(1) + B_{q'}(2) - B_{q'}(3) + D_x(2)\} + 2$
else return $D_x(2) + 3 + D_y(2)$.

In what follows, the generic name $\mathsf{distance\_graphs}(D_x, D_y)$ will refer to the function $\mathsf{distance\_(3,6)\vee(4,4)\text{-}graphs}(D_x, D_y)$ in the case of (4,4)-graphs and (3,6)-graphs and to the function $\mathsf{distance\_(6,3)\text{-}graphs}(D_x, D_y)$ in the case of (6,3)-graphs.

5.3 Distance decoder

Here we explain how, using the decomposition tree $T(G)$, one can find the distance between any two vertices of G. First, we will describe the labels of vertices of G.

Let v be a median vertex of G (which we assume to be an inner vertex), and let $u_0, \ldots, u_{k-1}$ be its neighbors in counterclockwise order around v. Recall that the cones $C_v(u_i)$, $i \in \{0, \ldots, k-1\}$ of G were defined as follows: $C_v(u_i) = B'_i \cap A'_{i+1(mod\ k)}$. Each vertex $y \in V \setminus C_v(u_i)$ is separated from a vertex $x \in C_v(u_i)$ by zone $Z(A'_i, B'_i)$ or by zone $Z(A'_{i+1}, B'_{i+1})$. From previous results we know that, if two vertices x and y lie in two 1-neighboring or 2-neighboring cones, then $d(x,y)$ is realized via their projections on the zone separating these cones, and if x and y belong to p-neighboring cones with $p > 2$, then $d(x,y)$ is realized via v. For any vertex $x \in C_v(u_i)$ and index $j = i, i+1, i+2\ (mod\ k)$, let D^j_x be the distance label of x with respect to the cut $\{A'_j, B'_j\}$ (D_x was defined in the previous subsection with respect to an arbitrary cut $\{A, B\}$). Let also G_i be a subgraph of G induced by $C_v(u_i)$ ($i \in \{0, \ldots, k-1\}$).

Assume that a decomposition tree $T(G)$ of G and its NCA-depth labeling scheme are given. For a vertex x of G, let $S(x)$ be the deepest node of $T(G)$ containing x and A_x be the label of $S(x)$ in the NCA-depth labeling scheme. Let also $S_0, S_1, \ldots, S_h$ be the nodes of the path of $T(G)$ from the root (G, v) (which is S_0) to the node $S(x) = S_h$.

In the distance labeling scheme for (4,4)-,(3,6)- and (6,3)-graphs, the label $L(x)$ will be the concatenation of A_x, and $h+1$ tuples $\tau_0^x, \tau_1^x, \ldots, \tau_h^x$ where τ_q^x ($q \in \{0, \ldots, h\}$) is defined as follows. Let S_q be a node (G_q, v_q) of $T(G)$. Assume that x belongs to a cone $C_{v_q}(u_i)$ of G_q for some $i \in \{0, ..., \delta_{G_q}(v_q) - 1\}$. Then,

$$\tau_q^x := \overset{1}{(i,} \quad \overset{2}{\delta_{G_q}(v_q),} \quad \overset{3}{d_{G_q}(x, v_q),} \quad \overset{4}{D_x^i,} \quad \overset{5}{D_x^{i+1},} \quad \overset{6}{D_x^{i+2})}$$

where zones and projections are considered in graph G_q. If $x = v_q$, we set $\tau_q^{v_q} := (\delta_{G_q}(v_q), \delta_{G_q}(v_q), 0, 0, 0, 0)$.

Since the depth of $T(G)$ is $O(\log n)$, $L(x)$ is of length $O(\log^2 n)$ bits for any $x \in V$. Note that computation of those tuples can be incorporated into the algorithm of building $T(G)$, leading to an $O(n^2 \log n)$ time computation of all labels $L(x), x \in V$ (for a graph G_q, the paths $Pr(x, Z(A'_j, B'_j))$ ($j = i, i+1, i+2$) and corresponding distances can be computed by running Bread-First-Searches from v_q and $Z(A'_j, B'_j)$ ($j = i, i+1, i+2$)).

Algorithm DISTANCE_DECODER: Distance decoder for (4,4),(3,6) and (6,3)-graphs.

Input: two labels $L(x) = A_x \circ \tau_0^x \circ \tau_1^x \circ \cdots \circ \tau_h^x$ and $L(y) = A_y \circ \tau_0^y \circ \tau_1^y \circ \cdots \circ \tau_q^y$.

Output: $d(x, y)$, the distance between x and y in G.

Method:

use A_x and A_y to find the depth l in $T(G)$ of
  the nearest common ancestor of $S(x)$ and $S(y)$;
extract from $L(x)$ and $L(y)$ the tuples τ_l^x and τ_l^y;
if $\tau_l^x(1) = \tau_l^x(2)$ then output $\tau_l^y(3)$ and stop; /* $x = v_q$ */
if $\tau_l^y(1) = \tau_l^y(2)$ then output $\tau_l^x(3)$ and stop; /* $y = v_q$ */
/* if the cones are 1-neighboring */
if ($\tau_l^x(1) = \tau_l^y(1) - 1$ or $\tau_l^y(1) = 0$ and $\tau_l^x(1) = \tau_l^x(2) - 1$) then output
  distance_graphs($\tau_l^x(5), \tau_l^y(4)$) and stop;
if ($\tau_l^y(1) = \tau_l^x(1) - 1$ or $\tau_l^x(1) = 0$ and $\tau_l^y(1) = \tau_l^x(2) - 1$) then output
  distance_graphs($\tau_l^y(5), \tau_l^x(4)$) and stop;
/* if the cones are 2-neighboring */
if ($\tau_l^x(1) = \tau_l^y(1) - 2$ or $\tau_l^y(1) = 0$ and $\tau_l^x(1) = \tau_l^x(2) - 2$ or
  $\tau_l^y(1) = 1$ and $\tau_l^x(1) = \tau_l^x(2) - 1$) then output
  distance_graphs($\tau_l^x(6), \tau_l^y(4)$) and stop;
if ($\tau_l^y(1) = \tau_l^x(1) - 2$ or $\tau_l^x(1) = 0$ and $\tau_l^y(1) = \tau_l^x(2) - 2$ or
  $\tau_l^x(1) = 1$ and $\tau_l^y(1) = \tau_l^x(2) - 1$) then output
  distance_graphs($\tau_l^y(6), \tau_l^x(4)$) and stop;
else output $\tau_l^x(3) + \tau_l^y(3)$.

5.4 Routing from $x \in A$ to $y \in B$

From Lemma 4.13 we know that any vertex $x \in A$ contains one or two neighbors v_x and u_x such that $I(x, y) \cap \{v_x, u_x\} \neq \emptyset$ for any vertex $y \in B$. Thus the message from x should be forwarded to that of these neighbors which is closer to y. If $x \in A \setminus \partial A$, then $u_x, v_x \in A$ and this decision can be taken in $O(1)$ time by decoding the distances $d(y, v_x)$ and $d(y, u_x)$.

Define $help(v_x)$ to be equal to 1 if x and v_x are separated by the cut $\{A, B\}$ and 0 otherwise ($help(u_x), help(v_y), help(u_y)$ are defined analogously). Then, in this case we can make a routing decision in $O(1)$ time from the label

$$R_x := (\overset{1}{D_x},\ \overset{2}{D_{v_x}},\ \overset{3}{D_{u_x}},\ \overset{4}{port(x, v_x)},\ \overset{5}{port(x, u_x)},\ \overset{6}{help(v_x)},\ \overset{7}{help(u_x)})$$

of x and the label D_y of $y \in B$ (vice versa, to route from $y \in B \setminus \partial B$ to $x \in A$ we need the label

$$R_y := (\overset{1}{D_y},\ \overset{2}{D_{v_y}},\ \overset{3}{D_{u_y}},\ \overset{4}{port(y, v_y)},\ \overset{5}{port(y, u_y)},\ \overset{6}{help(v_y)},\ \overset{7}{help(u_y)})$$

of y and the label D_x of x).

We assert that the same labels R_x and R_y suffice for the routing decision in case $x \in \partial A$. By second assertion of Lemma 4.13, the neighbors u_x and v_x of x either both are vertices of ∂A, or one of them belong to ∂A and another to ∂B, or both are vertices of ∂B. In the first case, both distances $d(u_x, y)$ and $d(v_x, y)$ can be decoded as before. In the second case only the distance from y to the vertex of ∂A can be decoded using R_x and R_y, say $d(u_x, y)$. If $d(x, y) = d(u_x, y) + 1$, then the message is forwarded to u_x, otherwise it is sent to v_x. Finally, if $u_x, v_x \in \partial B$, then G is a (3,6)-graph and the routing decision can be taken by employing the items $d(b, q'), d(b, q'')$ of D_y (here q' and q'' are the end vertices of $Pr(y, Z(A, B))$. Namely, if u_x is above v_x, then the message is forwarded to u_x if $d(b, q'') \le d(b, u_x)$ and to v_x otherwise.

function routing_decision(R_x, R_y)
if $R_x(6) \neq 1$ then
  if distance_graphs$(R_x(1), R_y(1))$ = distance_graphs$(R_x(2), R_y(1)) + 1$ then output $R_x(4)$
  else output $R_x(5)$
else if $R_x(7) \neq 1$ then
    if distance_graphs$(R_x(1), R_y(1))$ = distance_graphs$(R_x(3), R_y(1)) + 1$ then output $R_x(5)$
    else output $R_x(4)$
  else extract $B_{q''}$ from $R_y(1)$
    extract B_{u_x} from $R_x(3)$
    if $B_{q''}(2) \le B_{u_x}(2)$ then output $R_x(5)$ else output $R_x(4)$

5.5 Routing decision

Here we explain how, using the decomposition tree $T(G)$, one can rout between any two vertices of G. The method is very similar to the one we used for distance decoding.

Let again v be a median vertex of G and $u_0, \ldots, u_{k-1}$ be its neighbors in counterclockwise order around v. For any vertex $x \in C_v(u_i)$ and index $j = i, i+1, i+2 \ (mod\ k)$, denote by R_x^j the routing label of x with respect to the cut $\{A'_j, B'_j\}$ (R_x was defined in the previous subsection with respect to an arbitrary cut $\{A, B\}$). Let $S(x)$ be the deepest node of the decomposition tree $T(G)$ of G containing x and A_x be the label of $S(x)$ in the NCA-depth labeling scheme of $T(G)$. Let also $S_0, S_1, \ldots, S_h$ be the nodes of the path of $T(G)$ from the root (G, v) (which is S_0) to the node $S(x) = S_h$. Denote as before by G_i a subgraph of G induced by $C_v(u_i)$ ($i \in \{0, \ldots, k-1\}$).

In the routing labeling scheme for (4,4)-,(3,6)- and (6,3)-graphs, the label $L(x)$ will be the concatenation of A_x, and $h+1$ tuples $\mu_0^x, \mu_1^x, \ldots, \mu_h^x$ where μ_q^x ($q \in \{0, \ldots, h\}$) is defined as follows. Let S_q be a node (G_q, v_q) of $T(G)$. Assume that x belongs to a cone $C_{v_q}(u_i)$ of G_q for some $i \in \{0, ..., \delta_{G_q}(v_q) - 1\}$. Then,

$$\mu_q^x := \overset{1}{(i,} \quad \overset{2}{\delta_{G_q}(v_q),} \quad \overset{3}{port_{G_q}(x, v_q),} \quad \overset{4}{R_x^i,} \quad \overset{5}{R_x^{i+1},} \quad \overset{6}{R_x^{i+2},} \quad \overset{7}{port_{G_q}(v_q, x))}.$$

If $x = v_q$, we set $\mu_q^{v_q} := (\delta_{G_q}(v_q), \delta_{G_q}(v_q), 0, 0, 0, 0, 0)$.

Clearly, again $L(x)$ is of length $O(\log^2 n)$ bits for any $x \in V$ and computation of those tuples can be incorporated into the algorithm for building $T(G)$, leading to an $O(n^2 \log n)$ time computation of all labels $L(x), x \in V$ (for a vertex x of a graph G_q, the special neighbors v_x^j and u_x^j can be computed by running Bread-First-Searches from $Z(A'_j, B'_j)$ $(j = i, i+1, i+2)$).

Algorithm ROUTING_DECISION: Routing decision for (4,4),(3,6) and (6,3)-graphs.
Input: two labels $L(x) = A_x \circ \mu_0^x \circ \mu_1^x \circ \cdots \circ \mu_h^x$ and $L(y) = A_y \circ \mu_0^y \circ \mu_1^y \circ \cdots \circ \mu_q^y$.
Output: $port_G(x, y)$, the output port number of the first edge on a shortest path from x to y in G.
Method:

use A_x and A_y to find the depth l in $T(G)$ of
  the nearest common ancestor of $S(x)$ and $S(y)$;
extract from $L(x)$ and $L(y)$ the tuples μ_l^x and μ_l^y;
if $\mu_l^x(1) = \mu_l^x(2)$ then output $\mu_l^y(7)$ and stop; /* $x = v_q$ */
if $\mu_l^y(1) = \mu_l^y(2)$ then output $\mu_l^x(3)$ and stop; /* $y = v_q$ */
/* if the cones are 1-neighboring */
if $(\mu_l^x(1) = \mu_l^y(1) - 1$ or $\mu_l^y(1) = 0$ and $\mu_l^x(1) = \mu_l^x(2) - 1)$ then output
  routing_decision$(\mu_l^x(5), \mu_l^y(4))$ and stop;
if $(\mu_l^y(1) = \mu_l^x(1) - 1$ or $\mu_l^x(1) = 0$ and $\mu_l^y(1) = \mu_l^x(2) - 1)$ then output
  routing_decision$(\mu_l^x(4), \mu_l^y(5))$ and stop;
/* if the cones are 2-neighboring */
if $(\mu_l^x(1) = \mu_l^y(1) - 2$ or $\mu_l^y(1) = 0$ and $\mu_l^x(1) = \mu_l^x(2) - 2$ or
  $\mu_l^y(1) = 1$ and $\mu_l^x(1) = \mu_l^x(2) - 1)$ then output
  routing_decision$(\mu_l^x(6), \mu_l^y(4))$ and stop;
if $(\mu_l^y(1) = \mu_l^x(1) - 2$ or $\mu_l^x(1) = 0$ and $\mu_l^y(1) = \mu_l^x(2) - 2$ or
  $\mu_l^x(1) = 1$ and $\mu_l^y(1) = \mu_l^x(2) - 1)$ then output
  routing_decision$(\mu_l^x(4), \mu_l^y(6))$ and stop;
else output $\mu_l^x(3)$.

5.6 Squaregraphs, cellular graphs, and trigraphs

In this subsection we point out that for three special subclasses of (4,4)- and (3,6)-graphs, namely planar cellular graphs, squaregraphs and trigraphs, the described labeling schemes can be greatly simplified based on additional geometric properties these classes possess. Recall that a *squaregraph* is a (4,4)-graph in which all inner faces are quadrangles, a planar *cellular graph* is a bipartite (4,4)-graph, and a *trigraph* is a (3,6)-graph in which all inner faces have length 3.

The following additional properties of those graphs are of use.

(i) *In bipartite (3,6)-,(4,4)-, or (6,3)-graphs, the two alternating cuts crossing an edge uv of G coincide and have the form $\{W(v, u), W(u, v)\}$. Since all faces are of even length, there are no turns in cuts and, therefore, all cut edges are horizontal in zones.*

Hence, no α-entries and no $r(p)$-, $s(p)$-entries will appear in B_p and B_q labels, simplifying the labels and the functions.

(ii) *In cellular graphs, in particular, in squaregraphs, the zones are gated, i.e. the projection of each vertex x to a zone $Z(A,B)$ is a singleton.*

Hence, $p' = p''$ and, therefore, only one out of the first three entries and only one out of the last two entries in the D_x labels will remain. Furthermore, $v_x = u_x$ and, hence, the entries D_{u_x}, $port(x,u_x)$ and $help(u_x)$ in the routing labels R_x will disappear. This all will simplify the labels as well as the functions (e.g, function distance_paths is not needed). For squaregraphs, even $help(v_x)$ entry can be omitted since each vertex $x \in \partial A$ has one (and only one) neighbor in B, which is v_x. Thus, for squaregraphs, the distance between any vertices $x \in A$ and $y \in B$ can be computed using the formula

$$d(x,y) = d(x,p') + d(y,q') + 1 + |d(p',a) - d(q',b)|,$$

and, therefore, algorithms DISTANCE_DECODER and ROUTING_DECISION can be implemented without any auxiliary functions.

(iii) *For squaregraphs, a stronger version of Lemma 4.16 holds: If $x \in C_v(u_i)$ and $y \in C_v(u_j)$ and the cones $C_v(u_i)$ and $C_v(u_j)$ are not 1-neighboring, then $d(x,y) = d(x,v) + d(v,y)$.*

Thus, in tuples τ_q^x and μ_q^x, the entries D_x^{i+2} and R_x^{i+2} will disappear. Also, the algorithms DISTANCE_DECODER and ROUTING_DECISION will be further simplified since the case of 2-neighboring cones will be omitted.

(iv) *In trigraphs each vertex of a zone has two twins (except maybe the boundary vertices), thus the definition of ϵ in formula (1) can be simplifed as follows*

$$\epsilon = \begin{cases} 0 & \textit{if } d(q,b) < d(p,a), \\ 1 & \textit{otherwise.} \end{cases}$$

and the distance between two paths $P \subseteq \partial A$ and $Q \subseteq \partial B$ with end-vertices p', p'' and q', q'' is given by

$$d(P,Q) = \begin{cases} d(p'',q') & \textit{if } P \textit{ is above } Q, \\ d(p',q'') & \textit{if } Q \textit{ is above } P, \\ 1 & \textit{otherwise.} \end{cases}$$

Therefore, only the first two entries of labels B_p and B_q are necessary. Moreover, using the simplified formula given above, one can avoid the auxiliary functions distance_vertices, distance_paths and implement directly the algorithms DISTANCE_DECODER and ROUTING_DECISION.

(v) *In squaregraphs and trigraphs, a median vertex can be found in linear time [16].*

Hence, we need only $O(n \log n)$ time for construction of a decomposition tree $T(G)$ and computation of all labels $L(x)$, $x \in V$. In each level of recursion, we need to find median vertices of current subgraphs, construct the corresponding cones and compute $(1/\log n)$th part of each label $L(x)$. This can be done in $O(n)$ total time for both squaregraphs and trigraphs. Since the recursion depth is $O(\log n)$, we get our $O(n \log n)$ time bound.

We have described the resulting simplified distance and routing labeling schemes in details for squaregraphs in the conference version of this paper.

6 Main result

From the properties of (4,4)-,(3,6)-, and (6,3)-graphs and the labeling schemes described in Sections 3 and 5, we deduce the main contribution of this paper.

Theorem 6.1 *The family of graphs of type (4,4),(3,6), and (6,3) with at most n vertices admits distance and routing labeling schemes with labels of size $O(\log^2 n)$ bits and a constant time distance decoder and routing decision. Moreover, the schemes are constructable in time $O(n^2 \log n)$, and in $O(n \log n)$ time for squaregraphs and trigraphs.*

References

[1] B. Awerbuch, A. Bar-Noy, N. Linial, D. Peleg, Improved routing strategies with succinct tables, J. Algorithms, 11(1990), 307–341.

[2] B. Awerbuch, D. Peleg, Routing with polynomial communicationspace tradeoff, SIAM J. Discrete Math. 5(1992), 151–162.

[3] E.M. Bakker, J. van Leeuwen, R.B. Tan, Linear interval routing, Algorithms Review., 2 (1991), 45-61.

[4] H.-J. Bandelt, Networks with Condorcet solutions, Europ. J. Oper. Res., 20 (1985), 314-326.

[5] H.-J. Bandelt, V. Chepoi, Cellular bipartite graphs, Europ. J. Combin., 17 (1996), 121–134.

[6] H.-J. Bandelt, V. Chepoi, Decomposition and l_1-embedding of weakly median graphs, Europ. J. Combin., 21 (2000), 701–714.

[7] H.-J. Bandelt, V. Chepoi, The algebra of metric betweenness I: subdirect representation, retractions, and equational characterization of weakly median graphs (submitted)

[8] H.-J. Bandelt, V. Chepoi, in preparation.

[9] O. Baues, N. Peyerimhoff, Curvature and geometry of tessellating plane graphs, Discrete Comput. Geom. 25 (2001), 141-159.

[10] O. Baues, N. Peyerimhoff, Geodesics in non-positively curved plane tesselations (submitted).

[11] B. Brešar, W. Imrich, S. Klavžar, H.M. Mulder, R. Škrekovski, Tiled partial cubes, J. Graph Th., 40 (2002), 91-103.

[12] M.A. Breuer, Coding the vertices of a graph, IEEE Trans. Inf. Th., IT-12 (1966), 148–153.

[13] M.A. Breuer, J. Folkman, An unexpected result on coding the vertices of a graph, J. Math. Anal. and Appl., 20 (1967), 583–600.

[14] V. Chepoi, M. Deza, V. Grishukhin, Clin d'oeil on L_1-embeddable planar graphs, Discr. Appl. Math. 80 (1997), 3–19.

[15] V. Chepoi, F. Dragan, Y. Vaxès, Center and diameter problems in plane triangulations and quadrangulations, in: Proc. 13th Ann. ACM-SIAM Symp. Discrete Algorithms (SODA 2002), ACM–SIAM 2002, pp.346-355.

[16] V. Chepoi, C. Fanciullini, Y. Vaxès, Median problem in some plane triangulations and quadrangulations, manuscript, 2002.

[17] V. Chepoi, A. Rollin, Interval routing in some planar networks, Theoret. Comput. Sc. 290 (2003), 1503–1540.

[18] L. Cowen, Compact routing with minimum stretch, in: Proc. 10th Ann. ACM-SIAM Symp. Discrete Algorithms (SODA 1999), ACM–SIAM 1999, pp. 255–260.

[19] Y. Dourisboure, C. Gavoille, Improved Compact Routing Scheme for Chordal Graphs, in: Proc. 16th Int. Conf. Distr. Comput. (DISC 2002), in: Lecture Notes in Comput. Sci. Vol. 2508, 2002, pp. 252-264.

[20] A.W.M. Dress, R. Scharlau, Gated sets in metric spaces, Aeq. Math. 34 (1987), 112–120.

[21] T. Eilam, C. Gavoille, D. Peleg, Compact routing schemes with low stretch factor, in: Proc. 17th Ann. ACM Symp. Prin. Distr. Comput. (PODC 1998), ACM 1998, pp. 11–20.

[22] P. Fraigniaud, C. Gavoille, Memory requirements for univesal routing schemes, in Proc. 14th Ann. ACM Symp. Prin. Distr. Comput. (PODC 1995), ACM, 1995, pp. 223–230.

[23] P. Fraigniaud, C. Gavoille, Interval routing schemes, Algorithmica, 21 (1998), 155–182.

[24] P. Fraigniaud, C. Gavoille, Routing in Trees, Proc. 28th Int. Colloquium on Automata, Languages and Programming (ICALP 2001), in: Lecture Notes in Comput. Sci.Vol. 2076, 2001, pp. 757–772.

[25] P. Fraigniaud, C. Gavoille, A Space Lower Bound for Routing in Trees, Proc. 19th Ann. Symp. Theoret. Aspects Comput. Sci. (STACS 2002), in: Lecture Notes in Comput. Sci. Vol. 2285, 2002, pp. 65-75.

[26] G.N. Frederickson, R. Janardan, Designing networks with compact routing tables, Algorithmica, 3 (1988), 171–190.

[27] C. Gavoille, A survey on interval routing schemes, Theoret. Comput. Sci., 245 (1999), 217–253.

[28] C. Gavoille, M. Gengler, Space-efficiency of routing schemes of stretch factor three, J. Parallel and Distr. Comput., 61 (2001), 679-687.

[29] C. Gavoille, N. Hanusse, Compact Routing Tables for Graphs of Bounded Genus, Proc. 26th Int. Colloquium on Automata, Languages and Programming (ICALP 1999), in: Lecture Notes in Comput. Sci. Vol. 1644, 1999, pp. 351-360.

[30] C. Gavoille, M. Katz, N.A. Katz, C. Paul, D. Peleg, Approximate Distance Labeling Schemes, Proc. 9th Ann. European Symp. Algorithms (ESA 2001), in: Lecture Notes in Comput. Sci. Vol. 2161, 2001, pp. 476-487.

[31] C. Gavoille, C. Paul, Split decomposition and distance labeling: an optimal scheme for distance hereditary graphs, in: Proc. Euro. Conf. Combinatorics, Graph Theory and Applications, Sep. 2001.

[32] C. Gavoille, C. Paul, Optimal distance labeling schemes, Proc. 11th Annual European Symp. Algorithms (ESA 2003), in: Lecture Notes in Comput. Sci. Vol. 2832, 2003, to appear.

[33] C. Gavoille, D. Peleg, S. Pérennes, R. Raz, Distance labeling in graphs, Proc. 12th Ann. ACM–SIAM Symp. Discrete Algorithms (SODA 2001), ACM–SIAM 2001, pp. 210–219.

[34] C. Gavoille, S. Pérennès, Memory requirements for routing in distributed networks, Proc. 15th Ann. ACM Symp. on Prin. Distr. Comput. (PODC 1996), ACM 1996, pp. 125–133.

[35] D. Harel, R. Tarjan, Fast algorithms for finding nearest common ancestor, SIAM J. Comput., 13 (1984), 338–355.

[36] S. Kannan, M. Naor, S. Rudich, Implicit representation of graphs, Proc. 20th Symp. Theory of Computing (STOC 1988), ACM 1988, pp. 334–343.

[37] M. Katz, N.A. Katz, A. Korman, D. Peleg, Labeling schemes for flow and connectivity, in Proc. 13th Ann. ACM–SIAM Symp. Discrete Algorithms (SODA 2002), ACM–SIAM 2002, pp. 927-936.

[38] M. Katz, N.A. Katz, D. Peleg, Distance labeling schemes for well-separated graph classes, Proc. 17th Ann. Symp. Theoret. Aspects of Comput. Sci. (STACS 2000), in: Lecture Notes in Comput. Sci. Vol. 1770, 2000, pp. 516–528.

[39] J. van Leeuwen, R.B. Tan, Computer networks with compact routing tables, in The book of L, Springer-Verlag, 1986.

[40] J. van Leeuwen, R.B. Tan, Interval routing, The Computer J., 30 (1987), 298–307.

[41] R.C. Lyndon, On Dehn's algorithm, Math. Annalen, 166 (1966), 208–228.

[42] R.C. Lyndon, A maximum principle for graphs, J. Combin. Th. 3 (1967), 34-37.

[43] R.C. Lyndon, P.E. Schupp, Combinatorial Group Theory, Springer-Verlag, Berlin/Heidelberg/New York, 1977.

[44] L. Narayanan, N. Nishimura, Interval routing on k-trees, J. Algorithms, 26 (1998), 325–369.

[45] L. Narayanan, S. Shende, Partial characterizations of networks supporting shortest path interval labeling schemes, Networks, 32 (1998), 103–113.

[46] D. Peleg, Proximity-Preserving Labeling Schemes and Their Applications, Proc. 25th Int. Workshop on Graph-Theoretic Concepts in Computer Science (WG 1999), in: Lecture Notes in Comput. Sci. Vol. 1665, 1999, pp. 30-41.

[47] D. Peleg, Informative Labeling Schemes for Graphs, Proc. 25th Int. Symp. on Math. Foundations of Computer Science (MFCS 2000), in: Lecture Notes in Comput. Sci. 1893, 2000, pp. 579-588.

[48] D. Peleg, Distributed computing – A localitysensitive approach, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 2000.

[49] D. Peleg, E. Upfal, A tradeoff between space and efficiency for routing tables, J. ACM, 36(1989), 510–530.

[50] C. Prisăcaru, P. Soltan, V. Chepoi, On embedding of planar graphs into hypercubes, Proc. Moldavian Acad. of Sci., ser. Math., 1 (1990), 43-50 (in Russian).

[51] N. Santoro, R. Khatib, Labeling and implicit routing in networks, The Computer J., 28 (1985), 5–8.

[52] J. P. Spinrad, Efficient Graph Representations, Fields Institute Monographs # 19, AMS, 2003.

[53] P. Soltan, D. Zambitski, C. Prisăcaru, Extremal problems on graphs and algorithms of their solution (in Russian), Stiinţa Chişinău, 1973.

[54] M. Thorup, Compact oracles for reachability and approximate distances in planar digraphs, FOCS 2001, pp. 242-251

[55] M. Thorup, U. Zwick, Approximate distance oracles, Proc. 33^{rd} Ann. ACM Symp. on Theory of Computing (STOC 2001), ACM, 2001, pp. 183–192.

[56] M. Thorup, U. Zwick, Compact routing schemes, in: Proc. 13th Ann. ACM Symp. on Par. Alg. and Arch. (SPAA 2001), ACM, 2001, pp. 1–10.

[57] A.Żuk, On the norms of the random walks on planar graphs, Ann. Inst. Fourier, 47 (1997), 1463–1490.