Local-to-Global Aspects in Metric Graph Theory and Distributed Computing

Jérémie Chalopin

Introduction

The notion of *coverings* is a fundamental notion that comes from (algebraic) topology [**137**]. It enables to express that two topological spaces are "locally similar". Its importance in algebraic topology comes from the fact that coverings are an important tool in the study of homotopy groups and, in particular, of the fundamental group of a space. In the case of graphs and cell complexes that we consider in this document (but also in many other non-pathological cases such as manifolds), every space X has a *universal cover* $\widetilde{X}$, i.e., a cover $\widetilde{X}$ that covers every space Y such that Y covers X. The universal cover $\widetilde{X}$ of X is the "largest" cover of X and $\widetilde{X}$ is the unique cover of X that is simply connected.

Local-to-global results are common and of great importance in geometry and originates with the classical Cartan-Hadamard theorem (see [**42**, **198**]) in Riemannian geometry relating the non-positive sectional curvature of a Riemannian manifold to the properties of its universal cover. This theorem was generalized by Gromov [**128**] to all geodesic metric spaces of non-positive curvature: if a geodesic metric space is locally non-positively curved, then its universal cover is globally non-positively curved (a CAT(0) space). This local-to-global approach enables to obtain characterizations of metric spaces via the properties of their distance functions, and also to construct such spaces starting from compact (or finite) spaces in which these properties are satisfied only locally.

The main subject of metric graph theory is the investigation and structural characterization of graph classes whose standard graph-metric satisfies the main metric and convexity properties of classical metric geometries like $\mathbb{R}^n$ endowed with l_2, l_1, or l_∞–metric, hyperbolic spaces, Boolean spaces, or trees. This gives rise to many graph classes such as Helly graphs, bridged graphs, hyperbolic graphs, median graphs, modular graphs, etc.; see [**18**] for a survey. It turned out that some of these graphs give rise to important cubical and simplicial complexes that can be viewed as geometric objects that enjoy many nice properties. Some of these complexes correspond to classical objects studied in modern geometric group theory such as CAT(0) cube complexes [**128**] or systolic complexes [**146**]. Local-to-global characterization of these graphs/complexes enable to construct infinite complexes in these classes by considering the universal covers of some finite complexes.

Since the seminal work of Angluin [**9**], covers of graphs have been used in distributed computing in order to express indistinguishability between processes in a network and establish impossibility results in distributed computing. The lifting lemma technique of Angluin has since been used in different distributed models such as message passing systems [**31–33**, **71**, **72**, **259**, **260**], local computation models [**49**, **170**], and mobile agent systems [**62**, **68**, **69**, **75**, **99**]. An important notion in these works is the notion of *minimum base* [**34**] (or quotient graph). While the universal cover is the "largest" cover of a graph/complex, the minimum base of a graph G is the "smallest" graph that is covered by G. The minimum base of a graph can be constructed from the *views* of the vertices of the graph that correspond to a rooted version of the universal cover of the graph. This minimum base encodes all the information the nodes of the network (or an agent evolving in a network) can gather about the network.

The work presented in this document is based on some of the research I have done since I arrived in Marseille in 2007. It is based on the results (or part of the results) published in the papers [**39**, **50**–**54**, **56**–**61**, **63**–**68**, **74**–**76**].

The first part of this document gather some results obtained in metric graph theory and related fields. In the first chapter, we present local-to-global characterizations for weakly modular graphs, basis graphs of matroids (resolving a conjecture by Maurer [**169**]), Helly graphs (answering a question raised independently by Prisner [**205**], Chepoi (unpublished), and Larrión et al. [**158**]), bucolic graphs, prime pre-median graphs (answering a question of Chastand [**78**, **79**]), and dual-polar graphs (allowing to provide a different and simpler proof of a difficult result by Brouwer and Cohen [**44**]).

In the second chapter, using the nice bijections between median graphs, CAT(0) cube complexes, and event structures, we provide counterexamples to two conjectures of Thiagarajan on event structures [**240**, **241**]. We also provide some positive results about these conjectures and establish in particular a bijection between special cube complexes (introduced by Wise and Haglund [**132**, **133**] in geometric group theory) and trace regular event structures that correspond exactly to the unfolding of 1-safe Petri nets [**240**].

In the third chapter, we study δ-hyperbolic graphs and provide a characterization of these graphs that is based on a cop and robber game. From this characterization, we derive an algorithm computing an approximation of the hyperbolicity of a graph in optimal $O(n^2)$ time (when the graph is given by its distance matrix). This algorithm is quite simple but its approximation factor is huge (1569). We also provide a $O(n^2)$ algorithm that compute an 8-approximation of the hyperbolicity of a graph. This algorithm is based on the introduction of a new graph parameter related to the hyperbolicity.

In the fourth chapter, we consider maximum and ample [**19**, **110**] (a.k.a. lopsided [**159**] or extremal [**35**]) classes that have been considered by people from computational machine learning [**119**, **155**, **174**, **216**, **217**]. Ample classes correspond to a subclass of partial cubes and generalize median graphs. We design unlabeled sample compression scheme for maximum classes and characterize such schemes for ample classes in a local-to-global way via representation maps and unique sink orientations. We also construct an example of a maximum class of dimension 3 without corners. This refutes several previous works in machine learning [**108**, **155**, **216**] and it implies that all previous constructions of optimal unlabeled sample compression schemes for maximum classes are erroneous.

In the second part of this document, we consider a very simple distributed model where an agent is evolving in a graph and wants to gather information about the underlying network. In the fifth chapter, we study what graph can be explored by an agent that terminates once it has explored all nodes. After recalling the well-known results in a classical model, we consider a model where the agent can gather local information at each node. Even if there are some similarity in the characterizations of explorable graphs in the two models, the algorithmic techniques used are much more involved and the proofs rely on some technical topological arguments. It also turns out that the complexity of the problem becomes unboundable in this new model. However, we identify a large subclass of explorable graphs (containing many of the classes considered in the first part of this document) that can be explored in linear time.

In the sixth chapter, we assume that the agent knows an upper bound on the size of the network. With this initial knowledge at hand, the agent can explore the network and halt. We show that it can build the minimum base of the network (and thus gather all information it can about it) and that the overhead in the complexity to compute it (compared to the exploration problem) is polynomial in the size of the minimum base.

In the seventh chapter, we consider the mapping problem where the agent aims at reconstructing a map of the underlying graph. This problem cannot be solved in general, but we show that if the agent is evolving in the visibility graph of a polygon, it can always reconstruct this graph, provided the port-numbers allowing the agent to navigate within the graph are assigned locally in a geometric way.

At the end of each chapter, a conclusion presents some related open questions and research directions.

Contents

Part 1

Topological and Geometrical Methods in Metric Graph Theory and Concurrency

CHAPTER 1

Local-to-Global Characterizations of Metric Graph Classes

Local-to-global results are common and of great importance in geometry and originates with the classical Cartan-Hadamard theorem (see [**42**, **198**]) in Riemannian geometry relating the non-positive sectional curvature of a Riemannian manifold to the properties of its universal cover. This theorem was generalized by Gromov [**128**] to all geodesic metric spaces of non-positive curvature: if a geodesic metric space is locally non-positively curved, then its universal cover is globally non-positively curved (a CAT(0) space). Analogously, Myers [**179**] characterized spheres by means of positive curvature. This local-to-global approach enables to obtain characterizations of metric spaces via the properties of their distance functions, and also to construct such spaces starting from compact (or finite) spaces in which these properties are satisfied only locally. This also allows to establish global topological conditions such as contractibility.

In the particular case of cube complexes, Gromov reformulated his general result as a combinatorial condition on the links of the vertices [**128**]: a cube complex X is CAT(0) if and only if X is simply connected and the links vertices of X are flag simplicial complexes. Consequently, the universal cover of a cube complex where links are flag is CAT(0). This key observation is one of the founding results in modern geometric group theory [**42**, **218**] and lead to many deep results [**4**, **132**, **133**, **257**, **258**].

The main subject of metric graph theory is the investigation and structural characterization of graph classes whose standard graph-metric satisfies the main metric and convexity properties of classical metric geometries like $\mathbb{R}^n$ endowed with l_2, l_1, or l_∞–metric, hyperbolic spaces, Boolean spaces, or trees. Among such properties one can mention convexity of balls or of neighborhoods of convex sets, Helly property for balls, isometric and low-distortion embeddings into classical host spaces, retractions, various four-point conditions, uniqueness or existence of medians, etc.; for a survey of this theory, see [**18**].

Later, it turned out that some of these graphs give rise to important cubical and simplicial complexes that can be viewed as geometric objects. In fact, it turns out that the 1-skeletons of CAT(0) cube complexes are exactly the median graphs [**82**, **214**], one of the main classes of graphs studied in metric graph theory [**18**, **136**]. Similarly to the local-to-global characterization of CAT(0) cubical complexes of [**128**], it was shown in [**82**] that the clique complexes of bridged graphs [**117**, **231**] are exactly the simply connected simplicial flag complexes in which the links of vertices do not contain induced 4- and 5-cycles. These complexes have been rediscovered and investigated in depth in the geometric group theory community by Januszkiewicz and Swiatkowski [**146**], by Haglund [**130**], and by Wise [**256**], who called them "systolic complexes" and considered them as simplicial complexes satisfying combinatorial nonpositive curvature property. These two results of [**82**] are obtained using minimal disk diagrams whose existence are ensured by the simple connectivity of the considered complexes.

In a series of papers [**39**, **56**, **60**], we established local-to-global characterizations for several other classes of graphs defined by their metric properties. Among other results, we characterized in such a way basis graphs of matroids (resolving a conjecture by Maurer [**169**] from 1973), bucolic graphs (a common generalization of bridged/systolic and median/CAT(0) graphs), Helly graphs (answering a question raised independently by Prisner [**205**], Chepoi (unpublished), and Larrión et al. [**158**]), dual-polar graphs of

Cameron [**47**] (allowing to provide a different and simpler proof of a difficult result by Brouwer and Cohen [**44**]), prime pre-median graphs (providing the answer to a question of Chastand [**78, 79**]). We also establish a general local-to-global characterization of weakly modular graphs (containing the last four classes of graphs).

A major difference between these classes and median and bridged graphs is that the 2-dimensional faces of the associated complexes consist of two types of cells (triangles and squares), while for median and bridged graphs, the associated complexes have only one type of 2-dimensional cells (squares and triangles respectively). This makes the techniques based on minimal disk diagrams very difficult to implement. Instead, in [**39, 56, 60**], we developed a general approach based on the level-by-level construction of the universal cover of the associated complex. Depending on the specific local conditions for each case, this approach is implemented differently.

We exemplify this by formulating three main results of this chapter for Helly graphs, basis graphs of matroids, and weakly modular graphs (we defer all formal definitions to respective sections).

Helly graphs are the graphs where the balls satisfy the Helly property, i.e., any collection of pairwise intersecting balls has a non-empty intersection. They are discrete analogues of geodesic injective spaces [**110, 144**]. Similarly to injective hulls, any graph isometrically embeds into a unique smallest Helly graphs. Finite Helly graphs have been characterized in several ways by Bandelt, Pesch, and Prisner [**23, 24**] leading to polynomial time algorithms to recognize them. It was conjectured (Prisner [**205**], Chepoi (unpublished), and Larrión et al. [**158**]) that Helly graphs are exacly the graphs with simply connected clique-complexes where the maximal cliques satisfy the Helly property, i.e., clique-Helly graphs with simply connected clique-complexes. Since being clique-Helly is a local condition, this is a local-to-global characterization of Helly graphs. The following theorem of [**56**] establishes this result.

THEOREM 1.1. *Let G be a clique-Helly graph and let $\widetilde{G}$ be the 1–skeleton of the universal cover $\widetilde{X}_{\triangle}(G)$ of the triangle complex $X_{\triangle}(G)$ of G. Then $\widetilde{G}$ is a Helly graph. In particular, G is a Helly graph if and only if G is clique-Helly and its triangle complex (and thus its clique complex) is simply connected.*

Matroids constitute an important unifying structure in combinatorics, algorithmics, and combinatorial optimization — cf. e.g. [**194**] and references therein. One of the standard way to represent a matroid is via its basis graph: the vertices of this graph are the bases of the matroids and two bases are adjacent if they differ by an elementary exchange. Basis graphs faithfully represent their matroids [**141, 169**], thus studying the basis graph amounts to studying the matroid itself. Moreover, Gelfand et al. [**124**] showed that the 1–skeleton of a basis matroid polyhedron coincides with the basis graph of the matroid.

In his seminal paper [**169**, Theorem 2.1], Maurer characterized the basis graphs of matroids as connected graphs satisfying three conditions: the interval condition, the link condition, and the positioning condition. The first two conditions are local conditions while the positioning condition is a global metric condition (in the spirit of the triangle and quadrangle conditions for weak modularity considered below). He conjectured that the link condition is redundant and that the positioning condition can be replaced by the simple connectivity of the associated triangle-square complex. In [**60**], we proved both conjectures. In particular, we establish the following local-to-global characterization of basis graphs of matroids.

THEOREM 1.2. *Let G be a connected graph such that for every vertex v, the ball of radius 3 around v is isomorphic to a ball of radius 3 of the basis graph of a matroid. Then the 1–skeleton of the universal cover $\widetilde{X}_{\triangle\square}(G)$ of its triangle-square complex $X_{\triangle\square}(G)$ is* *the basis graph of a matroid. In particular, G is the basis graph of a matroid if and only if G is simply connected and every ball of radius 3 in G is isomorphic to a ball of radius 3 in the basis graph of a matroid.*

Weak modularity is defined using two global metric conditions: the triangle and the quadrangle conditions. Weakly modular graphs have been introduced in [**16**, **80**] as a common far-reaching generalization of median, bridged, Helly, modular, and pseudo-modular graphs. Similarly to basis graphs of matroids, weakly modular graphs can be characterized by requiring that every ball of radius 3 is isomorphic to a ball of radius 3 in a weakly modular graph [**56**]:

THEOREM 1.3. *Let G be a graph where triangle and quadrangle conditions are satisfied at distance at most 3, and let $\widetilde{G}$ be the 1–skeleton of the universal cover $\widetilde{X}_{\triangle\square}(G)$ of the triangle-square complex $X_{\triangle\square}(G)$ of G. Then $\widetilde{G}$ is weakly modular. In particular, a graph G is a weakly modular graph if and only if G satisfies the triangle and quadrangle conditions at distance at most 3 and the triangle-square complex $X_{\triangle\square}(G)$ is simply connected.*

The results of this chapter on based on the papers [**39**], [**56**] and [**60**].

1. Preliminaries

In this document, unless stated otherwise, all graphs $G = (V, E)$ are finite, undirected, simple, and connected; V is the vertex-set and E is the edge-set of G. We write $u \sim v$ if two vertices u and v are adjacent. 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* $I(u, v) = \{x \in V : d(u, x) + d(x, v) = d(u, v)\}$ consists of all the vertices on shortest (u, v)–paths. A subgraph H of G is called *isometric* if $d_H(u, v) = d_G(u, v)$ for any two vertices u, v of H. A subgraph H is called *convex* if $I(u, v) \subseteq H$ for any two vertices u, v of H.

All complexes we consider are CW complexes. Following [**137**, Chapter 0], we call them simply *cell complexes* or just *complexes*. If all cells are simplices and the nonempty intersections of two cells is their common face, then X is called a *simplicial complex*. For a cell complex X, by $X^{(k)}$ we denote its *k–skeleton*. The cell complexes we consider have graphs (that is, one-dimensional simplicial complexes) as their 1–skeleta. Therefore, we use the notation $G(X) := X^{(1)}$. As morphisms between cell complexes we always consider *cellular maps*, that is, maps sending k–skeleton into the k–skeleton. The *star* $\mathrm{St}(v)$ of a vertex v is the set of cells containing v.

For a graph G, we define its *triangle* (respectively, *square*) complex $X_{\triangle}$ (respectively, $X_{\square}$) as a two-dimensional cell complex with 1–skeleton G, and such that the two-cells are (solid) triangles (respectively, squares) whose boundaries are identified by isomorphisms with (graph) triangles (respectively, squares) in G. A *triangle-square complex* $X_{\triangle\square}(G)$ is defined analogously, as the union of $X_{\triangle}$ and $X_{\square}$ sharing common 1–skeleton G. A triangle-square (resp. triangle, square) complex is *flag* if it coincides with the triangle-square (resp. triangle, square) complex of its 1-skeleton. Observe that the triangle, square, and triangle-squares complexes are 2-dimensional complexes (all faces have dimension 0, 1, or 2).

In some cases, we associate complexes of higher dimension to a graph G. The *clique complex* of a graph G is the abstract simplicial complex $X(G)$ having the cliques (i.e., complete subgraphs) of G as simplices. A simplicial complex X is a *flag simplicial complex* if X is the clique complex of its 1–skeleton. The dimension of a clique complex $X(G)$ is the dimension of its largest simplex, i.e., the size of the largest clique of G minus 1. In a simplicial complex X, the *link* of a vertex $x \in X$ is the simplicial complex $\mathrm{Link}(x, X)$ with a $(d-1)$-simplex for each d-simplex containing x, with simplices attached according to the attachments of the corresponding simplices.

An *d-cube* is an isometric copy of $[-1,1]^d$, and has the product structure, so that each subcube of $[-1,1]^d$ is obtained by restricting some of the coordinates to $+1$ and some to -1. A *cube complex* is obtained from a collection of cubes of various dimensions by isometrically identifying certain subcubes. The *dimension* $\dim(X)$ of a cube complex X is the largest value of d for which X contains a d-cube. Observe that in general, in a square complex or in a triangle-square complex, we do not ask for the intersection of two cells to be a cell. However, in the case of cube complexes, one only consider cube complexes where the nonempty intersection of two cells is a common face (as it is always the case when considering simplicial complexes). In a cube complex X, the *link* of a vertex $x \in X$ is the simplicial complex $\mathrm{Link}(x,X)$ with a $(d-1)$-simplex for each d-cube containing x, with simplices attached according to the attachments of the corresponding cubes. For every graph G that does not contain infinite hypercubes as induced subgraphs, G gives rise to a cube complex $X_{\mathrm{cube}}(G)$. The cube complex $X_{\mathrm{cube}}(G)$ spanned by G has Q as a cube if and only if the 1-skeleton of Q is an induced subgraph of G which is a hypercube. A cube complex X is *flag* if it is the cube complex $X_{\mathrm{cube}}(G)$ of its 1-skeleton $G = X^{(1)}$.

A cell complex X is called *simply connected* if it is connected and if every continuous mapping of the 1-dimensional sphere S^1 into X can be extended to a continuous mapping of the disk D^2 with boundary S^1 into X. Let C be a cycle in the 1–skeleton of X. Then a cell complex D is called a *singular disk diagram* (or Van Kampen diagram) for C if the 1–skeleton of D is a plane graph whose inner faces are exactly the 2–cells of D and there exists a cellular map $\varphi : D \to X$ such that $\varphi|_{\partial D} = C$ (for more details see [**165**, Chapter V]). According to Van Kampen's lemma [**165**, pp. 150–151], a cell complex X is simply connected if and only if for every cycle C of X, one can construct a singular disk diagram. A singular disk diagram with no cut vertices (i.e., its 1–skeleton is 2–connected) is called a *disk diagram.* A *minimal (singular) disk* for C is a (singular) disk diagram D for C with a minimum number of 2–faces. This number is called the *(combinatorial) area* of C and is denoted $\mathrm{Area}(C)$. If X is a simply connected triangle-square complex, then for each cycle C all inner faces in a singular disk diagram D of C are triangles or squares.

As morphisms between cell complexes we consider all *cellular maps*, i.e., maps sending (linearly) cells to cells. An *isomorphism* is a bijective cellular map being a linear isomorphism (isometry) on each cell. A *covering (map)* of a cell complex X is a cellular surjection $p\colon \widetilde{X} \to X$ such that $p|_{\mathrm{St}(\widetilde{v},\widetilde{X})}\colon \mathrm{St}(\widetilde{v},\widetilde{X}) \to \mathrm{St}(p(\widetilde{v}),X)$ is bijective for every vertex $\widetilde{v}$ in $\widetilde{X}$; compare [**137**, Section 1.3]. The space $\widetilde{X}$ is then called a *covering space.* A *universal cover* of X is a simply connected covering space $\widetilde{X}$. It is unique up to isomorphism. In particular, if X is simply connected, then its universal cover is X itself. (Note that X is connected iff $G(X) = X^{(1)}$ is connected, and X is simply connected iff $X^{(2)}$ is so.)

An important class of cube complexes studied in geometric group theory and combinatorics is the class of CAT(0) cube complexes. In this case, being CAT(0) is equivalent to the unicity of geodesics in the ℓ_2 metric; see [**42**] for this and other properties of CAT(0) spaces. Gromov [**128**] gave a beautiful combinatorial characterization of CAT(0) cube complexes. A cube complex X satisfying is a *nonpositively curved (NPC)* complex if it satisfies the following cube condition:

Cube condition: if $k \geq 2$ and three k-cubes of X pairwise intersect in a $(k-1)$-cube and all three intersect in a $(k-2)$-cube, then they are included in a $(k+1)$-dimensional cube of X;

The following theorem is the characterization of CAT(0) cube complexes given by Gromov [**128**], that can be also taken as their definition:

THEOREM 1.4 ([**128**]). *A cube complex X endowed with the ℓ_2-metric is CAT(0) if and only if X is simply connected and nonpositively curved. If Y is a nonpositively curved cube complex, then the universal cover $\widetilde{Y}$ of Y is a CAT(0) cube complex.*

2. Weakly Modular Graphs

In this section, we present the general approach to obtain local-to-global characterizations in more details. We exemplify it by considering the case of weakly modular graphs and sketching the proof of Theorem 1.3. We start by recalling the definition of weakly modular graphs.

DEFINITION 1.5 (Weak modularity). [**16**, **80**] A graph G is *weakly modular with respect to a vertex u* if its distance function d satisfies the following triangle and quadrangle conditions (see Figure 1.1):

- *Triangle condition* TC(u): for any two vertices v, w with $1 = d(v,w) < d(u,v) = d(u,w)$ there exists a common neighbor x of v and w such that $d(u,x) = d(u,v) - 1$.
- *Quadrangle condition* QC(u): for any three vertices v, w, z with $d(v,z) = d(w,z) = 1$ and $2 = d(v,w) \leq d(u,v) = d(u,w) = d(u,z) - 1$, there exists a common neighbor x of v and w such that $d(u,x) = d(u,v) - 1$.

A graph G is called *weakly modular* if G is weakly modular with respect to any vertex u.

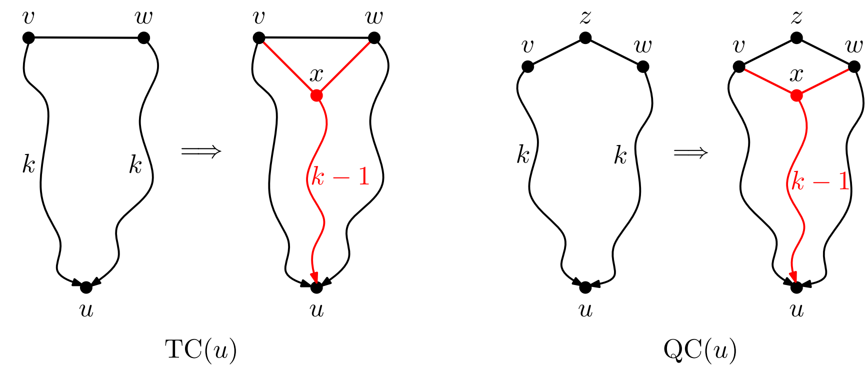


FIGURE 1.1. Triangle and quadrangle conditions

We say that a graph G is *locally weakly modular with respect to a vertex u* if it satisfies the two following conditions (See Figure 1.1):

- *Local triangle condition* LTC(u): for any two adjacent vertices v, w such that $d(u,v) = d(u,w) = 2$ there exists a common neighbor x of u, v and w.
- *Local quadrangle condition* LQC(u): for any three vertices v, w, z such that $z \sim v, w$ and $d(v,w) = d(u,v) = d(u,w) = d(u,z) - 1 = 2$, there exists a common neighbor x of u, v and w.

A graph G is *locally weakly modular* if G is locally weakly modular with respect to any vertex u.

The weak modularity of a graph G implies that any cycle of $X_{\triangle\square}(G)$ admits a disk diagram, and consequently that G is simply connected. Consequently, in order to prove Theorem 1.3, consider a locally weakly modular graph G and let $\widetilde{G}$ be the 1–skeleton of the universal cover $\widetilde{X} := \widetilde{X}_{\triangle\square}(G)$ of the triangle-square complex $X := X_{\triangle\square}(G)$ of G. To prove Theorem 1.3, we want to establish that $\widetilde{G}$ is weakly modular.

To do so, we will construct the universal cover $\widetilde{X}$ of X as an increasing union $\bigcup_{i\geq 1} \widetilde{X}_i$ of triangle-square complexes. The complexes $\widetilde{X}_i$ will be in fact spanned by concentric combinatorial balls $\widetilde{B}_i$ in $\widetilde{X}$. The covering map f is then the union $\bigcup_{i\geq 1} f_i$, where $f_i : \widetilde{X}_i \to X$ is a locally injective cellular map such that $f_i|_{\widetilde{X}_j} = f_j$, for every $j \leq i$. We denote by $\widetilde{G}_i = G(\widetilde{X}_i)$ the underlying graph of $\widetilde{X}_i$. We denote by $\widetilde{S}_i$ the set of vertices $\widetilde{B}_i \setminus \widetilde{B}_{i-1}$.

Pick any vertex v of X as the base-point. Define $\widetilde{B}_0 = \{\widetilde{v}\} := \{v\}, \widetilde{B}_1 := B_1(v, G)$. Let $\widetilde{X}_1$ be the triangle-square complex of $B_1(v, G)$, and let $f_1 \colon \widetilde{X}_1 \to X$ be the cellular map induced by $\mathrm{Id}_{B_1(v,G)}$. Assume that, for $i \geq 1$, we have constructed the vertex sets $\widetilde{B}_1, \dots, \widetilde{B}_i$, and we have defined the triangle-square complexes $\widetilde{X}_1 \subseteq \cdots \subseteq \widetilde{X}_i$ (for any $1 \leq j < k \leq i$ we have an identification map $\widetilde{X}_j \to \widetilde{X}_k$) and the corresponding cellular maps $f_1, \dots, f_i$ from $\widetilde{X}_1, \dots, \widetilde{X}_i$, respectively, to X so that the graph $\widetilde{G}_i = G(\widetilde{X}_i)$ and the complex $\widetilde{X}_i$ satisfy the following conditions:

- (P_i) $B_j(\widetilde{v}, \widetilde{G}_i) = \widetilde{B}_j$ for any $j \leq i$;
- (Q_i) $\widetilde{G}_i$ satisfies the triangle and quadrangle conditions with respect to $\widetilde{v}$;
- (R_i) for any $\widetilde{u} \in \widetilde{B}_{i-1}$, f_i defines a bijection between the star of $\widetilde{u}$ in $\widetilde{X}_i$ and the star of $u = f_i(\widetilde{u})$ in X;
- (S_i) for any $\widetilde{u} \in \widetilde{S}_i$, f_i defines an injection between the star of $\widetilde{u}$ in $\widetilde{X}_i$ and the star of $u = f_i(\widetilde{u})$ in X.

It can be easily checked that $\widetilde{B}_1, \widetilde{G}_1, \widetilde{X}_1$ and f_1 satisfy the conditions (P_1),(Q_1),(R_1) and (S_1). Now we construct the set $\widetilde{B}_{i+1}$, the graph $\widetilde{G}_{i+1}$ having $\widetilde{B}_{i+1}$ as the vertex-set, the triangle-square complex $\widetilde{X}_{i+1}$ having $\widetilde{G}_{i+1}$ as its 1-skeleton, and the map f_{i+1} : $\widetilde{X}_{i+1} \to X$. Let

$$Z = \{(\widetilde{w}, z) : \widetilde{w} \in \widetilde{S}_i \text{ and } z \in B_1(f_i(\widetilde{w}), G) \setminus f_i(B_1(\widetilde{w}, \widetilde{G}_i))\}.$$

If $(\widetilde{w}, z) \in Z$, $w = f_i(\widetilde{w})$ is a neighbor of z but $\widetilde{w}$ has no neighbor mapped to z by f_i. In order to ensure the local bijection between the star of $\widetilde{w}$ and the star of w, one has to add such a neighbor in the next level. However, in order to have bijections between stars, squares and triangles in the stars must be preserved. Therefore, on Z we define a binary relation $\equiv$ indicating which vertices of the next level have to be merged. We set $(\widetilde{w}, z) \equiv (\widetilde{w}', z')$ if and only if $z = z'$ and one of the following two conditions is satisfied:

- (Z1) $\widetilde{w}$ and $\widetilde{w}'$ are the same or adjacent in $\widetilde{G}_i$;
- (Z2) there exists $\widetilde{u} \in \widetilde{B}_{i-1}$ adjacent in $\widetilde{G}_i$ to $\widetilde{w}$ and $\widetilde{w}'$ and such that $f_i(\widetilde{u})f_i(\widetilde{w})zf_i(\widetilde{w}')$ is a square in G.

The crux of the proof is that the relation $\equiv$ is an equivalence relation on Z. Using Z and $\equiv$, we can define $\widetilde{G}_{i+1}$, $\widetilde{X}_{i+1}$ and f_{i+1}. Let $\widetilde{S}_{i+1}$ denote the set of equivalence classes of $\equiv$, i.e., $\widetilde{S}_{i+1} = Z/_{\equiv}$. For an ordered pair $(\widetilde{w}, z) \in Z$, we will denote by $[\widetilde{w}, z]$ the equivalence class of $\equiv$ containing $(\widetilde{w}, z)$. Set $\widetilde{B}_{i+1} := \widetilde{B}_i \cup \widetilde{S}_{i+1}$. Let $\widetilde{G}_{i+1}$ be the graph having $\widetilde{B}_{i+1}$ as the vertex set, in which two vertices $\widetilde{a}, \widetilde{b}$ are adjacent if and only if one of the following conditions holds:

(1) $\widetilde{a}, \widetilde{b} \in \widetilde{B}_i$ and $\widetilde{a}\widetilde{b}$ is an edge of $\widetilde{G}_i$,
(2) $\widetilde{a} \in \widetilde{B}_i$, $\widetilde{b} \in \widetilde{S}_{i+1}$ and $\widetilde{b} = [\widetilde{a}, z]$,
(3) $\widetilde{a}, \widetilde{b} \in \widetilde{S}_{i+1}$, $\widetilde{a} = [\widetilde{w}, z]$, $\widetilde{b} = [\widetilde{w}, z']$ for a vertex $\widetilde{w} \in \widetilde{B}_i$, and $z \sim z'$ in G.

Finally, we define the map $f_{i+1} : \widetilde{B}_{i+1} \to V(X)$ in the following way: if $\widetilde{a} \in \widetilde{B}_i$, then $f_{i+1}(\widetilde{a}) = f_i(\widetilde{a})$, otherwise, if $\widetilde{a} \in \widetilde{S}_{i+1}$ and $\widetilde{a} = [\widetilde{w}, z]$, then $f_{i+1}(\widetilde{a}) = z$. Notice that f_{i+1} is well-defined because all ordered pairs representing $\widetilde{a}$ have one and the same vertex z in the second argument.

One can prove that our inductive properties (P_{i+1}), (Q_{i+1}),(R_{i+1}), and (S_{i+1}) hold for $\widetilde{G}_{i+1}$ and f_{i+1}. In particular, we show that the image of a square (respectively, a triangle) of $\widetilde{G}_{i+1}$ is a square (respectively, a triangle) of G. This allows us to define the triangle-square complex $\widetilde{X}_{i+1}$.

Let $\widetilde{X}_v$ denote the triangle-square complex obtained as the directed union $\bigcup_{i\geq 0} \widetilde{X}_i$, with a vertex v of X as the base-point. Denote by $\widetilde{G}_v$ the 1–skeleton of $\widetilde{X}_v$. Since each $\widetilde{G}_i$ satisfies the triangle and quadrangle conditions with respect to $\widetilde{v}$, the graph $\widetilde{G}_v$ also satisfies the triangle and quadrangle conditions with respect to $\widetilde{v}$.

Therefore, $\widetilde{G}_v$ is weakly modular and the complex $\widetilde{X}_v$ is simply connected. Let $f = \bigcup_{i\geq 0} f_i$ be the map from $\widetilde{X}_v$ to X. Using property (R_i), one can show that $f : \widetilde{X}_v \to X$ is a covering map.

This finishes the proof, since $\widetilde{X}_v$ is a simply connected, and thus it is the universal covering space of X. It is then unique, i.e., not depending on v. Thus, $\widetilde{G}(= \widetilde{G}_v)$ is weakly modular with respect to every vertex v.

3. Basis Graphs of Matroids

A *matroid* on a finite set of elements E is a collection $\mathcal{B}$ of subsets of E, called *bases,* which satisfy the following exchange property: for all $A, B \in \mathcal{B}$ and $a \in A \setminus B$ there exists $b \in B \setminus A$ such that $A \setminus \{a\} \cup \{b\} \in \mathcal{B}$ (the base $A \setminus \{a\} \cup \{b\}$ is obtained from the base A by an *elementary exchange*). The *basis graph* $G = G(\mathcal{B})$ of a matroid $\mathcal{B}$ is the graph whose vertices are the bases of $\mathcal{B}$ and edges are the pairs A, B of bases differing by an elementary exchange, i.e., $|A\Delta B| = 2$.

Equivalently, a matroid on E can be defined as a simplicial complex X on E such that for any two simplices I_1, I_2 with $|I_1| < |I_2|$, there exists $e \in I_2 \setminus I_1$ such that $I_1 \cup \{e\}$ is a simplex of X. The simplices of X are called the *independent sets* of the matroids. The bases of the matroid are the maximal independent sets, i.e., the facets of X. With these definitions, it is easy to see that all bases of a matroid have the same size, and therefore X is a pure simplicial complex, i.e., all facets of X have the same dimension d (the rank of the matroid). By the definition of the basis graph, two basis are adjacents if and only if the two corresponding facets of X intersect in a face of dimension $d - 1$.

By the exchange property, basis graphs are connected. For any two bases A and B at distance 2 there exist at most four bases adjacent to A and B: if $A \setminus B = \{a_1, a_2\}$ and $B \setminus A = \{b_1, b_2\}$, then these bases have the form $A \setminus \{a_i\} \cup \{b_j\} = B \setminus \{b_j\} \cup \{a_i\}$ for $i, j \in \{1, 2\}$. On the other hand, the exchange property ensures that at least one of the pairs $A \setminus \{a_1\} \cup \{b_1\}, A \setminus \{a_2\} \cup \{b_2\}$ or $A \setminus \{a_1\} \cup \{b_2\}, A \setminus \{a_2\} \cup \{b_1\}$ must be bases. Together with A and B, this pair of bases C, C' induce a square in the basis graph. Therefore, A and B, together with their common neighbors induce a square, a pyramid, or an octahedron, i.e., basis graphs satisfy what we call the *interval condition*:

- *Interval condition* IC: for every pair of vertices u, v at distance 2, the interval $I(u, v)$ induces a square, a pyramid, or an octahedron.

The exchange property of bases also shows that if A, C, B, C' induce a square in the basis graph, then for any other base $D \in \mathcal{D}$, the equality $d(D, A) + d(D, B) = d(D, C) + d(D, C')$ holds (i.e., the total number of elementary exchanges to transform D to A and B equals to the total number of exchanges to transform D to C and C'). Following [**169**], we call this property of basis graphs the positioning condition.

A graph G satisfies the *positioning condition with respect to a vertex* v if the following holds:

- *Positioning condition* PC(v): for each square $u_1u_2u_3u_4$ of G, we have $d(v, u_1) + d(v, u_3) = d(v, u_2) + d(v, u_4)$.

A graph G satisfies the *positioning condition* (PC) if G satisfies PC(v) for every vertex v of G.

Finally, by Lemma 1.8 of [**169**], the subgraph induced by all bases adjacent to a given base is the line graph of a bipartite graph; we will call it the *link condition.*

In [**169**, Theorem 2.1] Maurer showed the basis graphs of matroids as connected graphs satisfying the three conditions above:

THEOREM 1.6 ([**169**, Theorems 2.1&3.1]). *A graph $G = (V, E)$ is the basis graph of a matroid if and only if G is a connected graph satisfying the interval condition, the positioning condition, and some vertex of G of finite degree satisfies the link condition.*

The fact that these three conditions are satisfied by any basis graph is quite straightforward. On the other hand, the proof that these three conditions are sufficient is quite involved. Maurer conjectured that the link condition is redundant. In [**60**], we showed that this is indeed the case:

THEOREM 1.7. *A graph $G = (V, E)$ is the basis graph of a matroid if and only if G is a connected graph satisfying the interval condition, the positioning condition, and has at least one vertex with finitely many neighbors.*

Maurer observed that the triangle-square complex of any basis graph is simply connected and he conjectured that this topological condition can replace the positioning condition. In the original formulation of Maurer, the conjecture is false [**109**]. However, by slightly strengthening the local conditions, our Theorem 1.2 answers positively to Maurer's conjecture and consequently provides a local-to-global characterization of basis graphs of matroids.

A graph G satisfies the *local positioning condition* (LPC) if for each square $u_1u_2u_3u_4$ and each vertex v such that $d(v, u_1) = d(v, u_3) = 2$, we have $d(v, u_2) + d(v, u_4) = 4$.

Theorem 1.2 is then a consequence of the following theorem:

THEOREM 1.8. *Let G be a connected graph satisfying the interval and the local positioning conditions, and having at least one vertex with finitely many neighbors. Then the 1–skeleton $\widetilde{G}$ of the universal cover $\widetilde{X}(G)$ of its triangle-square complex $X(G)$ is the basis graph of a matroid (and thus $\widetilde{X}(G)$ is finite).*

Observe that given a connected graph G satisfying the local conditions of Theorem 1.8, if the triangle-square complex is not simply connected, then its universal cover is finite. Not that this property is usually not true for the other classes of graphs we consider.

Basis graphs of matroids are not weakly modular but they enjoy metric conditions as well. The interval and positioning conditions imply the triangle condition, but the quadrangle condition is not always satisfied. However, basis graphs of matroids enjoy the weaker square-pyramid condition.

A graph G satisfies the *square-pyramid condition with respect to a vertex v* if the following holds:

- *Square-pyramid condition* SPC(v): for any three vertices u, w, w' of G with $u \sim w, w'$ and $2 = d(w, w') \leq d(v, u) = d(v, w') + 1 = d(v, w) + 1 = k + 1$, either there exists $x \sim w, w'$ such that $d(v, x) = k - 1$, or there exists $x \sim u, w, w'$ and $x' \sim u, w, w'$ such that $x \nsim x'$, and $d(x, v) = d(x', v) = k$.

A graph G satisfies the *square-pyramid condition* if G satisfies SPC(v) for every vertex v of G.

As in the case of weakly modular graphs, one direction in the proof of Theorem 1.8 is easy. To prove the converse, consider a connected graph G satisfying the interval and the local positioning conditions, and having at least one vertex with finitely many neighbors. In view of Maurer's theorem 1.6 and Theorem 1.7, it is enough to show that $\widetilde{G}$ satisfies the positioning condition.

To prove this result, we use a scheme similar to the one used in the proof of Theorem 1.3 presented in Section 2. The main differences between the two proofs are the following:

- The induction hypothesis (Q_i) that $\widetilde{G}_i$ satisfies TC(v) and QC(v) is replaced by the following induction hypothesis:
  (Q_i) $\widetilde{G}_i$ satisfies TC(v), SPC(v), and PC(v).
- In the definition of the equivalence relation $\equiv$ on the set Z, we add the following condition:
  (Z3) there exists a square in $\widetilde{S}_i$ containing $\widetilde{w}$ and $\widetilde{w}'$ such that its image under f_i together with z induces a pyramid in G.

The proof of the inductive steps follows the same ideas as the proof of Theorem 1.3. However, the proof is different since the local conditions at hand are different. In particular, establishing that $\equiv$ is an equivalence relation on Z is much more difficult in this case.

In [**60**], we established a characterization of the basis graphs of even Δ-matroid that is analogous to Theorem 1.8.

4. Helly Graphs

In this section, we consider Helly graphs and sketch the proof of Theorem 1.1. We start by recalling the definitions of Helly graphs.

Recall that a family of subsets $\mathcal{F}$ of a set X satisfies the *Helly property* if for any subfamily $\mathcal{F}'$ of $\mathcal{F}$, the intersection $\bigcap \mathcal{F}' = \bigcap\{F : F \in \mathcal{F}'\}$ is nonempty if and only if $F \cap F' \neq \emptyset$ for any pair $F, F' \in \mathcal{F}'$.

A graph G is a *Helly graph* if the family of balls of G satisfies the Helly property, i.e., every collection of pairwise intersecting balls of G has a nonempty intersection. A graph G is a 1–*Helly graph* if the family of unit balls (i.e., balls of radius 1) of G has the Helly property. A *clique-Helly graph* is a graph in which the collection of maximal cliques has the Helly property. Observe that a Helly graph is 1-Helly and that a 1-Helly graph is clique-Helly, but the converses do not hold. Indeed, any cycle of length at least 7 is 1-Helly but not Helly, cycles of lengths 4 to 6 are clique-Helly but not 1-Helly.

Helly graphs are the discrete analogues of hyperconvex spaces: namely, the requirement that radii of balls are from the nonnegative reals is modified by replacing the reals by the integers. In perfect analogy with hyperconvexity, there is a close relationship between Helly graphs and absolute retracts. A graph is an *absolute retract* exactly when it is a retract of any larger graph into which it embeds isometrically. Then absolute retracts and Helly graphs are the same [**23, 138**]. In particular, for any graph G there exists a smallest Helly graph comprising G as an isometric subgraph.

A vertex x of a graph G is *dominated* by another vertex y if the unit ball $B_1(x)$ is included in $B_1(x)$. A graph G is *dismantlable* if there exists a well order $\prec$ on its vertices such that every vertex v is dominated by a vertex $w \prec v$ in the subgraph of G induced by the vertices $u \preceq v$. Such a well-order $\prec$ is called a *dismantling* order.

The following theorem summarizes some known characterizations of finite Helly graphs:

THEOREM 1.9. *For a finite graph G, the following statements are equivalent:*

(i) G is a Helly graph;
*(ii) [**138**] G is a retract of a strong product of paths;*
*(iii) [**24**] G is a dismantlable clique-Helly graph;*
*(iv) [**23**] G is a weakly modular 1–Helly graph.*

We established the following equivalences in [**56**]. Observe that the equivalence $(i) \Leftrightarrow (iv)$ corresponds to Theorem 1.1.

THEOREM 1.10. *For a graph G, the following conditions are equivalent:*

(i) G is Helly;
(ii) G is 1–Helly and weakly modular;

(iii) G is clique-Helly and dismantlable;
(iv) G is clique-Helly with a simply connected triangle complex.

Moreover, if the clique complex $X(G)$ of G is finite-dimensional, then the conditions (i)–(iv) are equivalent to

(v) G is clique-Helly with a contractible clique complex.

Note that this theorem holds for arbitrary graphs (including infinite and not locally finite graphs). In order to establish these results in full generality, one has to use transfinite inductions. To prove Theorem 1.10, we establish the equivalence $(i) \Leftrightarrow (ii)$ and the sequence of implications $(ii) \Rightarrow (iii) \Rightarrow (iv) \Rightarrow (ii)$.

The proof of $(i) \Rightarrow (ii)$ is trivial. The proof of the reverse implication uses a different approach than in the finite case considered by Bandelt and Pesch [**23**]. We use in particular the notion of *Helly-critical* sets introduced by Polat [**204**]. The proof of $(ii) \Rightarrow (iii)$ relies on the following property of 1-Helly weakly modular graphs that is a generalisation of a result of Bandelt and Pesch [**23**] to arbitrary graphs.

PROPOSITION 1.11. *Let G be a 1-Helly weakly modular graph. Then for any two vertices u, v of G with $d(u, v) = k + 1 \geq 1$ there exists a vertex $y \in B_1(v) \cap B_k(u)$ that is adjacent to all vertices of $B_1(v) \cap B_k(u)$.*

This proposition implies that any Breadth-First-Search order is a dismantling order. Since dismantlability of G implies the simple connectivity of the triangle complex $X_{\triangle}(G)$ (and the contractibility of the clique complex $X(G)$ when G has no infinite cliques), we have $(iii) \Rightarrow (iv)$ and $(iii) \Rightarrow (v)$. Since $(v) \Rightarrow (iv)$ is trivial, it remains to show that $(iv) \Rightarrow (ii)$, i.e., to prove Theorem 1.1.

A first step towards the proof of Theorem 1.1 is the following proposition. We say that a graph G satisfies the (C_4, W_4)-*condition* if every square of G "lives" in a 4-wheel W_4, i.e., for every square $abcd$ of G, there exists a vertex $x \sim a, b, c, d$.

PROPOSITION 1.12. *A weakly modular graph G is 1–Helly if and only if G is clique-Helly and satisfies the (C_4, W_4)-condition.*

With the previous proposition at hand, the next proposition enables to conclude the proof of Theorem 1.1.

PROPOSITION 1.13. *Let G be a clique-Helly graph and let $\widetilde{G}$ be the 1–skeleton of the universal cover $\widetilde{X} := \widetilde{X}_{\triangle}(G)$ of the triangle complex $X := X_{\triangle}(G)$ of G. Then $\widetilde{G}$ is weakly modular, clique-Helly, and satisfies the (C_4, W_4)-condition.*

To prove this result, we use a scheme similar to the one used in the proof of Theorem 1.3 presented in Section 2. The main differences between the two proofs are the following:

- instead of considering the triangle-square $X_{\triangle\square}(G)$ complex $X_{\triangle\square}(G)$ of G, we consider the triangle complex $X_{\triangle}(G)$ of G
- We add the following induction hypothesis:
  (Q$'_i$) for every 4-cycle $\widetilde{w}_1, \widetilde{w}_2, \widetilde{w}_3, \widetilde{w}_4$ of $\widetilde{G}_i$ with $d(\widetilde{v}, \widetilde{w}_1) < d(\widetilde{v}, \widetilde{w}_2) = d(\widetilde{v}, \widetilde{w}_4) < d(\widetilde{v}, \widetilde{w}_3)$, there exists a vertex $\widetilde{u} \in \widetilde{G}_i$ such that $\widetilde{u} \sim \widetilde{w}_1, \widetilde{w}_2, \widetilde{w}_3, \widetilde{w}_4$. In other words, we ask for every "vertical" 4-cycle of $\widetilde{G}_i$ to live in a W_4.
- In the definition of the equivalence relation $\equiv$ on the set Z, we replace the condition (Z2) by the following condition:
  (Z2) there exist $\widetilde{u} \in \widetilde{B}_{i-1}$ and $\widetilde{u}' \in \widetilde{B}_i$ such that $\widetilde{u} \sim \widetilde{w}, \widetilde{w}'$, that $\widetilde{u}' \sim \widetilde{u}, \widetilde{w}, \widetilde{w}'$, and that $f_i(\widetilde{u}'), f_i(\widetilde{w}), f_i(\widetilde{u}), f_i(\widetilde{w}'), z$ induce a W_4 in G.

The proof of the inductive steps follows the same ideas as the proof of Theorem 1.3. However, the proof is different since the local conditions at hand are different (local weak-modularity vs. clique-Hellyness).

There are many examples of Helly graphs. For example, replacing each cube of a CAT(0) cube complex by a clique (this operation is called *thickening*) leads to a Helly graph. Therefore studying Helly groups (i.e. groups acting geometrically on Helly graphs/complexes) is more general than studying groups acting on CAT(0) cube complexes. Similarly, using the result of Lang [**156**], one can show that any hyperbolic group (i.e. a group acting geometrically on a Gromov-hyperbolic graph) is Helly. In a recent work [**55**], we studied Helly groups, providing other examples of Helly groups, as well as establishing their group-theoretical properties. In particular, we proved that Helly groups are biautomatic, generalizing similar results for hyperbolic groups [**48**,**116**] and CAT(0) cubical groups [**182**] (see also [**42**]). Recently, Huang and Osajda [**143**] established that Garside groups of finite type and FC-Artin groups are Helly. Their proof uses our local-to-global characterization of Helly graphs and using our result [**55**], it establishes that these groups are biautomatic.

5. Bucolic Graphs

In a graph G, a vertex m is a *median* of three vertices u, v, w if $m \in I(u,v) \cap I(v,w) \cap I(w,v)$, i.e., if m lies simultaneously on (u,v), (u,w) and (v,w)-shortest paths. A graph G is *median* if every triplet of vertices has a unique median, i.e., if $|I(u,v) \cap I(v,w) \cap I(w,v)| = 1$ for every triplet u, v, w of vertices.

Median graphs are bipartite and therefore they satisfy the triangle condition. In fact the median graphs are exactly the bipartite graphs satisfying the quadrangle condition without induced $K_{2,3}$.

Median graphs contain plenty of cubes and isometrically embed into hypercubes (in fact, they are exactly the retracts of hypercubes [**15**]).

As mentioned above, median graphs are exactly the 1-skeletons of CAT(0) cube complexes [**82**,**214**]. In fact, their square complexes can be characterized in the following way.

Theorem 1.14 ([**82**]). *A graph G is a median graph if and only if its square-complex $X_\square(G)$ is simply connected and satisfies the 3-cube condition.*

- *3-cube condition*: any three squares of $X(G)$, pairwise intersecting in an edge of G, and all three intersecting in a vertex of G, are included in the 2-skeleton of a 3-dimensional cube (see Figure 1.2);

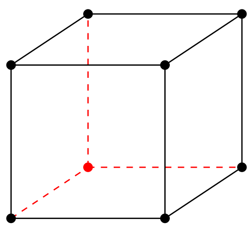


Figure 1.2. The 3-cube condition

This theorem shows that weak local and topological conditions can ensure strong global metric conditions. In fact, convex sets in median graphs are gated and satisfy the Helly property. Moreover, finite median graphs can be obtained by successive gated amalgams of their cubes [**145**, **244**].

An induced subgraph H of a graph G is *gated* [**111**] if for every vertex x outside H there exists a vertex x' in H (the *gate* of x) such that $x' \in I(x,y)$ for any y of H. Since the intersection of gated sets is gated, for any set S of vertices of a graph G, the *gated hull* $\langle\langle S\rangle\rangle$ is the smallest gated set of G containing S ($\langle\langle S\rangle\rangle$ is the intersection of all gated sets containing S). A graph G is a *gated amalgam* of two graphs G_1 and G_2

if G_1 and G_2 are (isomorphic to) two intersecting gated subgraphs of G whose union is all of G. Any gated subset S of a graph G gives rise to a partition F_a ($a \in S$) of the vertex-set of G; viz., the *fiber* F_a of a relative to S consists of all vertices x (including a itself) having a as their gate in S.

Rephrasing the previous result of [**145, 244**], median graphs are exactly the graphs that can be obtained by gated amalgams of Cartesian products of K_2. In [**15**], Bandelt showed that median graphs are exactly the retracts of hypercubes (i.e., Cartesian products of edges).

The structure theory of graphs based on Cartesian multiplication and gated amalgamation was further elaborated for more general classes of graphs. A graph with at least two vertices is said to be *prime* [**17, 78**] if it is neither a Cartesian product nor a gated amalgam of smaller graphs. A graph G is said to be *elementary* [**78**] if the only proper gated subgraphs of G are singletons. Observe that by the results of [**145, 244**], K_2 is the only prime (or elementary) median graph. Observe that a graph G is elementary if and only if the gated hull of any of its edges is the whole vertex set.

In a similar spirit, it was shown in [**22**] that quasi-median graphs (the weakly modular graphs not containing induced $K_{2,3}$ and $K_4 - e$ [**177**]) are exactly the graphs obtained by gated amalgams of Hamming graphs (Cartesian products of complete graphs) and that they are the retracts of Hamming graphs: the prime (or elementary) quasi-median graphs are the complete graphs. Bandelt and Chepoi [**17**] presented a similar decomposition scheme of weakly median graphs (the weakly modular graphs in which the vertex x in the triangle and quadrangle conditions is unique) in which the prime (or elementary) graphs are the hyperoctahedra and their subgraphs, the 5-wheel W_5, and the 2-connected plane bridged graphs. Generalizing the proof of the decomposition theorem of [**17**], Chastand [**78, 79**] presented a general framework of fiber-complemented graphs (graphs where fibers are gated) allowing to establish many general properties, previously proved only for particular classes of graphs. An important subclass of fiber-complemented graphs is the class of pre-median graphs [**78, 79**], i.e., weakly modular graphs without induced $K_{2,3}$ and W_4^-. Chastand showed that in the class of pre-median graphs, elementary and prime graphs coincide and asked for a characterization of these graphs (see [**78**, p. 121]). We provide such a characterization of prime (or elementary) pre-median graphs in Section6.

In this section, we investigate graphs that can be obtained by gated amalgams of Cartesian products of bridged (and weakly-bridged) graphs.

A graph is *bridged* if any isometric cycle has length 3. Alternatively, a graph G is bridged if and only if the balls $B_r(A, G)$ around convex sets A of G are convex [**117,231**]. Equivalently, a graph is bridged if and only if it is weakly modular and does not contain induced 4- and 5-cycles [**80**].

Similarly to bridged graphs, graphs with convex balls have been characterized in [**117, 231**] via conditions on isometric cycles. These graphs are not weakly modular in general (consider for example, the cycle of length 5 or the Petersen graph). A graph G is *weakly bridged* if G is a weakly modular graph and has convex balls. Alternatively, a graph G is weakly bridged if and only if G is weakly modular and does not contain induced C_4 [**89, 191**].

Analogously to the local-to-global characterization of median graphs, the clique complexes of bridged graphs have been characterized in the following way:

THEOREM 1.15 ([**82**]). *A graph G is bridged if and only if its clique complex $X(G)$ is simply connected and the links of vertices do not contain induced* 4- *and* 5-*cycles.*

The flag simplicial complexes satisfying the conditions of Theorem 1.15 have been rediscovered by Januszkiewicz and Swiatkowski [**146**], by Haglund [**130**], and by Wise [**256**] who called them "systolic complexes". Systolic complexes and groups turned

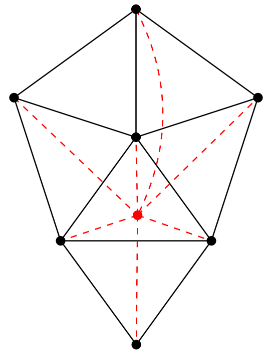


FIGURE 1.3. The $\widehat{W}_5$-wheel condition.

out to be good combinatorial analogs of CAT(0) (nonpositively curved) metric spaces and groups [**130**, **146**, **190**–**193**, **206**, **207**].

For an integer $k \geq 4$, a flag simplicial complex X is *locally k-large* if every cycle consisting of less than k edges in any of its links of simplices has some two consecutive edges contained in a 2-simplex of this link, i.e., the links do not contain induced cycles of length $< k$. A flag simplicial complex is *k-systolic* if it is locally k-large, connected and simply connected. A flag simplicial complex is *systolic* if it is 6-systolic [**130**, **146**, **256**].

Simplicial complexes corresponding to weakly bridged graphs have been introduced in [**89**, **191**] under the name of *weakly systolic* complexes. A flag simplicial complex X is *weakly systolic* [**89**, **191**] if connected and simply connected, locally 5-large, and satisfies the following local condition:

$\widehat{W}_5$-wheel condition: for each extended 5-wheel of X, there exists a vertex v adjacent to all vertices of this extended 5-wheel (see Figure 1.3).

THEOREM 1.16 ([**89**]). *A graph G is weakly bridged if and only if its clique complex $X(G)$ is weakly systolic.*

When considering cell complexes defined by weakly modular graphs where prime graphs are (weakly-)bridged graphs, one has to consider cells which are Cartesian products of cliques, i.e., prisms.

A *prism* is a convex polytope which is a Cartesian product of a finite number of finite-dimensional simplices. Faces of a prism are prisms of smaller dimensions. Particular instances of prisms are simplices and cubes (products of intervals). A *prism complex* is a cell complex X in which all cells are prisms so that the intersection of two prisms is empty or a common face of each of them. Cube complexes are prism complexes in which all cells are cubes and simplicial complexes are prism complexes in which all cells are simplices. The 1-skeleton of a prism of X is a *Hamming graph* without infinite cubes and cliques. Every graph G that does not contain infinite cliques or infinite hypercubes as induced subgraphs, G gives rise to a prism complex $X_{\text{prism}}(G)$. The prism complex $X_{\text{prism}}(G)$ spanned by G has P as a prism if and only if the 1-skeleton of P is an induced subgraph of G which is a Hamming graph. A prism complex X is *flag* if it is the prism complex $X_{\text{prism}}(G)$ of its 1-skeleton $G = X^{(1)}$.

We now define bucolic[1] graphs and bucolic complexes, a common generalization of bridged graphs/systolic simplicial complexes and median graphs/CAT(0) cube complexes.

DEFINITION 1.17. A graph G is *bucolic* if it is weakly modular, does not contain infinite cliques and does not contain induced subgraphs of the form $K_{2,3}$, W_4, and W_4^-. A bucolic graph is *strongly bucolic* if it does not contain induced W_5.

[1]The term bucolic is inspired by systolic, where b stands for bridged and c for cubical.

Definition 1.18 (Bucolic complexes). A prism complex X is *bucolic* if it is flag, connected and simply connected, and satisfies the following three local conditions:

Wheel condition: the 1-skeleton $X^{(1)}$ of X does not contain induced W_4 and satisfies the $\widehat{W}_5$-wheel condition;

Cube condition: if $k \geq 2$ and three k-cubes of X pairwise intersect in a $(k-1)$-cube and all three intersect in a $(k-2)$-cube, then they are included in a $(k+1)$-dimensional cube of X;

Prism condition: if a cube and a simplex of X intersect in a 1-simplex, then they are included in a prism of X.

A bucolic complex X is *strongly bucolic* if $G(X)$ does not contain induced W_5, i.e., a prism complex X is strongly bucolic if it is flag, connected, simply connected, and satisfies the cube and prism conditions, as well as the following local condition:

Strong-wheel condition: the 1-skeleton $X^{(1)}$ of X does not contain induced W_4 and W_5.

Gromov's characterization of CAT(0) cube complexes can be rephrased as follows: A cube complex is CAT(0) if and only if it is simply connected and satisfies the cube condition. Observe that if X is a flag cube complex, the 3-cube condition implies the cube condition.

Our main result on bucolic complexes is the following characterization via their 1- and 2-skeleta.

Theorem 1.19. *For a prism complex X, the following conditions are equivalent:*

(i) X is a (strongly) bucolic complex;

(ii) the 2-skeleton $X^{(2)}$ of X is a connected and simply connected triangle-square flag complex satisfying the (strong-)wheel, the 3-cube, and the 3-prism conditions;

(iii) the 1-skeleton $G(X) = X^{(1)}$ of X is a (strongly) bucolic graph not containing infinite hypercubes.

Moreover, if X is a connected flag prism complex satisfying the (strong-)wheel, the cube, and the prism conditions, then the universal cover $\widetilde{X}$ of X is (strongly) bucolic.

Observe that Condition (*ii*) provides a local-to-global characterization of bucolic complexes, while Condition (*iii*) provides a global metric characterization via their 1-skeleta (bucolic graphs). The next result shows that bucolic graphs are pre-median graphs in which all primes are weakly-bridged and that they are the retracts of the Cartesian products of their primes.

Theorem 1.20. *For a graph $G = (V, E)$ not containing infinite cliques, the following conditions are equivalent:*

(i) G is a bucolic (respectively, strongly bucolic) graph;

(ii) G is a retract of the (weak) Cartesian product of weakly bridged (respectively, bridged) graphs;

(iii) G is a pre-median graph in which all elementary (or prime) gated subgraphs are edges or 2-connected weakly bridged (respectively, bridged) graphs.

Moreover, if G is finite, then the conditions (i)–(iii) are equivalent to the following condition:

(iv) G can be obtained by successive applications of gated amalgamations from Cartesian products of 2-connected weakly bridged (respectively, bridged) graphs.

Theorem 1.20 allows us to show further non-positive-curvature-like properties of bucolic complexes. The following corollary (whose proof uses Whitehead's theorem) completes the analogy with the Cartan-Hadamard theorem.

Corollary 1.21. *Locally-finite bucolic complexes are contractible.*

Chronologically, Theorem 1.19 is the first local-to-global result we obtained. The proof of $(i) \Rightarrow (ii)$ is trivial. The proof of $(ii) \Rightarrow (iii)$ uses the same "universal cover" technique as in the proof of Theorem 1.3 for weakly modular graphs. The inductive properties (P_i)–(S_i) we establish as well as the definition of the equivalence relation $\equiv$ on Z (via Conditions (Z1) and (Z2)) are the same as in the proof of Theorem 1.3. However, the local properties at hand to establish them are weaker than in the case of general weakly modular graphs. Using weak modularity, one can establish the 3-cube and 3-prism conditions as well as the simple connectivity of the triangle-square complex, thus proving $(iii) \Rightarrow (ii)$. To prove that $(ii)\&(iii) \Rightarrow (i)$, one has to establish the Cube and Prism conditions for all dimensions k. The proof uses induction on k and is relatively technical.

The most difficult part of the proof of Theorem 1.20 are the implications $(i) \Rightarrow (iii)$ and $(iii) \Rightarrow (ii)$. Since bucolic graphs are pre-median, prime bucolic graphs are exactly elementary bucolic graphs [**78**, Lemma 4.8], to establish $(i) \Rightarrow (iii)$, it suffices to show that *all prime bucolic graphs are the 2-connected weakly bridged graphs or* K_2. Chastand [**79**, Theorem 3.2.1] proved that any fiber-complemented graph G whose primes are moorable is a retract of the Cartesian product of its primes. Therefore, to establish $(i) \Rightarrow (iii)$, we prove that *weakly bridged graphs are moorable.* The proof that $(ii) \Rightarrow (i)$ and $(iv) \Rightarrow (i)$ follows from the fact that weakly bridged graphs are bucolic and that the class of bucolic graphs is closed by taking products, gated amalgams, and retracts. The proof of $(i) \Rightarrow (iv)$ follows from the characterization of prime bucolic graphs and a result of Chastand [**78**, Theorem 5.4].

To show that prime bucolic graphs are the 2-connected weakly bridged graphs or K_2, we prove that the gated hull of each edge is a weakly bridged graph. The proof of this result was subsequently generalized to characterize all prime pre-median graphs (Theorem 1.24) and the outline of this proof will be presented in the next section.

A map $f : V(G) \to V(G)$ is a *mooring* of G onto a vertex u if $f(u) = u$, for every $v \neq u$, $f(v) \in I(v, u) \cap B_1(v)$, and for every edge vw, $f(v)$ and $f(w)$ coincide or are adjacent. A graph is *moorable* if for every vertex u of G, there exists a mooring of G onto u. Mooring can be viewed as a combing property of graphs — the notion coming from geometric group theory [**116**]. In [**81**], it was proved that the father map of any breadth-first-search (BFS) order of a locally-finite graph is a mooring (and a dismantling order). However, for weakly bridged graphs, BFS orders do not always provide moorings. Chepoi and Osajda [**89**] proved that locally-finite weakly bridged graphs are moorable (and dismantlable) using lexicographic-breadth-first-search (LexBFS) orders (instead of BFS orders). Polat [**203**] showed that all graphs admit a BFS order and, extending the result of [**81**], he showed that this BFS order provides a mooring (and a dismantling order) of non-locally-finite bridged graphs. Unfortunately, not every non-locally-finite graph admits a LexBFS order. In order to circumvent this problem, we refined Polat's definition of BFS and defined a well-ordering of the vertices of a graph, which is intermediate between BFS and LexBFS, that we called SimpLexBFS [**39**, Section 7]. We show that any (non-locally-finite) graph without infinite cliques admits a SimpLexBFS and that for weakly bridged graphs SimpLexBFS provides a mooring (and a dismantling order).

Further we established some CAT(0)-like property of groups acting on bucolic complexes [**39**].

THEOREM 1.22. *If X is a locally-finite bucolic complex and F is a finite group acting by cell automorphisms on X, then there exists a prism π of X which is invariant under the action of F. The center of the prism π is a point fixed by F.*

Replacing each prism by a regular Euclidean prism, each bucolic complex gives rise to a geometric prism complex. Since geometric systolic complexes in which cells are regular Euclidean simplices are not CAT(0) [**82, 146**], the geometric bucolic complexes are neither. However in [**40**], we showed that if the primes of a bucolic graph are chordal, then the associated geometric bucolic complex is CAT(0).

6. Prime Pre-median Graphs

Recall that a weakly modular graph is *pre-median* if it does not contain $K_{2,3}$ and W_4^- as an induced subgraph [**78**,**79**]. The following result is a corollary of Theorem 1.3.

THEOREM 1.23. *A graph G is pre-median if and only if G is locally weakly modular, does not contain induced $K_{2,3}$ and W_4^-, and its triangle-square complex is simply connected.*

Pre-median graphs are fiber-complemented and in the class of pre-median graphs, prime and elementary graphs coincide. Moreover, finite pre-median graphs (as finite fiber-complemented graphs) can be obtained as gated amalgams of Cartesian products of their primes [**78**, Theorem 5.4]. This leads Chastand [**78**, p. 121] to ask for a characterization of prime pre-median graphs.

By M_4 we denote the graph consisting of an induced 4–cycle (x_1, x_2, x_3, x_4) and four pairwise adjacent vertices a_1, a_2, a_3, a_4 such that $a_1 \sim x_1, x_2; a_2 \sim x_2, x_3; a_3 \sim x_3, x_4; a_4 \sim x_4, x_1$ and $a_1 \nsim x_3, x_4; a_2 \nsim x_1, x_4; a_3 \nsim x_1, x_2; a_4 \nsim x_2, x_3$ (see Figure 1.4, right).

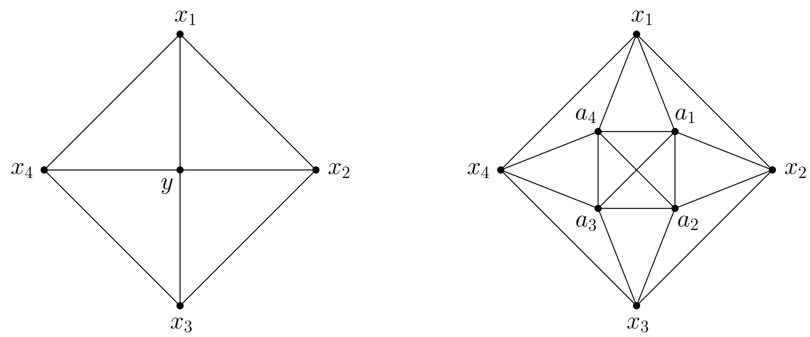


FIGURE 1.4. A W_4 (left) and a M_4 (right)

The following result answers Chastand's question:

THEOREM 1.24. *For a graph G, the following conditions are equivalent:*
(i) G is a prime pre-median graph;
(ii) G is a 2-connected pre-median graph and each square of G is included in an induced W_4 or M_4;
(iii) G is a 2-connected pre-median graph and its triangle complex $X_\triangle(G)$ (and hence its clique complex $X(G)$) is simply connected;
(iv) G is a 2-connected locally weakly modular graph not containing induced $K_{2,3}, W_4^-$, and its triangle complex $X_\triangle(G)$ is simply connected.

Note that any prime graph has to be 2-connected. Under this assumption, Condition (*ii*) shows that some local conditions characterize prime pre-median graphs in the class of 2-connected pre-median graphs. Observe that the triangle-square complex $X_{\triangle\square}(G)$ of any weakly modular graph is simply connected. Condition (*ii*) shows that in fact its triangle complex $X_\triangle(G)$ is simply connected. Condition (*iii*) shows that the simple connectivity of $X_\triangle(G)$ identifies the prime pre-median graphs. Condition (*iv*) is a typical local-to-global characterization of prime pre-median graphs.

In [**56**, Proposition 4.4], we show that some classical graphs are prime pre-median graphs: hyperoctahedra, half-cubes, Johnson graphs, the Schläfli graph, and the Gosset graph.

The equivalence $(iii) \Leftrightarrow (iv)$ follows from Theorem 1.3. The proof of $(ii) \Rightarrow (iii)$ follows from the fact that the triangle-square complex of a weakly modular graph is simply connected. To prove $(iii) \Rightarrow (i)$, i.e., that a 2–connected pre-median graph G with a simply connected triangle complex is prime, we use the fact that a pre-median graph is prime if and only if it is elementary [**78**, p. 121]. The implication $(iii) \Rightarrow (i)$ is then an immediate consequence of the following more general result (the proof uses minimal disk diagrams):

LEMMA 1.25. *If G is a 2–connected graph whose triangle complex is simply connected, then G is elementary.*

The most difficult part of the proof is to establish $(i) \Rightarrow (ii)$, i.e., to show that each square of a prime pre-median graph is included in a W_4 or M_4. Let H be an induced subgraph of a graph G. A 2-path $P = (a, v, b)$ (i.e., a path of length 2) of G is H–*fanned* [**39**] if $a, v, b \in V(H)$ and if there exists an (a, b)–path P' in H avoiding v and such that v is adjacent to all vertices of P', i.e., $v \sim P'$. Notice that P' can be chosen to be an induced path of G. A path $P = (x_0, x_1, \ldots, x_{k-1}, x_k)$ of G with $k > 2$ is H–*fanned* if every three consecutive vertices (x_i, x_{i+1}, x_{i+2}) of P form an H–fanned 2-path. When H is clear from the context (typically when $H = G$), we say that P is *fanned*.

The following lemma gives sufficient conditions for a square to be extended to a W_4 or M_4 and will be used to show that all squares admit such extensions.

LEMMA 1.26. *If $C = (v_1, v_2, v_3, v_4)$ is an induced 4–cycle of a pre-median graph G such that the 2–path (v_1, v_2, v_3) if fanned, then C is included in an induced W_4 or M_4. In particular, C is null-homotopic and all 2-paths of C are fanned.*

In order to apply the previous lemma to each 4-cycle, we prove that any 2-path is fanned.

LEMMA 1.27. *Let $a, b,$ and v be vertices of a pre-median graph G such that a and b can be connected by a fanned path avoiding v. If $v \sim a, b$, then there exists a fanned (a, b)–path P such that $v \sim P$; in particular, the 2-path (a, v, b) is fanned. If $v \sim a$ and $d(v, b) = 2$, then there exists a fanned (a, b)–path P such that $v \sim P \setminus \{b\}$.*

In weakly modular graphs, gatedness can be characterized locally [**80**]: A subgraph H of a weakly modular graph G is gated if and only if for any two distinct vertices u, v of H, any common neighbor of u, v in G belongs to H. Using this property, the gated hull $\langle\langle S \rangle\rangle$ of any set S inducing a connected subgraph of a finite weakly modular graph G can be constructed by the following procedure:

Algorithm 1.1: Gated-Hull(S)

$U \leftarrow S$;
while *there exists* $u, v \in U$ *and* $w \notin U$ *such that* $w \sim u, v$ **do**
  $U \leftarrow U \cup \{w\}$;
return (U);

We can extend the procedure GATED-HULL to arbitrary weakly modular graphs G in the following way. Let $\lhd$ be a well-order on $V(G)$ and let S be any subset of vertices inducing a connected subgraph of G. We define a subgraph K of G by (possibly transfinite) induction as follows. Set $H_0 := G(S)$. Given an ordinal α, assume that for every $\beta < \alpha$, we have defined H_β, and let $H_{<\alpha}$ be the subgraph induced by $\bigcup_{\beta<\alpha} V(H_\beta)$. Let

$$X = \{v \in V(G) \setminus V(H_{<\alpha}) : \text{ there exist } x, y \in V(H_{<\alpha}) \text{ such that } v \sim x, y\}.$$

If X is nonempty, then let v be the least element of $(X, \lhd)$ and define H_α to be the subgraph of G induced by $V(H_{<\alpha} \cup \{v\})$. If X is empty, then set $K := H_{<\alpha}$. In this case, one can show that K is the gated hull of S in G.

Let $T = a_0b_0c_0$ be a triangle in G and let H_0, H_1, H_2 be the subgraphs respectively induced by $\{a_0\}, \{a_0, b_0\}$ and $\{a_0, b_0, c_0\}$. Then for any ordinal α we define the subgraphs $H_{<\alpha}$ and H_α as in the transfinite version of the algorithm GATED-HULL. Let K be the gated hull of $\{a_0, b_0, c_0\}$ computed by the algorithm.

Using transfinite induction, we show that any 2-path of K is K-fanned.

LEMMA 1.28. *For any ordinal α, H_α is 2-connected and any 2-path of H_α is K–fanned. In particular, K is 2-connected and any 2-path of K is K–fanned.*

Combining Lemmas 1.26 and 1.28, we obtain that each induced 4-cycle C of K is included in an induced W_4 or M_4, establishing $(i) \Rightarrow (ii)$ and concluding the proof of Theorem 1.24.

7. Dual-Polar Graphs

As mentioned in Section 3, the basis graph of a matroid can be seen as the incidence graph of the facets of its independent set complex that is a pure simplicial complex. Other interesting classes of graphs arise in a similar way from incidence geometries.

A *point-line geometry* is a triple $\Pi = (P, L; R)$ of sets P, L and a relation $R \subseteq P \times L$ between P and L [**229**, **243**]. Elements of P are called *points*, and elements of L are called *lines.* If $(p, \ell) \in R$, then we say that the point p *lies* on the line ℓ or that the line ℓ *contains* the point p. If two points p, q lie on a common line ℓ, then we say that p and q are *collinear.* The *collinearity graph* $G := G(\Pi)$ of Π is the graph whose vertex set is the set P of points so that $p, q \in P$ define an edge if and only if p and q are collinear. A set $S \subseteq P$ of points is called a *subspace* of Π if for every line ℓ either $|\ell \cap S| \leq 1$ or $\ell \subseteq S$. The intersection of any collection of subspaces is a subspace, thus for any subset X of P there exists the smallest subspace containing X. A subspace S is called a *singular subspace* if any two points of S are collinear , i.e., the subgraph of $G(\Pi)$ induced by S is a clique. A point y of a subspace S (in particular, of a line ℓ) of a point-line geometry Π is a *nearest point* to a point x, if y is a closest to x point of S with respect to the graph-metric of G, i.e., $d_G(x, y) = \min\{d_G(x, y') : y' \in S\}$.

For a point-line geometry $\Pi = (P, L; R)$, consider the following conditions:

- (Q1) For a point p and a line ℓ not containing p, either exactly one point on ℓ is collinear with p, or all points on ℓ are collinear with p.
- (Q2) Every line contains at least three points.
- (Q3) For every point p there exists a point q such that p and q are not collinear.

A *polar space* is a point-line geometry $\Pi = (P, L; R)$ satisfying (Q1) and (Q2) (respectively, (Q2′)). In addition, if (Q3) is satisfied, then Π is said to be *nondegenerate.* The *rank* of a polar space Π is the length n of maximal chains of subspaces (ordered by inclusion). Similarly to matroids, polar spaces define pure simplicial complexes on the point-set: any maximal proper subspace U has rank $n-1$ (moreover, together with its subspaces, it is a projective space). Moreover, for a maximal subspace U and a point $p \in P \setminus U$ there exists a unique maximal subspace W such that W contains p and the rank of $U \cap W$ is $n-2$. When considering non-degenerate point-line geometries where subspaces are closed by intersection, these two conditions characterize the subspaces of polar spaces [**242**] (this was in fact the original definition of polar spaces given by Tits [**242**]). Polar spaces represent one of the fundamental types of incidence geometries.

A polar space $\Pi = (P, L; R)$ of rank n gives rise to another point-line geometry $\Pi^* = (P^*, L^*; R^*)$. The point set P^* is the set of all $(n-1)$–dimensional subspaces of Π, and the line set L^* is the set of all $(n-2)$–dimensional subspaces of Π, where the relation $R^* \subseteq P^* \times L^*$ is defined as $(W, U) \in R^*$ if $W \supseteq U$. A *dual polar space* is a point-line geometry Π^* obtained from some (generalized) polar space Π in this way. A *dual polar graph* G is the collinearity graph of a dual polar space Π^*. A dual polar graph can be completely recovered from the original polar space and its subspace poset. A characterization of dual polar graphs was given by Cameron [**47**]:

THEOREM 1.29. *[47] A graph G is the collinearity graph of a dual polar space Γ of rank n if and only if the following axioms are satisfied:*

- (A1) *for any point p and any line ℓ of Γ (i.e., maximal clique of G), there is a unique point of ℓ nearest to p in G;*
- (A2) *G has diameter n;*
- (A3&4) *the gated hull $\langle\langle u,v\rangle\rangle$ of two vertices u,v at distance 2 has diameter 2;*
- (A5) *for every pair of nonadjacent vertices u,v and every neighbor x of u in $I(u,v)$ there exists a neighbor y of v in $I(u,v)$ such that $d(u,v)=d(x,y)=d(u,y)+1=d(x,v)+1$.*

As noticed in [**18**], from this characterization immediately follows that dual polar graphs are weakly modular. We show that dual polar graphs can be characterized as a natural subclass of weakly modular graphs. Generalizing the interval condition for basis graphs of matroids, we say that a graph G is *thick* if every pair of vertices u,v at distance 2 in G are contained in an induced square of G. The graphs K_4^- and $K_{3,3}^-$ are represented in Figure 1.5.

THEOREM 1.30. *A graph $G=(V,E)$ is a dual polar graph if and only if G is a thick weakly modular graph not containing induced K_4^- and isometric $K_{3,3}^-$.*

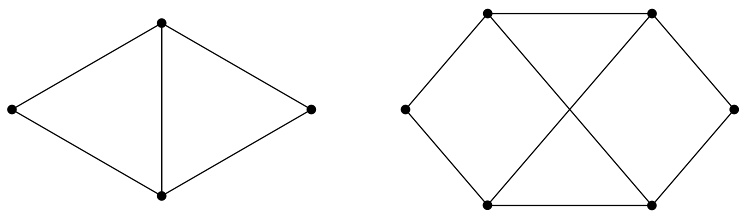


FIGURE 1.5. A K_4^- (left) and a $K_{3,3}^-$ (right)

We call a graph *locally dual polar* if it is thick, locally weakly modular, and does not contain induced K_4^- and isometric $K_{3,3}^-$.

THEOREM 1.31. *Let G be a locally dual polar graph. Then the 1–skeleton $\widetilde{G} := \widetilde{X}_{\triangle\square}(G)^{(1)}$ of the universal cover $\widetilde{X}_{\triangle\square}(G)$ of the triangle-square complex $X_{\triangle\square}(G)$ of G is a dual polar graph. If, moreover, G is locally finite, then $\widetilde{G}$ is a finite dual polar graph.*

The proof of the first assertion of this theorem is an immediate consequence of Theorem 1.3 and Theorem 1.30. The proof of the second assertion is quite technical and is in the spirit of the proof of Maurer's theorem 1.6 (even if the proofs are very different).

As an analogue of [**44**, Main Theorem (i)] we derive the following form of the local-to-global result.

THEOREM 1.32. *Every locally finite locally dual polar graph G is a quotient of a dual polar graph by a group action with the minimal displacement at least* 7.

We showed in [**56**] that the conditions of Theorems 1.30, 1.31 and 1.32 are implied by those of [**44**, Main Theorem (i)], i.e., Theorems 1.30, 1.31 can be viewed as a sharpening of Main Theorem (i) of [**44**].

In [**56**, Chapter 6], we also investigated weakly modular graph not containing induced K_4^- and isometric $K_{3,3}^-$ that we named *sweakly modular* graphs (or *swm*-graphs). Since these graphs are not thick, they are not dual polar graphs in general, but we showed that they can be build from cells that are dual-polar gated subgraphs and that they enjoy nice geometric and topological properties (for example, they are contractible). Similar constructions exist for complexes of oriented matroids (COMs) [**20**]. An interesting question is whether one can find analogous constructions for complexes where cells are basis polytopes of matroids.

8. Conclusion

Bridged graphs are the graphs where the balls around convex sets are convex [**117**, **231**]. When considering geodesic convexity in normed spaces (such as Euclidean spaces), balls around convex sets are convex if and only if balls are convex [**230**]. For geodesic convexity in general metric spaces, these two properties are no longer equivalent. In [**117**, **231**], characterizations of graphs that have convex balls have been given: they are exactly the graphs in which all isometric cycles have length 3 or 5 and for any two vertices u, v, any two neighbors x, y of u that lie on shortest paths from u to v are adjacent. While graphs with convex balls around convex sets (i.e., bridged or systolic graphs) have been thoroughly investigated, the structure and the properties of graphs with convex balls is less clear. It would be interesting to investigate their convexity and metric properties. In particular, one would like to provide a a local-to-global characterization of such graphs.

CONJECTURE 1.33. *A graph G has convex balls if and only if its triangle-pentagon complex is simply connected and the balls of radius ≤ 3 are convex.*

Bridged graphs are exactly the weakly modular graphs without W_4 and W_5 [**82**]. This characterization of bridged graphs was essential in establishing their local-to-global characterization [**82**]. Graphs with convex balls are not weakly modular: they do not satisfy the triangle condition. A first step would be to characterize them using metric properties in the spirit of weak modularity.

Weakly bridged graphs are the weakly modular graphs with convex balls. The triangle complex of any weakly-bridged graph is simply connected and one can ask whether the convexity of balls and the simple connectivity of the triangle complex are sufficient conditions to characterize weakly bridged graphs.

QUESTION 1.34. *Are weakly bridged graphs exactly the graphs with convex balls that have a simply connected triangle complex.*

When considering weakly modular graphs, one associate a 2-dimensional cell complex $X_{\triangle\square}(G)$ to each weakly modular graph G. When we consider Helly graphs, bridged graphs, median graphs, or bucolic graphs, one can consider cell complexes of higher dimensions (clique complexes for Helly and bridged graphs, cube complexes for median graphs, and prism complexes for bucolic graphs). These cell complexes have very interesting properties: for example, when we consider locally finite graphs, we obtain contractible cell complexes.

In [**56**], we associated complexes of higher dimension to L_1-embeddable weakly modular graphs (that form a subclass of pre-median graphs) and swm-graphs and study their property. A natural objective would be to find a natural way to associate a cell complex of higher dimension to any weakly modular graph in such a way that the obtained cell complexes inherit some nice properties. In [**56**], we proposed the definition of such a complex, but we were not able to establish any nice property satisfied by these complexes.

In a recent work [**55**], we studied groups acting geometrically on Helly graphs. In particular, we proved that Helly groups are biautomatic, generalizing similar results for hyperbolic groups [**48, 116**] and CAT(0) cubical groups [**182**]. In fact, we showed that hyperbolic groups and CAT(0) cubical groups are particular Helly groups. It would be interesting to see if we can identify other classes of groups acting on some subclasses of weakly modular graphs that are also Helly groups.

Systolic and bucolic groups (i.e. groups acting geometrically on bridged/systolic complexes and bucolic complexes respectively) are not Helly [**140**]. However, systolic groups are biautomatic. Since bucolic groups are a common generalization of systolic groups and CAT(0) cubical groups, one can wonder whether bucolic groups are biautomatic.

QUESTION 1.35. *Are bucolic groups biautomatic?*

We can also show that groups acting geometrically on weakly bridged graphs (i.e., groups acting geometrically on weakly systolic complexes) are biautomatic, and one can wonder whether it is still true when considering groups acting geometrically on graphs with convex balls.

CHAPTER 2

On Thiagarajan's Conjectures

Event structures, introduced by Nielsen, Plotkin, and Winskel [**183**, **253**, **254**], are a widely recognized abstract model of concurrent computation. An event structure (or more precisely, a prime event structure or an event structure with binary conflict) is a partially ordered set of the occurrences of actions, called events, together with a conflict relation. The partial order captures the causal dependency of events. The conflict relation models incompatibility of events so that two events that are in conflict cannot simultaneously occur in any state of the computation. Consequently, two events that are neither ordered nor in conflict may occur concurrently. More formally, an event structure is a triple $\mathcal{E} = (E, \leq, \#)$, consisting of a set E of events, and two binary relations $\leq$ and $\#$, the causal dependency $\leq$ and the conflict relation $\#$ with the requirement that the conflict is inherited by the partial order $\leq$. The pairs of events not in $\leq \cup \geq \cup \#$ define the concurrency relation $\|$. The domain of an event structure consists of all computation states, called configurations. Each computation state is a subset of events subject to the constraints that no two conflicting events can occur together in the same computation and if an event occurred in a computation then all events on which it causally depends have occurred too. Therefore, the domain of an event structure $\mathcal{E}$ is the set $\mathcal{D}(\mathcal{E})$ of all finite configurations ordered by inclusion. An event e is said to be enabled by a configuration c if $e \notin c$ and $c \cup \{e\}$ is a configuration. The degree of an event structure $\mathcal{E}$ is the maximum number of events enabled by a configuration of $\mathcal{E}$. The future (or the principal filter, or the residual) of a configuration c is the set of all finite configurations c' containing c.

Among other things, the importance of event structures stems from the fact that several fundamental models of concurrent computation lead to event structures. Nielsen, Plotkin, and Winskel [**183**] proved that every 1-safe Petri net N unfolds into an event structure $\mathcal{E}_N$. Later results of [**184**] and [**254**] show in fact that 1-safe Petri nets and event structures represent each other in a strong sense. In the same vein, Stark [**233**] established that the domains of configurations of trace automata are exactly the conflict event domains; a presentation of domains of event structures as trace monoids (Mazurkiewicz traces) or as asynchronous transition systems was given in [**215**] and [**26**], respectively. In both cases, the events of the resulting event structure are labeled (in the case of trace monoids and trace automata by the letters of a possibly infinite trace alphabet $M = (\Sigma, I)$) in a such a way that any two events enabled by the same configuration are labeled differently (such a labeling is usually called a nice labeling).

The Nice Labeling Conjecture. The *nice labeling conjecture* was formulated by Rozoy and Thiagarajan in [**215**] and asserts that

CONJECTURE 2.1 ([**215**]). *Every event structure with finite degree admits a nice labeling with a finite number of labels.*

A nice labeling is a labeling of events with the letters from some finite alphabet such that any two co-initial events (i.e., any two events which are concurrent or in minimal conflict) have different labels. The nice labelings of event structures arise when studying the equivalence of three different models of distributed computation: labeled event structures, net systems, and distributed monoids. The nice labeling conjecture can be viewed as a question about a local-to-global finite behavior of such models.

Assous, Bouchitté, Charretton, and Rozoy [**12**] proved that the event structures of degree 2 admit nice labelings with 2 labels and noticed that Dilworth's theorem implies that the conflict-free event structures of degree n have nice labelings with n labels. They also showed that finding the least number of labels in a nice labeling of a finite event structure is NP-hard. Santocanale [**221**] proved that all event structures of degree 3 with tree-like partial orders have nice labelings with 3 labels. Chepoi and Hagen [**88**] proved that the nice labeling conjecture holds for event structures with 2-dimensional domains, i.e., for event structures not containing three pairwise concurrent events.

The Conjecture on Regular Event Structures. To deal with *finite* 1-safe Petri nets, Thiagarajan [**239**,**240**] introduced the notions of regular event structure and trace-regular event structure. A regular event structure $\mathcal{E}$ is an event structure with a finite number of isomorphism types of futures of configurations and finite degree. A trace-regular event structure is an event structure $\mathcal{E}$ whose events can be nicely labeled by the letters of a finite trace alphabet $M = (\Sigma, I)$ in a such a way that the labels of any two concurrent events define a pair of I and there exists only a finite number of isomorphism types of labeled futures of configurations. These definitions were motivated by the fact that the event structures $\mathcal{E}_N$ arising from *finite* 1-safe Petri nets N are regular: Thiagarajan [**239**, **240**] proved that event structures of *finite* 1-safe Petri nets correspond to trace-regular event structures:

Theorem 2.2 ([**240**, Theorem 1]). *$\mathcal{E}$ is a trace-regular event structure if and only if there exists a finite 1-safe Petri net N such that $\mathcal{E}$ and $\mathcal{E}_N$ are isomorphic.*

This lead Thiagarajan to conjecture in [**239**, **240**] that

Conjecture 2.3 ([**239**, **240**]). *Regular event structures and trace-regular event structures are the same.*

Equivalently, this can be reformulated in the following way: *an event structure $\mathcal{E}$ is isomorphic to the event structure unfolding of a net system if and only if $\mathcal{E}$ is regular.*

Badouel, Darondeau, and Raoult [**14**] formulated two similar conjectures about conflict event domain that are recognizable by finite trace automata. The first one is equivalent to Conjecture 2.3, while the second one is formulated in a more general setting with an extra condition. In the particular case of event structures, their second conjecture can be reformulated as follows using a result of Schmitt [**223**]:

Conjecture 2.4 ([**14**]). *An event structure $\mathcal{E}$ is trace-regular if and only $\mathcal{E}$ is regular and has bounded $\natural$-cliques.*

Nielsen and Thiagarajan [**185**] established this conjecture for all regular conflict-free event structures and Badouel, Darondeau, and Raoult [**14**] proved it for context-free event domains, i.e., for domains whose underlying graph is a context-free graph sensu Müller and Schupp [**178**]. Morin [**176**] showed that any event structure admitting a regular nice labeling is trace-regular.

The Conjecture on the Decidability of the MSO Logic of Trace-Regular Event Structures. Thiagarajan and Yang [**241**] defined the monadic second order (MSO) theory $\mathrm{MSO}(\mathcal{E}_N)$ of an event structure unfolding $\mathcal{E}_N = (E, \leq, \#, \lambda)$ of a net system $N = (S, \Sigma, F, m_0)$ as the MSO theory of the relational structure $(E, (R_a)_{a\in\Sigma}, \leq)$. This immediately leads to the following fundamental question:

Question 2.5. *When* $\mathrm{MSO}(\mathcal{E}_N)$ *is decidable?*

It turns out that the MSO theory of trace event structures is not always decidable: [**241**] presented such an example suggested by I. Walukiewicz. To circumvent this example, Thiagarajan and Yang formulated the notion of a grid event structure and they showed that the MSO theory of event structures containing grids is undecidable. This leads Thiagarajan to conjecture in [**241**] that:

CONJECTURE 2.6 ([**241**]). *The* MSO *theory of a trace-regular event structure* $\mathcal{E}_N$ *is decidable if and only if* $\mathcal{E}_N$ *is grid-free.*

Notice also that preceding [**241**], Madhusudan [**166**] proved that the MSO theory of a trace event structure is decidable provided quantifications over sets are restricted to conflict-free subsets of events. In particular, this shows that the MSO theory of conflict-free trace-regular event structures is decidable.

With the event structure $\mathcal{E}_N$ one can associate two other MSO logics: the MSO logic $\mathrm{MSO}(\overrightarrow{G}(\mathcal{E}_N))$ of the directed graph $\overrightarrow{G}(\mathcal{E}_N)$ of the domain $\mathcal{D}(\mathcal{E}_N)$ of $\mathcal{E}_N$ and the MSO logic $\mathrm{MSO}(G(\mathcal{E}_N))$ of the undirected graph $G(\mathcal{E}_N)$ of the domain. This leads to the next question:

QUESTION 2.7. *When* $\mathrm{MSO}(\overrightarrow{G}(\mathcal{E}_N))$ *(respectively,* $\mathrm{MSO}(G(\mathcal{E}_N))$*) is decidable?*

Counterexamples. Unfortunately, it turned out that all previously formulated conjectures are false. The counterexamples are based on a more geometric and combinatorial view on event structures. We use the striking bijections between the domains of event structures, median graphs, and CAT(0) cube complexes. As mentioned in Chapter 1, it was proven in [**82, 214**] that 1-skeleta of CAT(0) cube complexes are exactly the median graphs. Barthélemy and Constantin [**25**] proved that the Hasse diagrams of domains of event structures are median graphs and every pointed median graph is the domain of an event structure. The bijection between pointed median graphs and event domains established in [**25**] can be viewed as the classical characterization of prime event domains as prime algebraic coherent partial orders provided by Nielsen, Plotkin, and Winskel [**183**]. More recently, this result was rediscovered in [**11**] in the language of CAT(0) cube complexes. In fact, the authors of [**11, 25**] were not aware of event structures: in [**25**], event structures are called *sites* and in [**11**], they are called *posets with inconsistent pairs*. Via these bijections, the events of an event structure $\mathcal{E}$ correspond to the parallelism classes of edges of the domain $D(\mathcal{E})$ viewed as a median graph.

A counterexample to the nice labeling conjecture was constructed by Chepoi in [**83**]. It is based on the bijections mentioned above and on the Burling's construction [**46**] of 3-dimensional box hypergraphs with clique number 2 and arbitrarily large chromatic numbers. The intersection graph of Burling was rediscovered by [**201**] that constructed triangle-free graphs that are intersection graph of segments in the plane with arbitrarily large chromatic numbers, disproving a famous Erdös conjecture and a more general conjecture of Scott [**224**]. Pawlik et al. also showed that Burling's graph is the intersection graph of frames in the plane. In [**70**], we study in details the restricted frame graphs of Pawlik et al. to extract more counterexamples to Scott's conjecture.

For CAT(0) cube complexes a question related to the nice labeling conjecture was independently formulated by V. Chepoi, F. Haglund, G. Niblo, M. Sageev: *is it true that the 1-skeleton of any CAT(0) cube complex of finite degree can be isometrically embedded into the Cartesian product of a finite number of trees?* A negative answer to this question was obtained in [**88**], based on a modification of the counterexample from [**83**]. However, in [**88**] it was shown that the answer is positive for 2-dimensional CAT(0) cube complexes. The nice labeling conjecture is also true in this case, i.e., for event structures with no three pairwise concurrent events [**88**]. Haglund [**131**] proved that this embedding question has a positive answer for hyperbolic CAT(0) cube complexes. Modifying the argument of [**131**], we can also show that the nice labeling conjecture is true for event structures with hyperbolic domains.

To deal with regular event structures, we show how to construct regular event domains from CAT(0) cube complexes. By a result of Gromov [**128**], CAT(0) cube complexes are exactly the universal covers of non-positively curved cube (NPC) complexes. Of particular importance for us are the CAT(0) cube complexes arising as universal covers of *finite* NPC complexes. We adapt the universal cover construction to directed

NPC complexes (Y, o) and show that every principal filter of the directed universal cover $(\widetilde{Y}, \widetilde{o})$ is the domain of an event structure. Furthermore, if the NPC complex Y is finite, then this event structure is regular. Motivated by this result, we call an event structure *strongly regular* if its domain is the principal filter of the directed universal cover $\widetilde{\mathbf{Y}} = (\widetilde{Y}, \widetilde{o})$ of a finite directed NPC complex $\mathbf{Y} = (Y, o)$. Using these techniques, we prove the following theorem:

THEOREM 2.8. *There exists a regular event structure with bounded ♮-cliques that are not trace-regular event structure. Consequently, Conjectures 2.3 and 2.4 are false.*

Our counterexample to Conjectures 2.3 and 2.4 is a strongly regular event structure derived from Wise's [**255**, **257**] nonpositively curved square complex $\mathbf{X}$ obtained from a tile set with six tiles. We also prove that other counterexamples to Thiagarajan's conjecture arise in a similar way from any aperiodic 4-deterministic tile set, such as the ones constructed by Kari and Papasoglu [**147**] and Lukkarila [**164**].

To address Conjecture 2.6, we use the same techniques as in the proof of Theorem 2.8. We construct an NPC square complex $\mathbf{Z}$ with one vertex, four edges, and three squares. We show that any principal filter of the universal cover of $\mathbf{Z}$ is the domain of a trace-regular event structure $\mathcal{E}_Z$. Then, we consider the event structure $\dot{\mathcal{E}}_Z$ obtained from $\mathcal{E}_Z$ by adding an event e_c for each configuration c of the domain in a such a way that e_c is in conflict with all events except those from c (those events precede e_c); $\dot{\mathcal{E}}_Z$ is called the *hairing* of $\mathcal{E}_Z$. The hairing $\dot{\mathcal{E}}_Z$ of $\mathcal{E}_Z$ is still trace-regular. We show that the graphs $G(\mathcal{E}_Z)$ and $G(\dot{\mathcal{E}}_Z)$ have infinite treewidth and bounded hyperbolicity. The first result implies that $\mathrm{MSO}(\dot{\mathcal{E}}_Z)$ is undecidable while the second result shows that $\dot{\mathcal{E}}_Z$ is grid-free, leading us to the following theorem.

THEOREM 2.9. *There exists a trace-regular event structure $\mathcal{E}_Z$ such that its hairing $\dot{\mathcal{E}}_Z$ is grid-free but the median graph $G(\mathcal{E}_Z)$ of $\mathcal{E}_Z$ has unbounded treewidth. Consequently, $\mathrm{MSO}(\dot{\mathcal{E}}_Z)$ is undecidable and thus Thiagarajan's Conjecture 2.6 is false.*

Positive Results. Even if the three conjectures turned out to be false, the work on them raised many important open questions and enables to establish a surprising bijection between 1-safe Petri nets (trace-regular event structures) and finite special cube complexes.

Haglund and Wise [**132**, **133**] detected pathologies which may occur in NPC complexes: self-intersecting hyperplanes, one-sided hyperplanes, directly self-osculating hyperplanes, and pairs of hyperplanes, which both intersect and osculate. They called the NPC complexes without such pathologies *special*. The main motivation for introducing and studying special cube complexes was the profound idea of Wise that the famous virtual Haken conjecture for hyperbolic 3-manifolds can be reduced to solving problems about special cube complexes. In a breakthrough result, Agol [**4**, **5**] completed this program and solved the virtual Haken conjecture using the deep theory of special cube complexes developed by Haglund and Wise [**132**, **133**]. The main ingredient in this proof is Agol's theorem that finite NPC complexes whose universal covers are hyperbolic are virtually special (i.e., they admit finite covers which are special cube complexes).

Refining the notion of strongly regular event structure, we call an event structure $\mathcal{E} = (E, \leq, \#)$ and its domain $\mathcal{D}(\mathcal{E})$ *cover-special* if $\mathcal{D}(\mathcal{E})$ is isomorphic to a principal filter of the universal cover of some finite (virtually) special complex.

As mentioned above, Thiagarajan [**239**] proved that event structures of *finite* 1-safe Petri nets correspond to trace-regular event structures.

We show that any cover-special event structure is trace-regular and is therefore the unfolding of a 1-safe Petri net:

THEOREM 2.10. *Any cover-special event structure $\mathcal{E}$ admits a trace-regular labeling, i.e., Thiagarajan's Conjecture 2.3 is true for cover-special event structures.*

Conversely, we show that to any finite 1-safe Petri net N one can associate a finite special cube complex $\mathbf{X}_N$ such that the domain of the event structure $\mathcal{E}_N$ (obtained as the unfolding of N) is a principal filter of the universal cover $\widetilde{\mathbf{X}}_N$ of $\mathbf{X}_N$.

THEOREM 2.11. *For any finite 1-safe Petri net N, there exists a finite directed special cube complex $\mathbf{X}_N$ such that the event structure unfolding of N is isomorphic to a cover-special event structure obtained from $\mathbf{X}_N$.*

This establishes a correspondence between 1-safe Petri nets and finite special cube complexes and provides a combinatorial characterization of trace-regular event structures.

Using Agol's breakthrough result [**4, 5**], another interesting consequence of Theorem 2.10 is that Thiagarajan's conjecture is true for strongly regular event structure arising from finite non-positively curved complexes that have a hyperbolic universal cover.

Concerning the decidability of the MSO theories of trace-regular event structures, we show that the decidability of each of $\mathrm{MSO}(G(\mathcal{E}_N)))$ and $\mathrm{MSO}(\overrightarrow{G}(\mathcal{E}_N))$ is equivalent to the fact that $G(\mathcal{E}_N)$ has finite treewidth and to the fact that $\overrightarrow{G}(\mathcal{E}_N)$ is a context-free graph. This completely answers Question 2.7.

THEOREM 2.12. *For a trace-regular event structure $\mathcal{E} = (E, \leq, \#, \lambda)$, the following conditions are equivalent:*

(1) $\mathrm{MSO}(\overrightarrow{G}(\mathcal{E}))$ *is decidable;*
(2) $\mathrm{MSO}_1(G(\mathcal{E}))$ *is decidable;*
(3) $\mathrm{MSO}_2(G(\mathcal{E}))$ *is decidable:*
(4) $G(\mathcal{E})$ *has finite treewidth;*
(5) the clusters of $G(\mathcal{E})$ *have bounded diameter;*
(6) $\overrightarrow{G}(\mathcal{E})$ *is context-free.*

We also prove that if $\mathrm{MSO}(\overrightarrow{G}(\mathcal{E}_N))$ is decidable, then $\mathrm{MSO}(\mathcal{E}_N)$ is decidable (the converse is not true). We prove that $\mathrm{MSO}(\dot{\mathcal{E}}_N)$ is decidable if and only if $\mathrm{MSO}(\overrightarrow{G}(\mathcal{E}_N))$ is decidable, i.e., if and only if $G(\mathcal{E})$ has finite treewidth. All these results provide partial answers to Question 2.5.

THEOREM 2.13. *For a trace-regular event structure $\mathcal{E} = (E, \leq, \#, \lambda)$,* $\mathrm{MSO}(\dot{\mathcal{E}})$ *is decidable if and only if* $\mathrm{MSO}(\overrightarrow{G}(\mathcal{E}))$ *is decidable. In particular,* $\mathrm{MSO}(\dot{\mathcal{E}})$ *is decidable if and only if $G(\mathcal{E})$ has finite treewidth.*

Since $\mathrm{MSO}(\mathcal{E})$ is a fragment of $\mathrm{MSO}(\dot{\mathcal{E}})$, we obtain the following corollary of Theorem 2.13:

COROLLARY 2.14. *For any trace-regular event structure $\mathcal{E} = (E, \leq, \#, \lambda)$, if $G(\mathcal{E})$ has finite treewidth, then* $\mathrm{MSO}(\mathcal{E})$ *is decidable.*

The results of this chapter are based on the papers [**50, 51**] and [**52**].

1. Event Structures and Net Systems

1.1. Event Structures and their Domains. An *event structure* is a triple $\mathcal{E} = (E, \leq, \#)$, where

- E is a set of *events*,
- $\leq \subseteq E \times E$ is a partial order of *causal dependency*,
- $\# \subseteq E \times E$ is a binary, irreflexive, symmetric relation of *conflict*,
- $\downarrow e := \{e' \in E : e' \leq e\}$ is finite for any $e \in E$,
- $e \# e'$ and $e' \leq e''$ imply $e \# e''$.

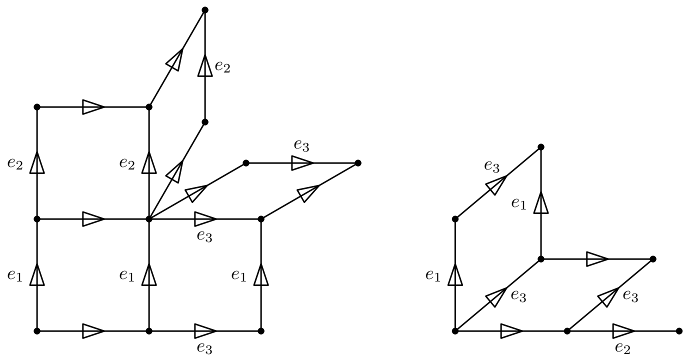


FIGURE 2.1. Two examples where $e_1 \natural_{(3)} e_2$: $e_1 \| e_3$ and $e_2 \#_\mu e_3$

Two events e', e'' are *concurrent* (notation $e' \| e''$) if they are order-incomparable and they are not in conflict. The conflict $e' \# e''$ between two elements e' and e'' is said to be *minimal* (notation $e' \#_\mu e''$) if there is no event $e \neq e', e''$ such that either $e \leq e'$ and $e \# e''$ or $e \leq e''$ and $e \# e'$. We say that e is an *immediate predecessor* of e' (notation $e \lessdot e'$) if and only if $e \leq e', e \neq e'$, and for every e'' if $e \leq e'' \leq e'$, then $e'' = e$ or $e'' = e'$.

Given two event structures $\mathcal{E}_1 = (E_1, \leq_1, \#_1)$ and $\mathcal{E}_2 = (E_2, \leq_2, \#_2)$, a map $f : E_1 \to E_2$ is an isomorphism if f is a bijection such that $e \leq_1 e'$ iff $f(e) \leq_2 f(e')$ and $e \#_1 e'$ iff $f(e) \#_2 f(e')$ for every $e, e' \in E_1$. If such an isomorphism exists, then $\mathcal{E}_1$ and $\mathcal{E}_2$ are said to be isomorphic; notation $\mathcal{E}_1 \equiv \mathcal{E}_2$.

A *configuration* of an event structure $\mathcal{E} = (E, \leq, \#)$ is any finite subset $c \subset E$ of events which is *conflict-free* ($e, e' \in c$ implies that e, e' are not in conflict) and *downward-closed* ($e \in c$ and $e' \leq e$ implies that $e' \in c$) [**254**]. Notice that $\varnothing$ is always a configuration and that $\downarrow e$ and $\downarrow e \setminus \{e\}$ are configurations for any $e \in E$. The *domain* of an event structure is the set $\mathcal{D} := \mathcal{D}(\mathcal{E})$ of all configurations of $\mathcal{E}$ ordered by inclusion; (c', c) is a (directed) edge of the Hasse diagram of the poset $(\mathcal{D}(\mathcal{E}), \subseteq)$ if and only if $c = c' \cup \{e\}$ for an event $e \in E \backslash c$. An event e is said to be *enabled* by a configuration c if $e \notin c$ and $c \cup \{e\}$ is a configuration. Denote by $en(c)$ the set of all events enabled at the configuration c. Two events are called *co-initial* if they are both enabled at some configuration c. Note that if e and e' are co-initial, then either $e \#_\mu e'$ or $e \| e'$. It is easy to see that two events e and e' are in minimal conflict $e \#_\mu e'$ if and only if $e \# e'$ and e and e' are co-initial. The *degree* $\deg(\mathcal{E})$ of an event structure $\mathcal{E}$ is the least positive integer d such that $|en(c)| \leq d$ for any configuration c of $\mathcal{E}$. We say that $\mathcal{E}$ has *finite degree* if $\deg(\mathcal{E})$ is finite. The *future* (or the *principal filter*) $\mathcal{F}(c)$ of a configuration c is the set of all configurations c' containing c: $\mathcal{F}(c) = \uparrow c := \{c' \in \mathcal{D}(\mathcal{E}) : c \subseteq c'\}$, i.e., $\mathcal{F}(c)$ is the principal filter of c in the ordered set $(\mathcal{D}(\mathcal{E}), \subseteq)$.

For an event structure $\mathcal{E} = (E, \leq, \#)$, let $\natural$ be the least irreflexive and symmetric relation on the set of events E such that $e_1 \natural e_2$ if one of the following holds:

(1) $e_1 \| e_2$, or
(2) $e_1 \#_\mu e_2$, or
(3) there exists an event e_3 that is co-initial with e_1 and e_2 at two different configurations such that $e_1 \| e_3$ and $e_2 \#_\mu e_3$ (see Figure 2.1 for examples).

If $e_1 \natural e_2$ and this comes from condition (3), then we write $e_1 \natural_{(3)} e_2$. A $\natural$-*clique* is any complete subgraph of the graph whose vertices are the events and whose edges are the pairs of events $e_1 e_2$ such that $e_1 \natural e_2$.

A *labeled event structure* $\mathcal{E}^\lambda = (\mathcal{E}, \lambda)$ is defined by an *underlying event structure* $\mathcal{E} = (E, \leq, \#)$ and a *labeling* λ that is a map from E to some alphabet Σ. Two labeled event structures $\mathcal{E}_1^{\lambda_1} = (\mathcal{E}_1, \lambda_1)$ and $\mathcal{E}_2^{\lambda_1} = (\mathcal{E}_2, \lambda_2)$ are isomorphic (notation $\mathcal{E}_1^{\lambda_1} \equiv \mathcal{E}_2^{\lambda_2}$)

if there exists an isomorphism f between the underlying event structures $\mathcal{E}_1$ and $\mathcal{E}_2$ such that $\lambda_2(f(e_1)) = \lambda_1(e_1)$ for every $e_1 \in E_1$.

A labeling $\lambda : E \to \Sigma$ of an event structure $\mathcal{E}$ defines naturally a labeling of the directed edges of the Hasse diagram of its domain $\mathcal{D}(\mathcal{E})$ that we also denote by λ. A labeling $\lambda : E \to \Sigma$ of an event structure $\mathcal{E}$ is called a *nice labeling* if any two events that are co-initial have different labels [**215**]. A nice labeling of $\mathcal{E}$ can be reformulated as a labeling of the directed edges of the Hasse diagram of its domain $\mathcal{D}(\mathcal{E})$) subject to the following local conditions:

Determinism: the edges outgoing from the same vertex of $\mathcal{D}(\mathcal{E})$ have different labels;
Concurrency: the opposite edges of each square of $\mathcal{D}(\mathcal{E})$ are labeled with the same labels.

In the following, we use interchangeably the labeling of an event structure and the labeling of the edges of its domain.

1.2. Mazurkiewicz Traces. A *(Mazurkiewicz) trace alphabet* is a pair $M = (\Sigma, I)$, where Σ is a finite non-empty alphabet set and $I \subset \Sigma \times \Sigma$ is an irreflexive and symmetric relation called the *independence relation.* The relation $D := (\Sigma \times \Sigma) \setminus I$ is called the *dependence relation.* As usual, Σ^* is the set of finite words with letters in Σ. For $\sigma \in \Sigma^*$, $last(\sigma)$ denotes the last letter of σ. The independence relation I induces the equivalence relation $\sim_I$, which is the reflexive and transitive closure of the binary relation $\leftrightarrow_I$: *if* $\sigma, \sigma' \in \Sigma^*$ *and* $(a, b) \in I$, *then* $\sigma ab\sigma' \leftrightarrow_I \sigma ba\sigma'$. The $\sim_I$-equivalence class containing $\sigma \in \Sigma^*$ is called a *(Mazurkiewicz) trace* and will be denoted by $\langle\sigma\rangle$. The trace $\langle\sigma\rangle$ is *prime* if σ is non-null and for every $\sigma' \in \langle\sigma\rangle$, $last(\sigma) = last(\sigma')$. The partial ordering relation $\sqsubseteq$ between traces is defined by $\langle\sigma\rangle \sqsubseteq \langle\tau\rangle$ (and $\langle\sigma\rangle$ is said to be a *prefix* of $\langle\tau\rangle$) if there exist $\sigma' \in \langle\sigma\rangle$ and $\tau' \in \langle\tau\rangle$ such that σ' is a prefix of τ'.

1.3. Trace-Regular Event Structures. In this subsection, we recall the definitions of regular event structures, trace-regular event structures, and regular nice labelings of event structures. We closely follow the definitions and notations of [**185**, **239**, **240**]. Let $\mathcal{E} = (E, \leq, \#)$ be an event structure. Let c be a configuration of $\mathcal{E}$. Set $\#(c) = \{e' : \exists e \in c, e\#e'\}$. The *event structure rooted at c* is defined to be the triple $\mathcal{E}\backslash c = (E', \leq', \#')$, where $E' = E \setminus (c \cup \#(c))$, $\leq'$ is $\leq$ restricted to $E' \times E'$, and $\#'$ is $\#$ restricted to $E' \times E'$. It can be easily seen that the domain $\mathcal{D}(\mathcal{E}\backslash c)$ of the event structure $\mathcal{E}\backslash c$ is isomorphic to the principal filter $\mathcal{F}(c)$ of c in $\mathcal{D}(\mathcal{E})$ such that any configuration c' of $\mathcal{D}(\mathcal{E})$ corresponds to the configuration $c' \setminus c$ of $\mathcal{D}(\mathcal{E}\backslash c)$.

For an event structure $\mathcal{E} = (E, \leq, \#)$, define the equivalence relation $R_\mathcal{E}$ on its configurations in the following way: for two configurations c and c' set $cR_\mathcal{E}c'$ if and only if $\mathcal{E}\backslash c \equiv \mathcal{E}\backslash c'$. The *index* of an event structure $\mathcal{E}$ is the number of equivalence classes of $R_\mathcal{E}$, i.e., the number of isomorphism types of futures of configurations of $\mathcal{E}$. The event structure $\mathcal{E}$ is *regular* [**185**, **239**, **240**] if $\mathcal{E}$ has finite index and finite degree.

Now, let $\mathcal{E}^\lambda = (\mathcal{E}, \lambda)$ be a labeled event structure. For any configuration c of $\mathcal{E}$, if we restrict λ to $\mathcal{E}\backslash c$, then we obtain a labeled event structure $(\mathcal{E}\backslash c, \lambda)$ denoted by $\mathcal{E}^\lambda\backslash c$. Analogously, define the equivalence relation $R_{\mathcal{E}^\lambda}$ on its configurations by setting $cR_{\mathcal{E}^\lambda}c'$ if and only if $\mathcal{E}^\lambda\backslash c \equiv \mathcal{E}^\lambda\backslash c'$. The index of $\mathcal{E}^\lambda$ is the number of equivalence classes of $R_{\mathcal{E}^\lambda}$. We say that an event structure $\mathcal{E}$ admits a *regular nice labeling* if there exists a nice labeling λ of $\mathcal{E}$ with a finite alphabet Σ such that $\mathcal{E}^\lambda$ has finite index.

We continue by recalling the definition of trace-regular event structures from [**239**, **240**]. For a trace alphabet $M = (\Sigma, I)$, an *M-labeled event structure* is a labeled event structure $\mathcal{E}^\phi = (\mathcal{E}, \lambda)$, where $\mathcal{E} = (E, \leq, \#)$ is an event structure and $\lambda : E \to \Sigma$ is a labeling function which satisfies the following conditions:

(LES1) $e\#_\mu e'$ implies $\lambda(e) \neq \lambda(e')$,
(LES2) if $e \lessdot e'$ or $e\#_\mu e'$, then $(\lambda(e), \lambda(e')) \in D$,
(LES3) if $(\lambda(e), \lambda(e')) \in D$, then $e \leq e'$ or $e' \leq e$ or $e\#e'$.

We call λ a *trace labeling* of $\mathcal{E}$ with the trace alphabet $M = (\Sigma, I)$. The conditions (LES2) and (LES3) on the labeling function ensures that the concurrency relation $\|$ of $\mathcal{E}$ respects the independence relation I of M. In particular, since I is irreflexive, from (LES3) it follows that any two concurrent events are labeled differently. Since by (LES1) two events in minimal conflict are also labeled differently, this implies that λ is a finite nice labeling of $\mathcal{E}$.

An M-labeled event structure $\mathcal{E}^\lambda = (\mathcal{E}, \lambda)$ is *regular* if $\mathcal{E}^\lambda$ has finite index. Finally, an event structure $\mathcal{E}$ is called a *trace-regular event structure* [**239**, **240**] if there exists a trace alphabet $M = (\Sigma, I)$ and a regular M-labeled event structure $\mathcal{E}^\lambda$ such that $\mathcal{E}$ is isomorphic to the underlying event structure of $\mathcal{E}^\lambda$. From the definition immediately follows that every trace-regular event structure is also a regular event structure.

1.4. Net Systems and their Event Structure Unfoldings. In the following presentation of finite 1-safe Petri nets and their unfoldings to event structures, we closely follow the paper by Thiagarajan and Yang [**241**]. A *net system* (or, equivalently, a *finite 1-safe Petri net*) is a quadruplet $N = (S, \Sigma, F, m_0)$ where S and Σ are disjoint finite sets of *places* and *transitions* (called also *actions* or *events*), $F \subseteq (S \times \Sigma) \cup (\Sigma \times S)$ is the *flow relation*, and $m_0 \subseteq S$ is the *initial marking*. For $v \in S \cup \Sigma$, set ${}^\bullet v = \{u : (u, v) \in F\}$ and $v^\bullet = \{u : (v, u) \in F\}$. A *marking* of N is a subset of S. The *transition relation* (or the *firing rule*) $\longrightarrow \subseteq 2^S \times \Sigma \times 2^S$ is defined by $m \xrightarrow{a} m'$ if ${}^\bullet a \subseteq m$, $(a^\bullet - {}^\bullet a) \cap m = \varnothing$, and $m' = (m - {}^\bullet a) \cup a^\bullet$. The transition relation $\longrightarrow$ is extended to sequences of transitions as follows (this new relation is also denoted by $\longrightarrow$): (1) $m \xrightarrow{\varepsilon} m$ for any marking m and (2) if $m \xrightarrow{\sigma} m'$ for $\sigma \in \Sigma^*$ and $m' \xrightarrow{a} m''$ for $a \in \Sigma$, then $m \xrightarrow{\sigma a} m''$. $\sigma \in \Sigma^*$ is called a *firing sequence* at m if there exists a marking m' such that $m \xrightarrow{\sigma} m'$. Denote by FS the set of firing sequences at m_0. A marking m is *reachable* if there exists a firing sequence σ such that $m_0 \xrightarrow{\sigma} m$.

Given a net system $N = (S, \Sigma, F, m_0)$, there is a canonical way to associate a Σ-labeled event structure $\mathcal{E}_N$ with N. The trace alphabet associated with N is the pair (Σ, I), where $(a, b) \in I$ iff $(a^\bullet \cup {}^\bullet a) \cap (b^\bullet \cup {}^\bullet b) = \varnothing$. Observe that the trace alphabet (Σ, I) is independent of the initial marking of N. Given the trace alphabet (Σ, I), we call the traces of the form $\langle\sigma\rangle, \sigma \in FS$ *firing traces* of N. Denote by $\mathcal{FT}(N)$ the set of all firing traces of N and by $\mathcal{PFT}(N)$ the subset of $\mathcal{FT}(N)$ consisting of prime firing traces.

Example 2.15. In Figure 2.2, we present a net system N^* with 12 transitions $h_1, h_1', h_2, h_2', h_3, h_3', h_4, h_4', v_1, v_1', v_2, v_2'$ and 10 places $H_1, H_2, H_3, H_4, V_1, V_2, C_1, C_2, C_3, C_4$. The initial marking is given by the places containing tokens in the figure.

The trace alphabet (Σ, I) associated with the net system N^* has 12 letters $h_1, h_1', h_2, h_2', h_3, h_3', h_4, h_4', v_1, v_1', v_2, v_2'$. The letter v_1 is dependent from the letters v_1', v_2, v_2' (because of the place V_1 and/or V_2), h_2', and h_4 (because of C_1). The letter h_1 is dependent from the letters h_1', h_4, h_4' (because of the place H_1), h_2', h_2 (because of H_2), h_3', and v_2' (because of C_2). For the remaining letters, the dependency relation is defined in a similar way.

Observe that the letters h_1 and h_3 are independent, but there is no firing trace containing h_1 and h_3 as consecutive letters.

Following [**183**], the *event structure unfolding* of N is the event structure $\mathcal{E}_N = (E, \leq, \#, \lambda)$, where

- E is the set of prime firing traces $\mathcal{PFT}(N)$,
- $\leq$ is $\sqsubseteq$, restricted to $E \times E$,
- Let $e, e' \in E$. Then $e \# e'$ iff there does not exist a firing trace $\langle\sigma\rangle$ such that $e \sqsubseteq \langle\sigma\rangle$ and $e' \sqsubseteq \langle\sigma\rangle$,
- $\lambda : E \to \Sigma$ is given by $\lambda(\langle\sigma\rangle) = last(\sigma)$.

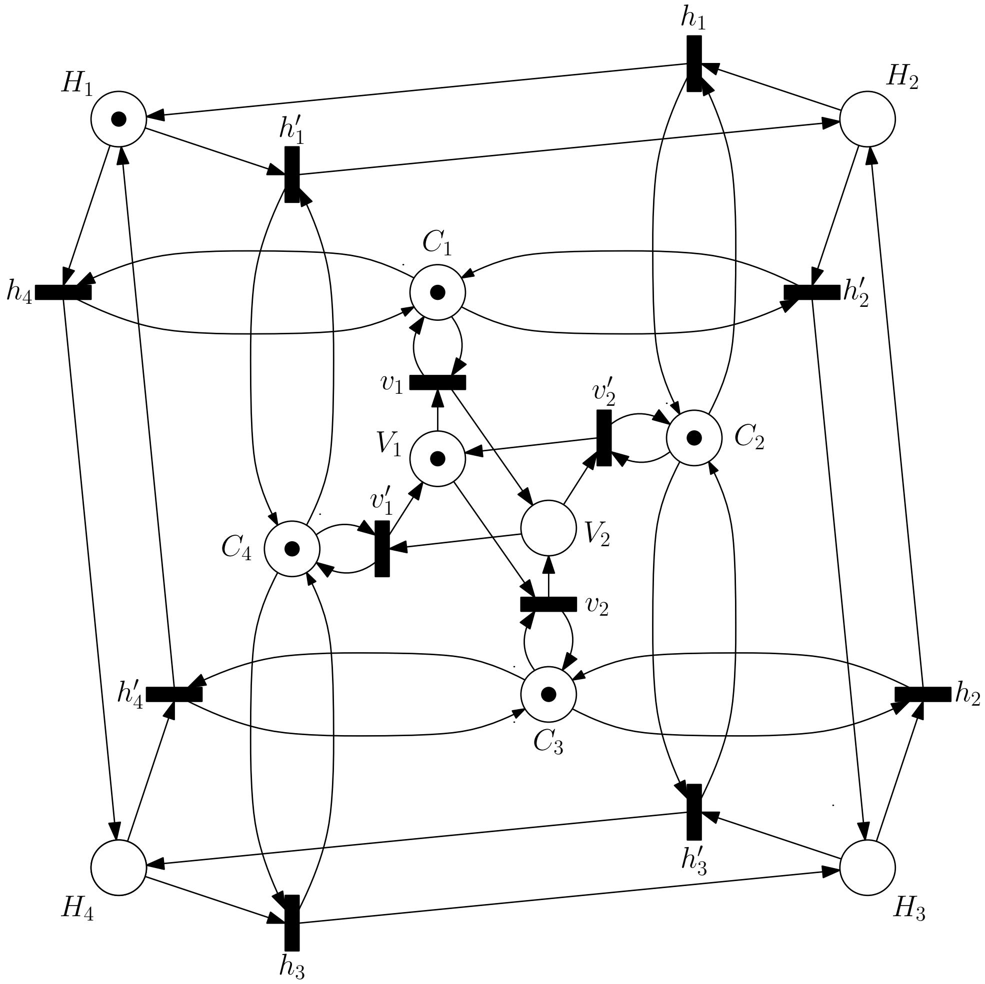


FIGURE 2.2. The net system N^* has 12 transitions (represented by rectangles) and 10 places (represented by circles).

The following result establishes the equivalence between unfoldings of net systems and trace-regular event structures:

THEOREM 2.16 ([**240**, Theorem 1]). *An event structure $\mathcal{E}$ is a trace-regular event structure if and only if there exists a net system N such that $\mathcal{E}$ and $\mathcal{E}_N$ are isomorphic.*

1.5. The MSO Theory of Trace Event Structures. We start with the definition of monadic second-order logic (MSO-logic). Let A be a universe and $\mathcal{A} = (A, (R_i)_{i \in I})$, where $R_i \subseteq A^{n_i}$ for $i \in I$ be a *relational structure*. The MSO logic of $\mathcal{A}$ has two types of variables: individual (or first-order) variables and set (or second-order) variables. The individual variables range over the elements of A and are denoted by x, y, z, etc. The set variables range over subsets of A and are denoted X, Y, Z, etc. *MSO-formulas* over the signature of $\mathcal{A}$ are constructed from the atomic formulas $R_i(x_1, \ldots, x_{n_i})$, $x = y$, and $x \in X$ (where $i \in I$, $x_1, \ldots, x_{n_i}, x, y$ are individual variables and X is a set variable) using the Boolean connectives $\neg, \vee, \wedge$, and quantifications over first order and second order variables. The notions of free variables and bound variables are defined as usual. A formula without free occurrences of variables is called an *MSO-sentence*. If $\varphi(x_1, \ldots, x_n, X_1, \ldots, X_m)$ is an MSO-formula such that at most the individual variables among $x_1, \ldots, x_n$ and the set variables among $X_1, \ldots, X_m$ occur freely in φ, and $a_1, \ldots, a_n \in A$ and $A_1, \ldots, A_m \subseteq A$, then $\mathcal{A} \models \varphi(a_1, \ldots, a_n, A_1, \ldots, A_m)$ means that φ evaluates to true in $\mathcal{A}$ when x_i evaluates to a_i and X_j evaluates to A_j.

The *MSO theory* of $\mathcal{A}$, denoted by MSO($\mathcal{A}$), is the set of all MSO-sentences φ such that $\mathcal{A} \models \varphi$. The MSO theory of $\mathcal{A}$ is said to be *decidable* if there exists an algorithm deciding for each MSO-sentence φ in MSO($\mathcal{A}$), whether $\mathcal{A} \models \varphi$.

Let $\mathcal{E}_N = (E, \leq, \#, \lambda)$ be a trace-regular event structure, which is the event structure unfolding of a net system $N = (S, \Sigma, F, m_0)$ (by Theorem 2.16, any trace-regular event structure admits such a representation). Thiagarajan and Yang [**241**] defined the MSO theory MSO($\mathcal{E}_N$) of $\mathcal{E}_N$ as the MSO theory of the relational structure $(E, (R_a)_{a\in\Sigma}, \leq)$, where E is the set of events, $R_a \subset E$ is the set of a-labeled events for $a \in \Sigma$, and $\leq \subseteq E \times E$ is the precedence relation. The MSO theory of a net system N is the MSO theory of its event structure unfolding [**241**].

As shown in [**241**], the conflict relation $\#$, the concurrency relation $\|$, and the notion of a configuration of $\mathcal{E}$, as well as other connectives of propositional logic such as $\wedge, \Rightarrow$ (implies) and $\equiv$ (if and only if), universal quantification over individual and set variables $(\forall x(\varphi), \forall X(\varphi))$, the set inclusion relation $\subseteq$ $(X \subseteq Y)$, can be defined as well. The conflict and concurrency relations $\#$ and $\|$ of $\mathcal{E}$ are defined in [**241**] in the following way:

- $x\widehat{\#}y := \neg(x \leq y) \wedge \neg(y \leq x) \wedge \bigvee_{(a,b)\in D}(R_a(x) \wedge R_b(y))$.
- $x\#y := \exists x'\exists y'(x' \leq x \wedge y' \leq y \wedge x'\widehat{\#}y')$.
- $x \parallel y := \neg(x \leq y) \wedge \neg(y \leq x) \wedge \neg(x\#y)$.

An interpretation $\mathcal{I}$ assigns to every individual variable an event in E and every set variable, a subset of E. Then $\mathcal{E}_N$ satisfies a formula φ under an interpretation $\mathcal{I}$, denoted by $\mathcal{E}_N \models_{\mathcal{I}} \varphi$, if the following holds [**241**]:

- $\mathcal{E}_N \models_{\mathcal{I}} R_a(x)$ iff $\lambda(\mathcal{I}(x)) = a$.
- $\mathcal{E}_N \models_{\mathcal{I}} x \leq y$ iff $\mathcal{I}(x) \leq \mathcal{I}(y)$.
- $\mathcal{E}_N \models_{\mathcal{I}} x \in X$ iff $\mathcal{I}(x) \in \mathcal{I}(X)$.
- $\mathcal{E}_N \models_{\mathcal{I}} \exists x(\varphi)$ iff there exists $e \in E$ and an interpretation $\mathcal{I}'$ such that $\mathcal{E} \models_{\mathcal{I}} \varphi$ where $\mathcal{I}'$ satisfies the conditions: $\mathcal{I}'(x) = e$, $\mathcal{I}'(y) = \mathcal{I}(y)$ for every individual variable y other than x, and $\mathcal{I}'(X) = \mathcal{I}(X)$ for every set variable X.
- $\mathcal{E}_N \models_{\mathcal{I}} \exists X(\varphi)$ iff there exists $E' \subseteq E$ and an interpretation $\mathcal{I}'$ such that $\mathcal{E} \models_{\mathcal{I}'} \varphi$ where $\mathcal{I}'$ satisfies: $\mathcal{I}'(X) = E'$, $\mathcal{I}'(x) = \mathcal{I}(x)$ for every individual variable x, and $\mathcal{I}'(Y) = \mathcal{I}(Y)$ for every set variable Y other than X.
- $\mathcal{E}_N \models_{\mathcal{I}} \neg\varphi$ and $\mathcal{E} \models_{\mathcal{I}} \varphi_1 \vee \varphi_2$ are defined in the standard way.

$\mathcal{E} \models \varphi$ will denote that $\mathcal{E}$ is a model of the sentence φ.

It turns out that the MSO theory of trace event structures is not always decidable: Figure 1 of [**241**] presented an example of such an event structure suggested by Igor Walukiewicz. To circumvent this example, Thiagarajan and Yang formulated the following notion.

The event structure $\mathcal{E} = (E, \leq, \#)$ is *grid-free* [**241**] if there does not exist three pairwise disjoint sets X, Y, Z of E satisfying the following conditions:

- $X = \{x_0, x_1, x_2, \ldots\}$ is an infinite set of events with $x_0 < x_1 < x_2 < \cdots$.
- $Y = \{y_0, y_1, y_2, \ldots\}$ is an infinite set of events with $y_0 < y_1 < y_2 < \cdots$.
- $X \times Y \subseteq \|$.
- There exists an injective mapping $g : X \times Y \to Z$ satisfying: if $g(x_i, y_j) = z$ then $x_i < z$ and $y_j < z$. Furthermore, if $i' > i$ then $x_{i'} \not< z$ and of $j' > j$ then $y_{j'} \not< z$.

The Σ-labelled event structure $(E, \leq, \#, \lambda)$ is said to be *grid-free* if the unlabeled event structure $(E, \leq, \#)$ is grid-free. The net system N is *grid-free* if the event structure $\mathcal{E}_N$ is grid-free. As noticed in [**241**], Walukiewicz's net system is not grid-free. Thiagarajan and Yang [**241**] proved that if a net system N is not grid-free, then the MSO theory MSO($\mathcal{E}_N$) is not decidable.

2. Domains, Median Graphs, and CAT(0) Cube Complexes

In this section, we recall the bijection between domains of event structures and median graphs/CAT(0) cube complexes established in [**25**]. We also introduce the special cube complexes of Haglund and Wise [**132**, **133**].

2.1. Bijection between Domains and Median Graphs/CAT(0) Cube Complexes. We start by providing some additional properties of median graphs. With any vertex v of a median graph $G = (V, E)$ is associated a *canonical partial order* $\leq_v$ defined by setting $x \leq_v y$ if and only if $x \in I(v, y)$; v is called the *basepoint* of $\leq_v$. Since G is bipartite, the Hasse diagram G_v of the partial order $(V, \leq_v)$ is the graph G in which any edge xy is directed from x to y if and only if the inequality $d_G(x, v) < d_G(y, v)$ holds. We call G_v a *pointed median graph.* There is a close relationship between pointed median graphs and median semilattices. A *median semilattice* is a meet semilattice $(P, \leq)$ such that (i) for every x, the *principal ideal* $\downarrow x = \{p \in P : p \leq x\}$ is a distributive lattice, and (ii) any three elements have a least upper bound in P whenever each pair of them does. The Hasse diagram of any median semilattice is a median graph, and conversely, every median graph defines a median semilattice with respect to any canonical order $\leq_v$ [**13**].

The canonical isometric embedding of a median graph G into a (smallest) hypercube can be determined by the so called *Djoković-Winkler ("parallelism")* relation Θ on the edges of G [**107**, **252**]. For median graphs, the equivalence relation Θ can be defined as follows. First say that two edges uv and xy are in relation Θ' if they are opposite edges of a 4-cycle $uvxy$ in G. Then let Θ be the reflexive and transitive closure of Θ'. Any equivalence class of Θ constitutes a cutset of the median graph G, which determines one factor of the canonical hypercube [**177**]. The cutset (equivalence class) $\Theta(xy)$ containing an edge xy defines a convex split $\{W(x, y), W(y, x)\}$ of G [**177**], where $W(x, y) = \{z \in V : d_G(z, x) < d_G(z, y)\}$ and $W(y, x) = V \setminus W(x, y)$ (we call the complementary convex sets $W(x, y)$ and $W(y, x)$ *halfspaces*). Conversely, for every convex split of a median graph G there exists at least one edge xy such that $\{W(x, y), W(y, x)\}$ is the given split. We denote by $\{\Theta_i : i \in I\}$ the equivalence classes of the relation Θ (in [**25**], they were called parallelism classes). For an equivalence class $\Theta_i, i \in I$, we denote by $\{A_i, B_i\}$ the associated convex split. We say that Θ_i *separates* the vertices x and y if $x \in A_i, y \in B_i$ or $x \in B_i, y \in A_i$. The isometric embedding φ of G into a hypercube is obtained by taking a basepoint v, setting $\varphi(v) = \varnothing$ and for any other vertex u, letting $\varphi(u)$ be all parallelism classes of Θ which separate u from v.

We continue with some additional notions for non-positively curved and CAT(0) cube complexes. A *midcube* of the d-cube c, with $d \geq 1$, is the isometric subspace obtained by restricting exactly one of the coordinates of d to 0. Note that a midcube is a $(d-1)$-cube. The midcubes a and b of a cube complex X are *adjacent* if they have a common face, and a *hyperplane* H of X is a subspace that is a maximal connected union of midcubes such that, if $a, b \subset H$ are midcubes, either a and b are disjoint or they are adjacent. Equivalently, a hyperplane H is a maximal connected union of midcubes such that, for each cube c, either $H \cap c = \varnothing$ or $H \cap c$ is a single midcube of c. The *carrier* $N(X)$ of a hyperplane H of X is the union of all cubes intersecting H.

THEOREM 2.17 ([**218**]). *Each hyperplane H of a CAT(0) cube complex X is a CAT(0) cube complex of dimension at most* $\dim(X) - 1$ *and $X \setminus H$ consists of exactly two components, called* halfspaces.

A 1-cube e (an edge) is *dual* to the hyperplane H if the 0-cubes of e lie in distinct halfspaces of $X \setminus H$, i.e., if the midpoint of e is in a midcube contained in H. The relation "dual to the same hyperplane" is an equivalence relation on the set of edges of X; denote this relation by Θ and denote by $\Theta(H)$ the equivalence class consisting of

1-cubes dual to the hyperplane H (Θ is precisely the parallelism relation on the edges of the median graph $X^{(1)}$).

Theorems 2.2 and 2.3 of Barthélemy and Constantin [**25**] establish the following bijection between event structures and pointed median graphs:

THEOREM 2.18 ([**25**]). *The (undirected) Hasse diagram of the domain* $(\mathcal{D}(\mathcal{E}), \subseteq)$ *of any event structure* $\mathcal{E} = (E, \leq, \#)$ *is a median graph. Conversely, for any median graph* G *and any basepoint* v *of* G*, the pointed median graph* G_v *is isomorphic to the Hasse diagram of the domain of an event structure.*

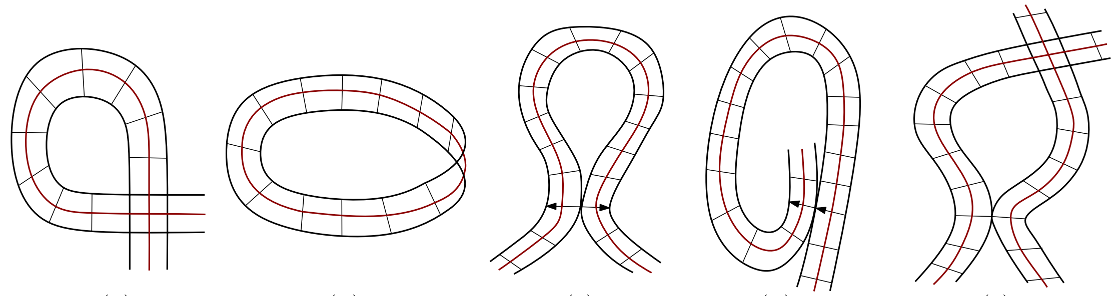
Using the bijection between median graphs and CAT(0) cube complexes of [**82**], we provided a new proof of the first part of Theorem 2.18 in [**51**]. Now, we recall briefly the canonical construction of an event structure from a pointed median graph (or a pointed CAT(0) cube complex) presented in [**25**]. Suppose that v is an arbitrary but fixed basepoint of a median graph G. The events of the event structure $\mathcal{E}_v$ are the hyperplanes of X. Two hyperplanes H and H' define concurrent events if and only if they cross. The hyperplanes H and H' are in precedence relation $H \leq H'$ if and only if $H = H'$ or H separates H' from v. Finally, the events defined by H and H' are in conflict if and only if H and H' do not cross and neither separates the other from v.

2.2. Special Cube Complexes. Consider a cube complex Y. Analogously to CAT(0) cube complexes, one can define the parallelism relation Θ' on the set of edges $E(Y)$ of Y by setting that two edges of Y are in relation Θ' if they are opposite edges of a common 2-cube of Y. Let Θ be the reflexive and transitive closure of Θ' and let $\{\Theta_i : i \in I\}$ denote the equivalence classes of Θ. For an equivalence class Θ_i, the hyperplane H_i associated to Θ_i is the cube complex consisting of the midcubes of all cubes of Y containing one edge of Θ_i. The edges of Θ_i are *dual* to the hyperplane H_i. Let $\mathcal{H}(Y)$ be the set of hyperplanes of Y. The *carrier* $N(H)$ of a hyperplane H of Y is the union of all cubes intersecting H. The *open carrier* $\mathring{N}(H)$ is the union of all open cubes intersecting H.

The hyperplanes of a cube complex Y do not longer satisfy the nice properties of the hyperplanes of CAT(0) cube complexes: they do not partition the complex in exactly two parts, they may self-intersect, self-osculate, two hyperplanes may at the same time cross and osculate, etc. Haglund and Wise [**132**] detected five types of pathologies which may occur in a cube complex (see Figure 2.3):

(a) self-intersecting hyperplane;
(b) one-sided hyperplane;
(c) directly self-osculating hyperplane;
(d) indirectly self-osculating hyperplane;
(e) a pair of hyperplanes, which both intersect and osculate.

A hyperplane H is *two-sided* if $\mathring{N}(H)$ is homeomorphic to the product $H \times (-1, 1)$, and there is a combinatorial map $H \times [-1, 1] \to X$ mapping $H \times \{0\}$ identically to H. As noticed in [**132**, p.1562], requiring that the hyperplanes of Y are two-sided is equivalent to defining an orientation on the dual edges of H such that all sources of such edges belong to one of the sets $H \times \{-1\}, H \times \{1\}$ and all sinks belong to the other one. This orientation is obtained by taking the equivalence relation generated by elementary parallelism relation: declare two oriented edges e_1 and e_2 of Y *elementary parallel* if there is a square of Y containing e_1 and e_2 as opposite sides and oriented in the same direction. Such an orientation o of the edges of Y is called an *admissible* orientation of Y. Observe that Y admits an admissible orientation if and only if every hyperplane H of Y is two-sided (one can choose an admissible orientation for each hyperplane independently). Given a cube complex Y and an admissible orientation o of Y, (Y, o) is called a *directed* cube complex.



(a) (b) (c) (d) (e)

FIGURE 2.3. A self-intersecting hyperplane (a), a one-sided hyperplane (b), a directly self-osculating hyperplane (c), an indirectly self-osculating hyperplane (d), and a pair of hyperplanes that inter-osculate (e).

We continue with the definition of each of the pathologies (in which we closely follow [**132**, Section 3]). The hyperplane is *one-sided* if it is not two-sided (see Figure 2.3(b)).

Two hyperplanes H_1 and H_2 *intersect* if there exists a cube Q and two distinct midcubes Q_1 and Q_2 of Q such that $Q_1 \subseteq H_1$ and $Q_2 \subseteq H_2$, i.e., there exists a square with two consecutive edges e_1, e_2 such that e_1 is dual to H_1 and e_2 is dual to H_2.

A hyperplane H of Y *self-intersects* if it contains more than one midcube from the same cube, i.e., there exist two edges e_1, e_2 dual to H that are consecutive in some square of Y (see Figure 2.3(a)).

Let v be a vertex of Y and let e_1, e_2 be two distinct edges incident to v but such that e_1 and e_2 are not consecutive edges in some square containing v. The hyperplanes H_1 and H_2 *osculate* at (v, e_1, e_2) if e_1 is dual to H_1 and e_2 is dual to H_2. The hyperplane H *self-osculate* at (v, e_1, e_2) if e_1 and e_2 are dual to H. Consider a two-sided hyperplane H and an admissible orientation o of its dual edges. Suppose that H self-osculate at (v, e_1, e_2). If v is the source of both e_1 and e_2 or the sink of both e_1 and e_2, then we say that H *directly self-osculate* at (v, e_1, e_2) (see Figure 2.3(c)). If v is the source of one of e_1, e_2, and the sink of the other, then we say that H *indirectly self-osculate* at (v, e_1, e_2) (see Figure 2.3(d)). Note that a self-osculation of a hyperplane H is either direct or indirect, and this is independent of the orientation of the edges dual to H.

Two hyperplanes H_1 and H_2 *inter-osculate* if they both intersect and osculate (see Figure 2.3(e)).

Haglund and Wise [**132**, Definition 3.2] called a cube complex Y *special* if its hyperplanes are two-sided, do not self-intersect, do not directly self-osculate, and no two hyperplanes inter-osculate. A finite NPC complex X is called *virtually special* [**132**,**133**] if X admits a finite special cover, i.e., there exists a finite special NPC complex Y and a covering map $\varphi : Y \to X$. The definition of a special cube complex Y depends only of the 2-skeleton $Y^{(2)}$ [**132**, Remark 3.4]. Since no hyperplane of Y self-osculate, any special cube complex and its 2-skeleton satisfy the 3-cube condition. In fact, Haglund and Wise proved that special cube complexes can be seen as nonpositively curved complexes [**132**, Lemma 3.13]. More precisely, they show that if X is a special cube complex, then X is contained in a unique smallest nonpositively curved cube complex with the same 2-skeleton as X. In the following, we consider only 2-dimensional special cube complexes, since they can always be canonically completed to NPC complexes that are also special.

A particular class of 2-dimensional cube complexes are the VH-complexes. A square complex X is a *VH-complex* (*vertical-horizontal complex*) if the 1-cells (edges) of X are partitioned into two sets V and H called *vertical* and *horizontal* edges respectively, and the edges in each square alternate between edges in V and H. Notice that if X is a

VH-complex, no three squares may pairwise intersect on three edges with a common vertex, and thus VH-complexes are particular NPC square complexes. A VH-complex X is a *complete square complex* (CSC) [**257**] if any vertical edge and any horizontal edge incident to a common vertex belong to a common square of X. By [**257**, Theorem 3.8], if X is a complete square complex, then the universal cover $\widetilde{X}$ of X is isomorphic to the Cartesian product of two trees.

3. Directed NPC Complexes

Since we can define event structures from their domains, universal covers of NPC complexes represent a rich source of event structures. To obtain regular event structures, it is natural to consider universal covers of finite NPC complexes. Moreover, since domains of event structures are directed, it is natural to consider universal covers of NPC complexes whose edges are directed. However, the resulting directed universal covers are not in general domains of event structures. In particular, the domains corresponding to pointed median graphs given by Theorem 2.18 cannot be obtained in this way. In order to overcome this difficulty, we introduce directed median graphs and directed NPC complexes. Using these notions, one can naturally define regular event structures starting from finite directed NPC complexes.

3.1. Directed Median Graphs. A *directed median graph* is a pair $\overrightarrow{G} = (G, o)$, where G is a median graph and o is an orientation of the edges of G in a such a way that opposite edges of squares of G have the same direction. By transitivity of Θ, all edges from the same parallelism class Θ_i of G have the same direction. Since each Θ_i partitions G into two parts, o defines a partial order $\prec_o$ on the vertex-set of G. For a vertex v of G, let $\mathcal{F}_o(v, G) = \{x \in V : v \prec_o x\}$ be the principal filter of v in the partial order $(V(G), \prec_o)$. For any canonical basepoint order $\leq_v$ of G, $(G, \leq_v)$ is a directed median graph. The converse is obviously not true: the 4-regular tree F_4 directed so that each vertex has two incoming and two outgoing arcs is a directed median graph which is not induced by a basepoint order.

Lemma 2.19. *For any vertex v of a directed median graph $\overrightarrow{G} = (G, o)$, the following holds:*

(1) $\mathcal{F}_o(v, G)$ together with $\prec_o$ is the domain of an event structure;
(2) for any vertex $u \in \mathcal{F}_o(v, G)$, the principal filter $\mathcal{F}_o(u, G)$ is included in $\mathcal{F}_o(v, G)$ and $\mathcal{F}_o(u, G)$ coincides with the principal filter of u with respect to the canonical basepoint order $\leq_v$ on $\mathcal{F}_o(v, G)$.

A *directed (x, y)-path* of a directed median graph $\overrightarrow{G} = (G, o)$ is a (x, y)-path $\pi(x, y) = (x = x_1, x_2, \ldots, x_{k-1}, x_k = y)$ of G in which any edge $x_i x_{i+1}$ is directed in $\overrightarrow{G}$ from x_i to x_{i+1}. One can show that any directed path of a directed median graph $\overrightarrow{G}$ is a shortest path of the median graph G.

3.2. Directed NPC Cube Complexes. A *directed NPC complex* is a directed cube complex (Y, o), where Y is a nonpositively curved cube complex and o is an admissible orientation of Y. Recall that this means that o is an orientation of the edges of Y in a such a way that the opposite edges of the same square of Y have the same direction. For an edge xy, we will denote $o(xy)$ by $\overrightarrow{xy}$ if x is the source and y is the sink of $o(xy)$ and by $\overrightarrow{yx}$ otherwise. Note that there exists an admissible orientation for Y if and only if the hyperplanes of Y are two-sided. An admissible orientation o of Y induces in a natural way an orientation $\widetilde{o}$ of the edges of its universal cover $\widetilde{Y}$, so that $(\widetilde{Y}, \widetilde{o})$ is a directed CAT(0) cube complex and $(\widetilde{Y}^{(1)}, \widetilde{o})$ is a directed median graph.

In the following, we need to consider directed colored NPC complexes and directed colored median graphs. A coloring ν of a directed NPC complex (Y, o) is an arbitrary map $\nu : E(Y) \to \Sigma$ where Σ is a set of colors. Note that a labeling is a coloring, but not the converse: *labelings* are precisely the colorings in which opposite edges of any square have the same color. In the following, we will denote a directed colored NPC complexes by bold letters like $\mathbf{Y} = (Y, o, \nu)$. Sometimes, we need to forget the colors and the orientations of the edges of these complexes. For a complex $\mathbf{Y}$, we denote by Y the complex obtained by forgetting the colors and the orientations of the edges of $\mathbf{Y}$ (Y is called the *support* of $\mathbf{Y}$), and we denote by (Y, o) the directed complex obtained by forgetting the colors of $\mathbf{Y}$. We also consider directed colored median graphs that will be the 1-skeletons of directed colored CAT(0) cube complexes. Again we will denote such directed colored median graphs by bold letters like $\mathbf{G} = (G, o, \nu)$. Note that (uncolored) directed NPC complexes can be viewed as directed colored NPC complexes where all edges have the same color.

When dealing with directed colored NPC complexes, we consider only homomorphisms that preserve the colors and the directions of edges. Since any coloring ν of a directed colored NPC complex Y leads to a coloring of its universal cover $\widetilde{Y}$, one can consider the colored universal cover $\widetilde{\mathbf{Y}} = (\widetilde{Y}, \widetilde{o}, \widetilde{\nu})$ of $\mathbf{Y}$.

Similarly, when we consider principal filters in directed colored median graphs $\mathbf{G} = (G, o, \nu)$, we say that two filters are isomorphic if there is an isomorphism between them that preserves the directions and the colors of the edges.

We now formulate the crucial regularity property of directed colored median graphs $(\widetilde{Y}^{(1)}, \widetilde{o}, \widetilde{\nu})$ when (Y, o, ν) is finite.

LEMMA 2.20 ([**51**]). *If $\mathbf{Y} = (Y, o, \nu)$ is a finite directed colored NPC complex, then $\widetilde{\mathbf{Y}}^{(1)} = (\widetilde{Y}^{(1)}, \widetilde{o}, \widetilde{\nu})$ is a directed median graph with at most $|V(Y)|$ isomorphism types of colored principal filters.*

PROPOSITION 2.21 ([**51**]). *Consider a finite (uncolored) directed NPC complex (Y, o). Then for any vertex $\widetilde{v}$ of the universal cover $\widetilde{Y}$ of Y, the principal filter $\mathcal{F}_{\widetilde{o}}(\widetilde{v}, \widetilde{Y}^{(1)})$ with the partial order $\prec_{\widetilde{o}}$ is the domain of a regular event structure with at most $|V(Y)|$ different isomorphism types of principal filters.*

We will call an event structure $\mathcal{E} = (E, \leq, \#)$ and its domain $\mathcal{D}(\mathcal{E})$ *strongly regular* if $\mathcal{D}(\mathcal{E})$ is isomorphic to a principal filter of the universal cover of some finite directed NPC complex. In view of Proposition 2.21, any strongly regular event structure is regular.

4. Directed Special Cube Complexes

4.1. Trace Labelings of Directed Special Cube Complexes. Consider a finite NPC complex Y and let $\mathcal{H} = \mathcal{H}(Y)$ be the set of hyperplanes of Y. We define a canonical labeling $\lambda_{\mathcal{H}} : E(Y) \to \mathcal{H}$ by setting $\lambda_{\mathcal{H}}(e) = H$ if the edge e is dual to H. For any covering map $\varphi : \widetilde{Y} \to Y$, $\lambda_{\mathcal{H}}$ is naturally extended to a labeling $\widetilde{\lambda}_{\mathcal{H}}$ of $E(\widetilde{Y})$ by setting $\widetilde{\lambda}_{\mathcal{H}}(e) = \lambda_{\mathcal{H}}(\varphi(e))$.

We call a strongly regular event structure $\mathcal{E} = (E, \leq, \#)$ and its domain $\mathcal{D}(\mathcal{E})$ *cover-special* if $\mathcal{D}(\mathcal{E})$ is isomorphic to a principal filter of the universal cover of some virtually special complex with an admissible orientation.

Let Y be a finite cube complex with two-sided hyperplanes and let o be an admissible orientation of Y. Since the hyperplanes of Y are two-sided, there exists a bijection between the labelings of the edges of Y (i.e., colorings in which opposite edges of each square have equal colors) and the labelings of the hyperplanes of Y. Let $M = (\Sigma, I)$ be a trace alphabet. Extending the definition of trace labelings of domains of event structures (pointed CAT(0) cube complexes), we call a labeling $\lambda : E(Y) \to \Sigma$ of (Y, o) a *trace labeling* if the following conditions hold:

(TL1) if there exists a square of Y in which two opposite edges are labeled a and two other opposite edges are labeled b, then $(a, b) \in I$;

(TL2) for any vertex v of Y, any two distinct outgoing edges $\overrightarrow{vx}, \overrightarrow{vy}$ have different labels and $(\lambda(\overrightarrow{vx}), \lambda(\overrightarrow{vy})) \in I$ iff $\overrightarrow{vx}$ and $\overrightarrow{vy}$ belong to a common square of Y;
(TL3) $(\lambda(\overrightarrow{xv}), \lambda(\overrightarrow{vy})) \in I$ iff $\overrightarrow{xv}$ and $\overrightarrow{vy}$ belong to a common square of Y;
(TL4) for any vertex v of Y, any two distinct incoming edges $\overrightarrow{xv}, \overrightarrow{yv}$ have different labels and $(\lambda(\overrightarrow{xv}), \lambda(\overrightarrow{yv})) \in I$ iff $\overrightarrow{xv}$ and $\overrightarrow{yv}$ belong to a common square of Y.

Since for a trace labeling λ all edges dual to a hyperplane of Y have the same label, λ defines in a canonical way a labeling $\lambda : \mathcal{H} \to \Sigma$ of the hyperplanes $\mathcal{H}$ of Y: for a hyperplane H, $\lambda(H) = \lambda(e)$ for any edge e dual to H. Notationally, for an edge xy of Y directed from x to y and its dual hyperplane H, we will write $\lambda(xy) = \lambda(\overrightarrow{xy}) = \lambda(H)$ to denote the (same) label of xy, $\overrightarrow{xy}$, and H.

REMARK 2.22. Notice that (TL1) is a consequence of the three other axioms (TL2)–(TL4). Observe that (TL2)–(TL4) are equivalent to the condition that for any two incident edges $e_1, e_2 \in Y$, $(\lambda(e_1), \lambda(e_2)) \in I$ iff e_1 and e_2 belong to a common square of Y. Consequently, for any two letters $a, b \in \Sigma$ such that there are no hyperplanes $H_a, H_b \in \mathcal{H}$ labeled respectively a and b and intersecting or osculating, the axioms (TL1)–(TL4) hold for a and b, no matter whether (a, b) is in I or in D.

The existence of trace labelings characterizes the special cube complexes among finite cube complexes:

THEOREM 2.23. *For a finite cube complex Y with two-sided hyperplanes the following conditions are equivalent:*

(1) Y is special;
(2) for any admissible orientation o of Y there exists a trace labeling λ of (Y, o);
(3) there exists an admissible orientation o of Y such that (Y, o) admits a trace labeling.

To prove that (3) ⇒ (1), we assume that there is a self-intersecting hyperplane, or a directly self-osculating hyperplane, or two inter-osculating hyperplanes in Y, and we show that in any of these cases, one of the conditions (TL2)–(TL4) is not satisfied at the osculating (or intersecting) vertex. The implication (2) ⇒ (3) is trivial and the implication (1) ⇒ (2) follows from the following proposition:

PROPOSITION 2.24. *Let Y be a special cube complex, $M = (\Sigma, I)$ be a trace alphabet, and $\lambda : E(Y) \to \Sigma$ be a trace labeling of Y. Then for any admissible orientation o of Y and any principal filter $\mathcal{D} = (\mathcal{F}_{\widetilde{o}}(\widetilde{v}, \widetilde{Y}^{(1)}), \prec_{\widetilde{o}})$ of $\widetilde{\mathbf{Y}} = (\widetilde{Y}, \widetilde{o})$, the labeling $\widetilde{\lambda}$ is a trace-regular labeling of $\mathcal{D}$ with the trace alphabet (Σ, I).*

Since Conditions (TL1)–(TL4) are local conditions that are preserved by covering maps, if λ is a trace labeling of a directed special cube complex (Y, o), then the labeling $\widetilde{\lambda}$ of the universal cover $(\widetilde{Y}, \widetilde{o})$ lifted from λ is a trace labeling. The proposition follows then from the fact that Conditions (LES1)–(LES3) are consequences of Conditions (TL1)–(TL4).

As a corollary of Proposition 2.24, we obtain the following result:

COROLLARY 2.25. *Any cover-special event structure $\mathcal{E}$ admits a trace-regular labeling, i.e., Thiagarajan's Conjecture 2.3 is true for cover-special event structures.*

As a corollary of Theorem 2.23 and Thiagarajan's Theorem 2.2, we obtain the following result:

COROLLARY 2.26. *For any (virtually) special cube complex Y, any admissible orientation o of Y, and any vertex $\widetilde{v}$ in the universal cover of $(\widetilde{Y}, \widetilde{o})$, there exists a finite 1-safe Petri net N such that the principal filter $(\mathcal{F}_{\widetilde{o}}(\widetilde{v}, \widetilde{Y}^{(1)}), \prec_{\widetilde{o}})$ is the domain of the event structure unfolding $\mathcal{E}_N$*

4.2. Strongly Hyperbolic Regular Event Structures. In this subsection, we show that Thiagarajan's conjecture holds for a large and natural class of strongly regular event structures, namely those arising from hyperbolic CAT(0) cube complexes. It turns out that strongly hyperbolic regular event structures are cover-special. This is a consequence of the solution by Agol [**4**] of the virtual Haken conjecture for hyperbolic 3-manifolds. This breakthrough result of Agol is based on the theory of special cube complexes developed by Haglund and Wise [**132**, **133**].

Similarly to nonpositive curvature, Gromov hyperbolicity is defined in metric terms (see Chapter 3). However, as for the CAT(0) property, the hyperbolicity of a CAT(0) cube complex can be expressed in a purely combinatorial way. In case of median graphs, i.e., of 1-skeletons of CAT(0) cube complexes, the hyperbolicity can be characterized in the following way:

Lemma 2.27 ([**84**, **129**]). *Let X be a CAT(0) cube complex. Then its 1-skeleton $X^{(1)}$ is δ-hyperbolic if and only if all isometrically embedded square grids are uniformly bounded.*

We call an event structure $\mathcal{E} = (E, \leq, \#)$ and its domain $\mathcal{D}(\mathcal{E})$ *hyperbolic* if $\mathcal{D}(\mathcal{E})$ is isomorphic to a principal filter of a directed CAT(0) cube complex, whose 1-skeleton is hyperbolic. We call an event structure $\mathcal{E} = (E, \leq, \#)$ and its domain $\mathcal{D}(\mathcal{E})$ *strongly hyperbolic regular* if there exists a finite directed NPC complex (X, o) such that $\widetilde{X}$ is hyperbolic and $\mathcal{D}$ is a principal filter of $(\widetilde{X}^{(1)}, \widetilde{o})$. Note that an event structure can be strongly regular and hyperbolic without being strongly regular hyperbolic (see Remark 2.68).

The main result of this subsection is based on the following very deep and important result of Agol [**4**], following much work of Haglund and Wise [**132**, **133**]. Agol's original result is formulated in group-theoretical terms. Its following reformulation (see, for example, [**41**, Theorem 6.7]) in the particular case of finite NPC complexes is particularly appropriate for our purposes:

Theorem 2.28 ([**4**]). *Let X be a finite nonpositively curved cube complex. If the fundamental group $\pi_1(X)$ of X is hyperbolic, then X is virtually special.*

The condition that $\pi_1(X)$ is hyperbolic is equivalent to the fact that the universal cover $\widetilde{X}$ of X is hyperbolic. Therefore, any finite NPC complex X that has a hyperbolic universal cover is virtually special. Consequently, any strongly hyperbolic regular event structure is cover-special, and thus the following theorem is a corollary of Theorems 2.25 and 2.28.

Theorem 2.29. *Any strongly hyperbolic regular event structure admits a trace-regular labeling, i.e., Thiagarajan's Conjecture 2.3 is true for strongly hyperbolic regular event structures.*

5. 1-Safe Petri Nets and Special Cube Complexes

5.1. The Results. Corollary 2.26 enables to associate a finite 1-safe Petri net to any finite directed special cube complex. In this subsection, we show a converse construction: namely, to any net system $N = (S, \Sigma, F, m_0)$, we associate a finite directed special cube complex $\mathbf{X}_N = (X_N, o)$ with a trace labeling $\lambda_N : E(X_N) \to \Sigma$ such that the domain $\mathcal{D}(\mathcal{E}_N)$ of the event structure unfolding $\mathcal{E}_N$ of N is a principal filter of the universal cover $\widetilde{\mathbf{X}}_N$ of $\mathbf{X}_N$.

Let $N = (S, \Sigma, F, m_0)$ be a net system. The transition relation $\longrightarrow \subseteq 2^S \times \Sigma \times 2^S$ defines a directed graph whose vertices are all markings of N and there is an arc from a marking m to a marking m' iff there exists a transition $a \in \Sigma$ such that $m \xrightarrow{a} m'$ (i.e., ${}^\bullet a \subseteq m$, $(a^\bullet - {}^\bullet a) \cap m = \varnothing$, and $m' = (m - {}^\bullet a) \cup a^\bullet$). Denote by G_N the connected component of the support of this graph that contains the initial marking m_0 and call

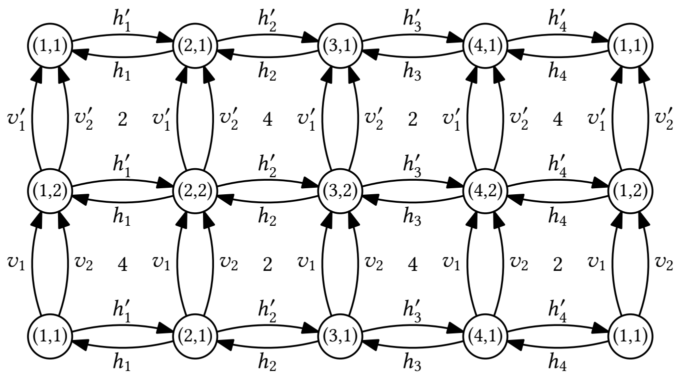


FIGURE 2.4. The special cube complex of the net system N^*. A vertex labeled (i, j) corresponds to the marking $\{H_i, V_j, C_1, C_2, C_3, C_4\}$ of N.

the undirected graph G_N the *marking graph* of N. Let $\vec{G}_N = (G_N, o)$ denote G_N whose edges are oriented according to $\longrightarrow$ (for notational conveniences we use o instead of $\longrightarrow$) and call $\vec{G}_N$ the *directed marking graph*. The marking graph G_N contains all markings reachable from m_0 but also it may contain other markings. Notice also that the directed marking graph $\vec{G}_N$ is deterministic and codeterministic, i.e., for any vertex m and any transition $a \in \Sigma$ there exists at most one arc $m \xrightarrow{a} m'$ and at most one arc $m'' \xrightarrow{a} m$. We will say that two distinct transitions $a, b \in \Sigma$ are *independent* if $({}^\bullet a \cup a^\bullet) \cap ({}^\bullet b \cup b^\bullet) = \varnothing$. Consider the trace alphabet (Σ, I) where $(a, b) \in I$ if and only if the transitions a and b are independent.

DEFINITION 2.30. The 2-dimensional *cube complex* X_N of N is defined in the following way. The 0-cubes and the 1-cubes of X_N are the vertices and the edges of the marking graph G_N. A 4-cycle (m, m_1, m', m_2) of G_N defines a square of X_N iff there exist two (necessarily distinct) independent transitions $a, b \in \Sigma$ such that $m \xrightarrow{a} m_1, m \xrightarrow{b} m_2, m_1 \xrightarrow{b} m'$, and $m_2 \xrightarrow{a} m'$.

The cube complex X_N can be transformed into a directed and colored cube complex $\mathbf{X}_N = (X_N, o, \lambda_N)$: an edge mm' of G_N is oriented from m to m' and $\lambda_N(mm') = a$ iff $m \xrightarrow{a} m'$ holds (clearly, Σ is the set of colors).

THEOREM 2.31. *(X_N, o) is a finite directed special cube complex with two-sided hyperplanes and λ_N is a trace labeling of X_N with the trace alphabet (Σ, I).*

To prove this theorem, we show that the hyperplanes of X_N are two-sided and that λ_N is a trace labeling of (X_N, o) by verifying the Conditions (TL1)–(TL4). Theorem 2.31 then follows from Theorem 2.23.

EXAMPLE 2.32. The special cube complex X_{N^*} of the net system N^* from Example 2.15 is representend in Figure 2.4. In the figure, the leftmost vertices (respectively, edges) should be identified with the rightmost vertices (respectively, edges) that have the same label. Similarly, the lower vertices and edges should be identified with the uppper vertices and edges. The complex X_{N^*} has 8 vertices, 32 edges, and 24 squares. A 4-cycle in the figure is a square of X_{N^*} if opposite edges have the same label and if the labels appearing on the edges of the square correspond to independent transitions of N^*. For example, on the right bottom corner, the directed 4-cycle labeled by h_4 and v_1 is not a square of X_{N^*} because the transitions h_4 and v_1 are not independent (as explained in Example 2.15). The number (2 or 4) in the middle of each 4-cycle represent the number of squares of X_{N^*} on the vertices of this 4-cycle.

Let $\widetilde{X}_N$ denotes the universal cover of the special cube complex X_N and let $\varphi : \widetilde{X}_N \to X_N$ denotes the covering map. Let $\widetilde{\mathbf{X}}_N = (\widetilde{X}_N, \widetilde{o}, \widetilde{\lambda}_N)$ be the directed colored CAT(0) cube complex, in which the orientation and the coloring are defined as in Section 4. For any lift $\widetilde{m}_0$ of m_0, denote by $\mathcal{E}_{X_N} = (E', \leq', \#', \widetilde{\lambda}_N)$ the Σ-labeled event structure whose domain is the principal filter $\mathcal{F}_{\widetilde{o}}(\widetilde{m}_0, \widetilde{X}_N^{(1)})$. Finally, let $\mathcal{E}_N = (E, \leq, \#, \lambda)$ be the event structure unfolding of N as defined in Subsection 1.4 and denote by $\mathcal{D}(\mathcal{E}_N)$ the domain of $\mathcal{E}_N$. The main result of this section is the following theorem:

THEOREM 2.33. *The event structures $\mathcal{E}_N = (E, \leq, \#, \lambda)$ and $\mathcal{E}_{X_N} = (E', \leq', \#', \widetilde{\lambda}_N)$ are isomorphic.*

By Theorem 2.31 and Corollary 2.26, we obtain a correspondence between trace-regular event structures and special cube complexes, leading to the following corollary:

COROLLARY 2.34. *Any trace-regular event structure is cover-special, and thus strongly regular.*

5.2. Geodesic Traces and Prime Traces. Let $M = (\Sigma, I)$ be a trace alphabet and let $\mathcal{E} = (E, \leq, \#)$ be an M-labeled event structure. Let $\mathcal{D}(\mathcal{E})$ denote the domain of $\mathcal{E}$ and let $G(\mathcal{E})$ and $X(\mathcal{E})$ denote the associated median graph and CAT(0) cube complex pointed at vertex v_0. Recall that any vertex v of the median graph $G(\mathcal{E})$ corresponds to a configuration $c(v)$ of $\mathcal{D}(\mathcal{E})$; in particular, $c(v_0) = \varnothing$.

Any shortest path $\pi = (v_0, v_1, \ldots, v_{k-1}, v_k = v)$ from v_0 to a vertex v of $G(\mathcal{E})$ gives rise to an word $\sigma(\pi)$ of Σ^*: the ith letter of $\sigma(\pi)$ is the label $\lambda(v_{i-1}v_i)$ of the edge $v_{i-1}v_i$. We will say that a word $\sigma \in \Sigma^*$ is *geodesic* if $\sigma = \sigma(\pi)$ for a shortest path π between v_0 and a vertex v of $G(\mathcal{E})$. The trace $\langle\sigma\rangle$ of a geodesic word σ is called a *geodesic trace.*

Two shortest (v_0, v)-paths π and π' of $G(\mathcal{E})$ are called *homotopic* if they can be transformed one into another by a sequence of elementary homotopies, i.e., there exists a finite sequence $\pi =: \pi_1, \pi_2, \ldots, \pi_{k-1}, \pi_k := \pi'$ of shortest (v_0, v)-paths such that for any $i = 1, \ldots, k-1$ the paths π_i and π_{i+1} differ only in a square $Q_i = (v_{j-1}, v_j, v_{j+1}, v'_j)$ of $X(\mathcal{E})$. Note that since $X(\mathcal{E})$ is simply connected, any two shortest (v_0, v)-paths are homotopic. Moreover, for any shortest (v_0, v)-path π, the geodesic trace $\langle\sigma(\pi)\rangle$ is exactly $\{\sigma(\pi') : \pi' \text{ is a } (v_0, v)\text{-shortest path}\}$, that we denote by $\langle\sigma_v\rangle$. This gives a natural bijection between the set of geodesic traces of $\mathcal{E}$ and the vertices of $G(\mathcal{E})$. Precedence and conflict relations between geodesic traces can be characterized geometrically in the following way:

LEMMA 2.35. *For two geodesic traces $\langle\sigma_u\rangle$ and $\langle\sigma_v\rangle$, we have*
- *$\langle\sigma_u\rangle \sqsubseteq \langle\sigma_v\rangle$ iff $u \in I(v_0, v)$,*
- *$\langle\sigma_u\rangle$ and $\langle\sigma_v\rangle$ are in conflict iff there does not exists a vertex w such that $u, v \in I(v_0, w)$*

Recall that a trace $\langle\sigma\rangle$ is *prime* if σ is non-null and for every $\sigma' \in \langle\sigma\rangle$, $last(\sigma) = last(\sigma')$. We call an interval $I(v_0, v)$ *prime* if the vertex v has degree 1 in the subgraph induced by $I(v_0, v)$. One can show that a geodesic trace $\langle\sigma_v\rangle$ is prime iff the interval $I(v_0, v)$ is prime. Denote by $\mathcal{PGT}(\mathcal{E})$ the set of prime geodesic traces of $\mathcal{E}$. There exists a bijection between the hyperplanes of $X(\mathcal{E})$ and the prime geodesic traces of $\mathcal{E}$:

LEMMA 2.36. *Each hyperplane H of $X(\mathcal{E})$ (i.e., each event of $\mathcal{E}$) gives a unique prime geodesic trace $\langle\sigma_H\rangle := \langle\sigma_v\rangle$ defined by the prime interval $I(v_0, v)$, where v' is the gate of v_0 in the carrier $N(H)$ of the hyperplane H and v is the neighbor of v' such that the edge $v'v$ is dual to H. Conversely, for each prime geodesic trace $\langle\sigma_u\rangle$ there exists a unique hyperplane H such that $\langle\sigma_u\rangle = \langle\sigma_H\rangle$.*

This bijection enables to characterize geometrically the precedence relation among prime geodesic traces:

LEMMA 2.37. *For two hyperplanes H_1, H_2 of $X(\mathcal{E})$ with prime geodesic traces $\langle\sigma_{v_1}\rangle$ and $\langle\sigma_{v_2}\rangle$, respectively, $H_1 \leq H_2$ holds iff $\langle\sigma_{v_1}\rangle \sqsubseteq \langle\sigma_{v_2}\rangle$.*

5.3. Sketch of the Proof of Theorem 2.33. Consider a net system $N = (S, \Sigma, F, m_0)$, the trace alphabet $M = (\Sigma, I)$ associated to N, its set of firing traces $\mathcal{F}T(N)$, and its set of prime firing traces $\mathcal{P}FT(N)$. Let $\mathcal{E}_N = (E, \leq, \#, \lambda)$ be the M-labeled event structure unfolding of a net system N (recall that $E = \mathcal{P}FT(N)$ and that $\lambda(\langle\sigma\rangle) = last(\sigma)$ for every $\langle\sigma\rangle \in E$).

Let also $\mathcal{E}_{X_N} = (E', \leq', \#', \widetilde{\lambda}_N)$ be the Σ-labeled event structure whose domain is the principal filter $\mathcal{F}_{\widetilde{o}}(\widetilde{m}_0, \widetilde{X}_N^{(1)})$ of the universal cover $\widetilde{\mathbf{X}}_N = (\widetilde{X}_N, \widetilde{o}, \widetilde{\lambda}_N)$ of the special cube complex (X_N, o, λ_N) of N. Let $\varphi : \widetilde{X}_N \mapsto X_N$ denote the covering map. Let $G(\mathcal{E}_{X_N})$ denotes the median graph of $\mathcal{E}_{X_N}$. From Theorem 2.31 and Proposition 2.24 it follows that $\widetilde{\lambda}_N$ is a trace labeling of the event structure $\mathcal{E}_{X_N}$. As explained in Subsection 5.2, there is a bijection between the configurations $\mathcal{E}_{X_N}$ and the geodesic traces of $\mathcal{E}_{X_N}$ and a bijection between the hyperplanes (events) of $\mathcal{E}_{X_N}$ and the prime geodesic traces of $\mathcal{E}_{X_N}$.

The following lemma establishes a bijection between the geodesic traces of $\mathcal{E}_{X_N}$ and the firing traces of N.

Lemma 2.38. *Any geodesic trace $\langle\sigma_{\widetilde{m}}\rangle$ of $\mathcal{E}_{X_N}$ is a firing trace of N. Conversely, for any firing trace $\langle\sigma\rangle$ there exists a geodesic trace $\langle\sigma_{\widetilde{m}}\rangle$ such that $\langle\sigma\rangle = \langle\sigma_{\widetilde{m}}\rangle$.*

In particular, there is a bijection between prime geodesic traces of $\mathcal{E}_{X_N}$ and the prime firing traces of N.

Consequently, there is a label-preserving bijection between the hyperplanes (events) of $\mathcal{E}_{X_N}$ and the hyperplanes (events) of $\mathcal{E}_N$. Therefore, to establish that the event structures $\mathcal{E}_N$ and $\mathcal{E}_{X_N}$ are isomorphic it remains to show that this bijection preserves the precedence and the conflict relations.

Consider two hyperplanes H_1, H_2 of $\mathcal{E}_{X_N}$ with respective prime geodesic traces $\langle\sigma_{\widetilde{v}_1}\rangle$ and $\langle\sigma_{\widetilde{v}_2}\rangle$. By Lemma 2.37, we have $H_1 \leq H_2$ iff $\langle\sigma_{\widetilde{u}}\rangle \sqsubseteq \langle\sigma_{\widetilde{v}}\rangle$. This shows that the precedence relation is preserved, since the prefix relation $\sqsubseteq$ is the precedence relation for $\mathcal{E}_N$. By definition of $\widetilde{v}_1$ and $\widetilde{v}_2$ (see Lemma 2.36) , H_1 and H_2 are in conflict if there is no $\widetilde{w}$ such that $\widetilde{v}_1, \widetilde{v}_2 \in I(\widetilde{m}_0, \widetilde{w})$. By Lemma 2.35, this holds if and only if there is no $\widetilde{w}$ such that $\langle\sigma_{\widetilde{u}}\rangle \sqsubseteq \langle\sigma_{\widetilde{w}}\rangle$ and $\langle\sigma_{\widetilde{v}}\rangle \sqsubseteq \langle\sigma_{\widetilde{w}}\rangle$. This shows that the conflict relation is preserved, since this is the definition of the conflict relation for $\mathcal{E}_N$. This proves that the bijection between geodesic traces and firing traces preserves the conflict relations in $\mathcal{E}_{X_N}$ and $\mathcal{E}_N$ and finishes the proof of Theorem 2.33.

6. Counterexamples to Thiagarajan's Conjecture on Regular Event Structures

In this section, we construct the domain $(\widetilde{W}_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ of a regular event structure (with bounded $\natural$-cliques) that does not admit a regular nice labeling, providing a counterexample to Conjectures 2.3 and 2.4. We also show that other counterexamples to Conjecture 2.3 arise from 4-way deterministic aperiodic tile sets.

6.1. Wise's Square Complex X and its Universal Cover $\widetilde{\mathbf{X}}$. We start with a directed colored CSC (complete square complex) **X** introduced by Wise [**257**]. Recall that in such complexes, the edges are classified vertical or horizontal, each edge has an orientation and a color, and any two incident edges belong to a square. The complex **X** consists of six squares as indicated in Figure 2.5 (reproducing Figure 3 of [**257**]). Each square has two vertical and two horizontal edges. The horizontal edges are oriented from left to right and vertical edges from bottom to top. Denote this orientation of edges by o. The vertical edges of squares are colored white, grey, and black and denoted a, b, and c, respectively. The horizontal edges of squares are colored by single or double arrow, and denoted x and y, respectively. The six squares are glued together by identifying edges of the same color and respecting the directions to obtain the square complex **X**. Note

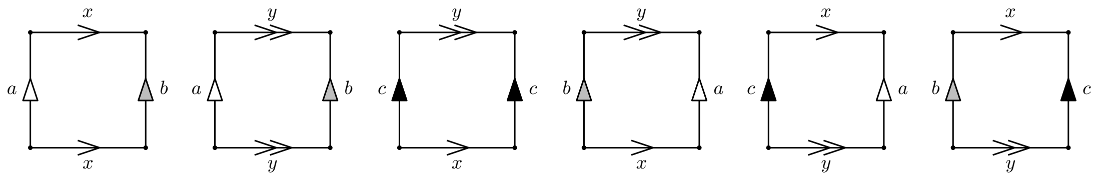


FIGURE 2.5. The 6 squares defining the complex **X**

that **X** has a unique vertex, five edges, and six squares. It can be directly checked that **X** is a complete square complex, and consequently (X, o) is a directed NPC complex. Let H_X denote the subcomplex of X consisting of the 2 horizontal edges and let V_X denote the subcomplex of X consisting of the 3 vertical edges.

The universal cover $\widetilde{H}_X$ of H_X is the 4-regular infinite tree F_4. Its edges inherit the orientations from their images in H_X: each vertex of $\widetilde{H}_X$ has two incoming and two outgoing arcs. Analogously, the universal cover $\widetilde{V}_X$ of V_X is the 6-regular infinite tree F_6 where each vertex has three incoming and three outgoing arcs. Let $\widetilde{v}_1$ be any vertex of $\widetilde{H}_X$. Then the principal filter of $\widetilde{v}_1$ is the infinite binary tree T_2 rooted at $\widetilde{v}_1$: all its vertices except $\widetilde{v}_1$ have one incoming and two outgoing arcs, while $\widetilde{v}_1$ has two outgoing arcs and no incoming arc. Analogously, the principal filter of any vertex $\widetilde{v}_2$ in the ordered set $\widetilde{V}_X$ is the infinite ternary tree T_3 rooted at $\widetilde{v}_2$.

Let $\widetilde{\mathbf{X}}$ be the universal cover of **X** and let $\varphi : \widetilde{\mathbf{X}} \to \mathbf{X}$ be a covering map. Let $\widetilde{X}$ denote the support of $\widetilde{\mathbf{X}}$. Since **X** is a CSC, by [**257**, Theorem 3.8], $\widetilde{X}$ is the Cartesian product $F_4 \times F_6$ of the trees F_4 and F_6. The edges of $\widetilde{\mathbf{X}}$ are colored and oriented as their images in **X**, and are also classified as horizontal or vertical edges. The squares of $\widetilde{\mathbf{X}}$ are oriented as their images in **X**, thus two opposite edges of the same square of $\widetilde{\mathbf{X}}$ have the same direction. This implies that all classes of parallel edges of $\widetilde{\mathbf{X}}$ are oriented in the same direction. Denote this orientation of the edges of $\widetilde{\mathbf{X}}$ by $\widetilde{o}$. The 1-skeleton $\widetilde{X}^{(1)}$ of $\widetilde{X}$ together with $\widetilde{o}$ is a directed median graph. Let $\widetilde{v} = (\widetilde{v}_1, \widetilde{v}_2)$ be any vertex of $\widetilde{X}$, where $\widetilde{v}_1$ and $\widetilde{v}_2$ are the coordinates of $\widetilde{v}$ in the trees F_4 and F_6. Then the principal filter $\mathcal{F}_{\widetilde{o}}(\widetilde{v}, \widetilde{X}^{(1)})$ of $\widetilde{v}$ is the Cartesian product of the principal filters of $\widetilde{v}_1$ in F_4 and of $\widetilde{v}_2$ in F_6, i.e., is isomorphic to $T_2 \times T_3$.

By Lemma 2.19, the orientation of the edges of $\mathcal{F}_{\widetilde{o}}(\widetilde{v}, \widetilde{X}^{(1)})$ corresponds to the canonical basepoint orientation of $\mathcal{F}_{\widetilde{o}}(\widetilde{v}, \widetilde{X}^{(1)})$ with $\widetilde{v}$ as the basepoint. Moreover, by Proposition 2.21, $\mathcal{F}_{\widetilde{o}}(\widetilde{v}, \widetilde{X}^{(1)})$ is the domain of a regular event structure with one isomorphism type of principal filters. We summarize this in the following result:

LEMMA 2.39. *For any vertex $\widetilde{v}$ of $\widetilde{X}$, $\mathcal{F}_{\widetilde{o}}(\widetilde{v}, \widetilde{X}^{(1)})$ is the domain of a regular event structure with one isomorphism class of futures.*

6.2. Aperiodicity of $\widetilde{\mathbf{X}}$. We recall here the main properties of $\widetilde{\mathbf{X}}$ established in [**257**, Section 5]. Let $\widetilde{v} = (\widetilde{v}_1, \widetilde{v}_2)$ be an arbitrary vertex of $\widetilde{\mathbf{X}}$, where $\widetilde{v}_1$ and $\widetilde{v}_2$ are defined as before. From the definition of the covering map, the loop of **X** colored y gives rise to a bi-infinite horizontal path P_y of $\widetilde{\mathbf{X}}^{(1)}$ passing via $\widetilde{v}$ and whose all edges are colored y and are directed from left to right. Analogously, there exists a bi-infinite vertical path P_c of $\widetilde{\mathbf{X}}^{(1)}$ passing via $\widetilde{v}$ and whose all edges are colored c and are directed from bottom to top.

The projection of P_y on the horizontal factor F_4 is a bi-infinite path P^h of F_4 passing via $\widetilde{v}_1$. Analogously, the projection of P_c on the vertical factor F_6 is a bi-infinite path P^v of F_6 passing via $\widetilde{v}_2$. Consequently, the convex hull $\text{conv}(P_y \cup P_c)$ of $P_y \cup P_c$ in the graph $\widetilde{\mathbf{X}}^{(1)}$ is isomorphic to the Cartesian product of $P^h \times P^v$ of the paths P^h and P^v. Therefore the subcomplex of $\widetilde{\mathbf{X}}$ spanned by $\text{conv}(P_y \cup P_c)$ is a directed plane Π_{yc}

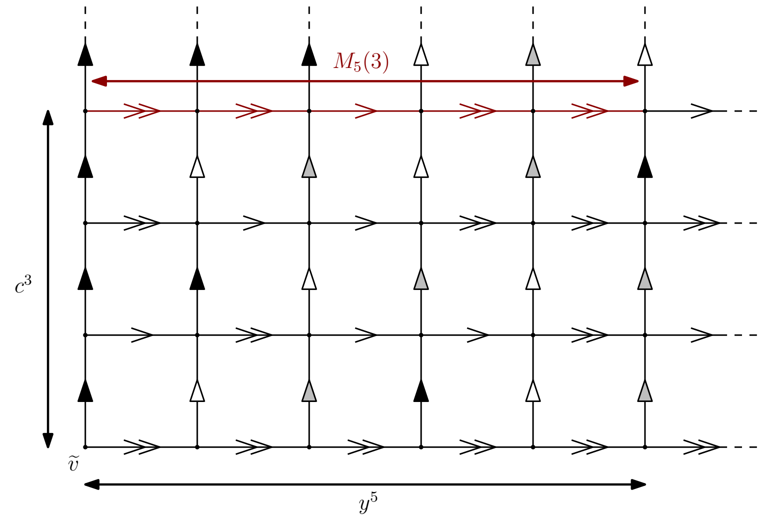


FIGURE 2.6. Part of the plane Π_{yc}^{++} appearing in $\widetilde{\mathbf{X}}$

tiled into squares (recall that each square is of one of 6 types and its sides are colored by the letters a, b, c, x, y), see Figure 2.6. Wise showed that the plane Π_{yc} is not tiled periodically by the preimages of the squares of $\mathbf{X}$.

THEOREM 2.40 ([**257**, Theorem 5.3]). *The plane Π_{yc} tiled into squares is not doubly periodic.*

In our counterexample we will use the following result of [**257**] that was used to show that the plane Π_{yc} is not tiled periodically by the preimages of the squares of $\mathbf{X}$. Denote by P_y^+ the (directed) subpath of P_y having $\widetilde{v}$ as a source (this is a one-infinite horizontal path). Analogously, let P_c^+ be the (directed) subpath of P_c having $\widetilde{v}$ as a source. The convex hull of $P_y^+ \cup P_c^+$ is a quarter of the plane Π_{yc}, which we denote by Π_{yc}^{++}. Any shortest path in $\widetilde{\mathbf{X}}^{(1)}$ from $\widetilde{v}$ to a vertex $\widetilde{u} \in \Pi_{yc}^{++}$ can be viewed as a word in the alphabet $A = \{a, b, c, x, y\}$. For an integer $n \geq 0$, denote by y^n the horizontal subpath of P_y^+ beginning at $\widetilde{v}$ and having length n. Analogously, for an integer $m \geq 0$, denote by c^m the vertical subpath of P_c^+ beginning at $\widetilde{v}$ and having length m. Let $M_n(m)$ denote the horizontal path of Π_{yc}^{++} of length n beginning at the endpoint of the vertical path c^m. $M_n(m)$ determines a word which is the label of the side opposite to y^n in the rectangle which is the convex hull of y^n and c^m (see Figure 2.6). Let $M_n(m)$ also denote this corresponding word.

PROPOSITION 2.41 ([**257**, Proposition 5.9]). *For each n, the words $\{M_n(m) : 0 \leq m \leq 2^n - 1\}$ are all distinct, and thus, every positive word in x and y of length n is $M_n(m)$ for some m.*

This proposition is called in [**257**] "period doubling". It immediately establishes Theorem 2.40 because it shows that the period of the infinite vertical strip of Π_{yc}^{++} of width n and bounded on the left by the path P_c^+ has period 2^n. Alternatively, every positive word in x and y appears in Π_{yc}^{++}, and thus Π_{yc} cannot be periodic.

6.3. The Square Complex W and its Universal Cover $\widetilde{W}$. Let $\beta\mathbf{X}$ denote the first barycentric subdivision of $\mathbf{X}$: each square C of $\mathbf{X}$ is subdivided into four squares C_1, C_2, C_3, C_4 by adding a middle vertex to each edge of C and connecting it to the center of C by an edge. This way each edge e of C is subdivided into two edges e_1, e_2, which inherit the orientation and the color of e. The four edges connecting the middle vertices of the edges of C to the center of C are oriented from left to right and from

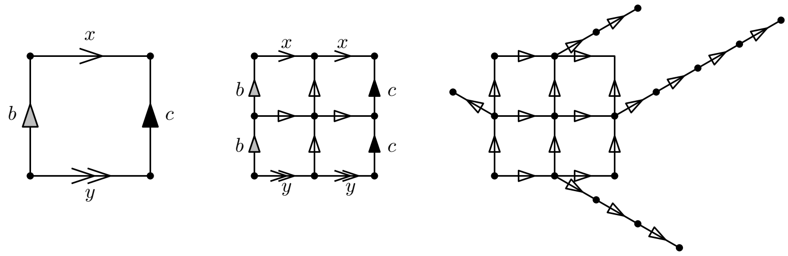


FIGURE 2.7. A square of $\mathbf{X}$ and the corresponding subcomplexes in $(\beta\mathbf{X}, o')$ and (W, o^*)

bottom to top (see the middle figure of Figure 2.7). Denote the resulting orientation by o'. This way, $(\beta\mathbf{X}, o')$ is a directed and colored square complex. Again, denote by βX the support of $\beta\mathbf{X}$. The universal cover $\widetilde{\beta X}$ of βX is the Cartesian product $\beta F_4 \times \beta F_6$ of the trees βF_4 and βF_6, where βF_4 is the first barycentric subdivision of F_4 and βF_6 is the first barycentric subdivision of F_6. Additionally, $(\widetilde{\beta X}, \widetilde{o}')$ is a directed CAT(0) square complex. We assign a *type* to each vertex of $\widetilde{\beta\mathbf{X}}$: the preimage of the unique vertex of $\mathbf{X}$ is of type 0 and is called a 0-*vertex*, the preimages of the middles of edges of $\mathbf{X}$ are of type 1 and are called 1-*vertices*, and the preimages of centers of squares of $\mathbf{X}$ are of type 2 and are called 2-*vertices*.

To encode the colors of the edges of $\mathbf{X}$, we introduce our central object, the square complex W (whose edges are no longer colored). Let $A = \{a, b, c, x, y\}$ and let $r : A \to \{1, 2, 3, 4, 5\}$ be a bijective map. The complex W is obtained from βX by adding to each 1-vertex z of βX a path R_z of length $r(\alpha)$ if z is the middle of an edge colored $\alpha \in A$ in $\mathbf{X}$. The path R_z has one end at z (called the *root* of R_z) and z is the unique common vertex of R_z and βX (we call such added paths R_z *tips*).

The square complex W has 27 vertices: the unique vertex of $\mathbf{X}$, the 6 vertices which are the barycenters of the original squares, 5 vertices which are the barycenters of the original edges of $\mathbf{X}$, and 15 vertices which are new vertices lying on tips. The complex W has 49 edges: 10 corresponding to the 5 original edges that have been subdivided, 24 connecting the barycenters of the original squares to the barycenters of the original edges and 15 forming the tips. The complex W has 24 squares: 4 for each original square.

Denote by o^* the orientation of the edges of W defined as follows: the edges of βX are oriented as in $(\beta X, o')$ and the edges of tips are oriented away from their roots (see the rightmost figure of Figure 2.7 for the encoding of the last square of Figure 2.5). As a result, we obtain a finite directed NPC square complex (W, o^*).

Consider the universal cover $\widetilde{W}$ of W. It can be viewed as the complex $\widetilde{\beta X}$ with a path of length $r(\alpha)$ added to each 1-vertex which encodes an edge of $\widetilde{\mathbf{X}}$ of color $\alpha \in A$. We say that the vertices of $\widetilde{W}$ lying only on tips are of type 3 and they are called 3-*vertices*. Let $\widetilde{o}^*$ denote the orientation of the edges of $\widetilde{W}$ induced by the orientation o^* of W. Then $(\widetilde{W}, \widetilde{o}^*)$ is a directed CAT(0) square complex. Since W is finite, by Proposition 2.21, the directed median graph $(\widetilde{W}^{(1)}, \widetilde{o}^*)$ has a finite number of isomorphisms types of principal filters $\mathcal{F}_{\widetilde{o}^*}(\widetilde{z}, \widetilde{W}^{(1)})$.

Let $\widetilde{v}$ be any 0-vertex of $\widetilde{W}$. Denote by $\widetilde{W}_{\widetilde{v}}$ the principal filter $\mathcal{F}_{\widetilde{o}^*}(\widetilde{v}, \widetilde{W}^{(1)})$ of $\widetilde{v}$ in $(\widetilde{W}^{(1)}, \prec_{\widetilde{o}^*})$. By Proposition 2.21, $\widetilde{W}_{\widetilde{v}}$ together with the partial order $\prec_{\widetilde{o}^*}$ is the domain of a regular event structure, which we call *Wise's event domain.* Since vertices of different types of $\widetilde{W}$ are incident to a different number of outgoing squares, any isomorphism between two filters of $(\widetilde{W}_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ preserves the types of vertices. We summarize all this in the following:

PROPOSITION 2.42. *$(\widetilde{W}_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ is the domain of a regular event structure. Any isomorphism between any two filters of $(\widetilde{W}_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ preserves the types of vertices.*

6.4. $(\widetilde{W}_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ does not have a Regular Nice Labeling. In this subsection we prove that the event structure associated with Wise's regular event domain is a counterexample to Thiagarajan's Conjecture 2.3.

THEOREM 2.43. *$(\widetilde{W}_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ does not admit a regular nice labeling. Consequently, Conjecture 2.3 is false.*

PROOF. Since $\widetilde{W}_{\widetilde{v}}$ is the principal filter of a 0-vertex $\widetilde{v}$, $\widetilde{W}_{\widetilde{v}}$ contains all vertices of $\widetilde{\mathbf{X}}$ located in the quarter of plane Π^{++}_{yc} of $\widetilde{\mathbf{X}}$, in particular it contains the vertices of the paths P^+_c and P^+_y. Notice also that $\widetilde{W}_{\widetilde{v}}$ contains the barycenters and the tips corresponding to the edges of Π^{++}_{yc}.

Suppose by way of contradiction that $\widetilde{W}_{\widetilde{v}}$ has a regular nice labeling λ. Since $\widetilde{W}_{\widetilde{v}}$ has only a finite number of isomorphism types of labeled filters, the vertical path P^+_c contains two 0-vertices, $\widetilde{z}'$ and $\widetilde{z}''$, which have isomorphic labeled principal filters. Let $\widetilde{z}'$ be the end of the vertical subpath c^k of P^+_c and $\widetilde{z}''$ be the end of the vertical subpath c^m of P^+_c, and suppose without loss of generality that $k < m$. Let $n > 0$ be a positive integer such that $m \leq 2^n - 1$. Consider the horizontal convex paths $M_n(k)$ and $M_n(m)$ of Π^{++}_{yc} of length n beginning at the vertices $\widetilde{z}'$ and $\widetilde{z}''$, respectively. For any $0 \leq i \leq n$, denote by $\widetilde{z}_{k,i}$ the ith vertex of $M_n(k)$ (in particular, $\widetilde{z}_{k,0} = \widetilde{z}'$). Analogously, denote by $\widetilde{z}_{m,i}$ the ith vertex of $M_n(m)$ (in particular, $\widetilde{z}_{m,0} = \widetilde{z}''$). In $\widetilde{W}_{\widetilde{v}}$, the paths $M_n(k)$ and $M_n(m)$ give rise to two convex horizontal paths $M^*_n(k)$ and $M^*_n(m)$ obtained from $M_n(k)$ and $M_n(m)$ by subdividing their edges. Denote by $\widetilde{u}_{k,i}$ the unique common neighbor of $\widetilde{z}_{k,i}$ and $\widetilde{z}_{k,i+1}$, $0 \leq i < n$, in $M^*_n(k)$ (and in $\widetilde{W}^{(1)}$). Analogously, denote by $\widetilde{u}_{m,i}$ the unique common neighbor of $\widetilde{z}_{m,i}$ and $\widetilde{z}_{m,i+1}$, $0 \leq i < n$ (see Figure 2.8). The paths $M^*_n(k)$ and $M^*_n(m)$ belong to the principal filters $\mathcal{F}_{\widetilde{o}^*}(\widetilde{z}', \widetilde{W}^{(1)})$ and $\mathcal{F}_{\widetilde{o}^*}(\widetilde{z}'', \widetilde{W}^{(1)})$, respectively.

By Proposition 2.41, the words $M_n(k)$ and $M_n(m)$ are different. Let f be an isomorphism between the filters $\mathcal{F}_{\widetilde{o}^*}(\widetilde{z}_{k,0}, \widetilde{W}^{(1)})$ and $\mathcal{F}_{\widetilde{o}^*}(\widetilde{z}_{m,0}, \widetilde{W}^{(1)})$. Since the words $M_n(k)$ and $M_n(m)$ are different, from the choice of the lengths of tips in the complexes W and $\widetilde{W}$ it follows that f cannot map the path $M^*_n(k)$ to the path $M^*_n(m)$ by a vertical translation, i.e., there exists an index $0 \leq j < n$ such that $f(\widetilde{z}_{k,j+1}) \neq \widetilde{z}_{m,j+1}$; let i be the smallest such index. Set $\widetilde{z} := f(\widetilde{z}_{k,i+1})$ and $\widetilde{u} := f(\widetilde{u}_{k,i})$. Since f preserves the types of vertices, $\widetilde{z}$ is a 0-vertex and $\widetilde{u}$ is a 1-vertex. Since f maps a convex path $M^*_n(k)$ to a convex path, $\widetilde{u}$ is the unique common neighbor of $\widetilde{z}_{m,i}$ and $\widetilde{z}$. Since each 1-vertex is the barycenter of a unique edge of $\widetilde{\mathbf{X}}$ and $\widetilde{z} \neq \widetilde{z}_{m,i+1}$, we deduce that $\widetilde{u} \neq \widetilde{u}_{m,i}$. The edge $\widetilde{z}_{k,i}\widetilde{u}_{k,i}$ is directed from $\widetilde{z}_{k,i}$ to $\widetilde{u}_{k,i}$. Analogously the edges $\widetilde{z}_{m,i}\widetilde{u}_{m,i}$ and $\widetilde{z}_{m,i}\widetilde{u}$ are directed from $\widetilde{z}_{m,i}$ to $\widetilde{u}_{m,i}$ and $\widetilde{u}$, respectively. Since $\widetilde{z}_{k,i}\widetilde{u}_{k,i}$ and $\widetilde{z}_{m,i}\widetilde{u}_{m,i}$ are parallel edges, they define the same event and therefore $\lambda(\widetilde{z}_{k,i}\widetilde{u}_{k,i}) = \lambda(\widetilde{z}_{m,i}\widetilde{u}_{m,i})$. On the other hand, since f maps the edge $\widetilde{z}_{k,i}\widetilde{u}_{k,i}$ to the edge $\widetilde{z}_{m,i}\widetilde{u}$ and since the map f preserves the labels, we have $\lambda(\widetilde{z}_{k,i}\widetilde{u}_{k,i}) = \lambda(\widetilde{z}_{m,i}\widetilde{u})$. As a result, $\widetilde{z}_{m,i}$ has two outgoing edges, $\widetilde{z}_{m,i}\widetilde{u}_{m,i}$ and $\widetilde{z}_{m,i}\widetilde{u}$, having the same label, contrary to the assumption that λ is a nice labeling. This contradiction shows that $(\widetilde{W}_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ does not admit a regular nice labeling. By Proposition 2.21, $(\widetilde{W}_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ is the domain of a regular event structure, establishing that Conjecture 2.3 is false. This concludes the proof of the theorem. □

We can show that our counterexample has $\natural$-cliques of size at most 11 and thus it disproves Conjecture 2.4 of Badouel et al.

PROPOSITION 2.44. *Wise's event domain $(\widetilde{W}_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ has bounded $\natural$-cliques. Consequently, $(\widetilde{W}_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ is a counterexample to Conjectures 2.4.*

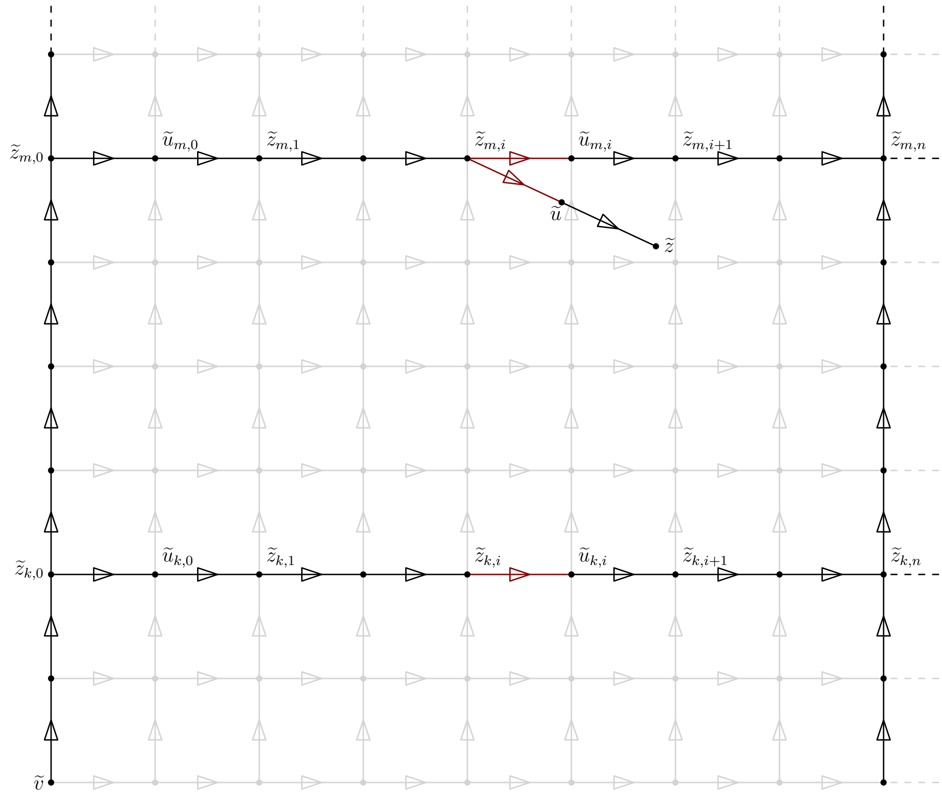


FIGURE 2.8. To the proof of Theorem 2.43

6.5. Other Counterexamples Arising from Aperiodic Tilings. Our counterexample $(\widetilde{W}_{\widetilde{v}}, \prec_{\widetilde{\partial}^*})$ of a regular 2-dimensional event domain without a regular labeling heavily uses the fact that the universal cover $\widetilde{\mathbf{X}}$ of Wise's complex $\mathbf{X}$ [**257**] contains a particular aperiodic tiled plane (that is called *antitorus* by Wise). In this subsection, we show that the relationship between the existence of aperiodic planes and nonexistence of regular labelings is more general. Namely, we explain how to obtain other counterexamples from 4-way deterministic aperiodic tile sets.

Tiles (or *Wang-tiles*) are unit squares with colored edges. The edges of a Wang tile are called *top* (or *North*), *right* (or *East*), *bottom* (or *South*) and *left* (or *West*) edges in a natural way. A *tile set* T is a finite collection of Wang-tiles, placed with their edges horizontal and vertical. A *tiling* is a mapping $f : \mathbb{Z}^2 \to T$ that assigns a tile to each integer lattice point of the plane. A tiling f is *valid* if every two adjacent tiles have the same color on their common edge. Note that a tile may not be rotated or flipped, i.e., each tile has a bottom-top and left-right orientation. A tiling f is *periodic* with period $(a, b) \in \mathbb{Z}^2 \setminus \{(0, 0)\}$ if for every $(x, y) \in \mathbb{Z}^2$, $f(x, y) = f(x + a, y + b)$. If there exists a valid periodic tiling with tiles of T, then there exists a valid *doubly periodic* tiling with tiles of T [**213**], i.e., a valid tiling f and two integers $a, b > 0$ such that $f(x, y) = f(x + a, y) = f(x, y + b)$ for every $(x, y) \in \mathbb{Z}^2$. A tile set T is called *aperiodic* if there exists a valid tiling with tiles of T, and there does not exist any periodic valid tiling with tiles of T.

Let $T = \{t_1, \ldots, t_n\}$ be a tile set. We consider each tile t_i as a unit square whose edges are directed and colored. Suppose that each square t_i has two vertical and two horizontal edges and suppose that the horizontal and the vertical edges of all squares are colored differently, i.e., the set of colors can be partitioned into horizontal colors and vertical colors. The horizontal edges are directed from left to right and the vertical edges are directed from bottom to top.

A Wang tile set is said to be *NW-deterministic* [**147**], if within the tile set there does not exist two different tiles that have the same colors on their top and left edges. *NE-deterministic*, *SW-deterministic*, and *SE-deterministic* tile sets are defined analogously. A Wang tile set is *4-way deterministic* [**147**] if it is NW-, NE-, SW-, and SE-deterministic. Kari and Papasoglu [**147**] presented a 4-way deterministic aperiodic tile set T_{KP}.

Given a 4-way deterministic set of tiles T, let $\mathbf{X}(T) = (X(T), o, \nu)$ be the finite square complex obtained by identifying all the vertices and gluing together the squares of T along the sides which have the same color respecting their orientation. Then $\mathbf{X}(T)$ is a VH-complex that has a unique vertex. Consequently, the universal cover $\widetilde{\mathbf{X}}(T)$ of $\mathbf{X}(T)$ is a CAT(0) VH-complex. Denote by $W(T)$ the finite directed NPC complex derived from $\mathbf{X}(T)$ in the same way as the complex W was derived from Wise's complex $\mathbf{X}$ in Subsection 6.3 (taking the first barycentric subdivision and adding tips of different lengths to encode the different colors). Let $(\widetilde{W}(T)_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ denote the 2-dimensional event domain derived from $\widetilde{\mathbf{X}}(T)$ in the same way as $(\widetilde{W}_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ was derived from $\widetilde{\mathbf{X}}$. Since $(\widetilde{W}(T)_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ comes from the universal cover of the finite directed NPC complex $W(T)$, $(\widetilde{W}(T)_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ is a strongly regular event structure. The following lemma establishes a connection between the existence of valid tilings for 4-way deterministic tile sets and the existence of directed planes in the universal covers of the derived VH-complexes.

LEMMA 2.45. *For a 4-way deterministic tile set T, the following conditions are equivalent:*

(i) there exists a valid tiling with the tiles of T;
(ii) the universal cover $\widetilde{\mathbf{X}}(T)$ of the square complex $\mathbf{X}(T)$ contains directed planes;
(iii) the strongly regular domain $(\widetilde{W}(T)_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ is not hyperbolic.

Note that if T is a 4-way deterministic aperiodic tile set, all the directed planes of $\widetilde{\mathbf{X}}(T)$ are tiled in an aperiodic way. In the case of the tile set of Wise [**257**] from Figure 2.5, the CAT(0) square complex $\widetilde{\mathbf{X}}$ contains aperiodic directed planes but it also contains some periodic directed planes.

We now explain how to derive a counterexample to Thiagarajan's conjectures from any 4-way deterministic aperiodic tile set.

We show that if the 2-dimensional event domain $(\widetilde{W}(T)_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ associated to a 4-way deterministic tile set T admits a regular nice labeling, then there exists a periodic tiling of the plane with tiles of T, establishing the following theorem.

THEOREM 2.46. *For any 4-way deterministic aperiodic tile set T, the NPC square complex $W(T)$ is not virtually special and the 2-dimensional event domain $(\widetilde{W}(T)_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ does not admit a regular nice labeling.*

Consequently, $(\widetilde{W}(T)_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ is a counterexample to Thiagarajan's Conjecture 2.3.

Using the tile set T_{KP} of [**147**], Lukkarila [**164**] proved that for 4-way deterministic tile sets the tiling problem is undecidable. An immediate consequence of this result and of Theorem 2.46 is that there exists an infinite number of counterexamples to Conjecture 2.3.

REMARK 2.47. Note that the VH-complex $W(T)$ derived from a 4-way deterministic tile set T is not necessarily a CSC complex. The proof of Proposition 2.44 can be extended to all CSC complexes, but we do not know if it holds for all VH-complexes. Consequently, we cannot directly generalize the proof of Proposition 2.44 to show that if T is aperiodic, then $(\widetilde{W}(T)_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ is a counterexample to Conjecture 2.4.

7. On the Decidability of the MSO Theory of Net Systems and of their Domains

7.1. The Results. Let $\mathcal{E} = (E, \leq, \#, \lambda)$ be a trace-regular event structure and let $\mathcal{D}(\mathcal{E})$ denote the domain of $\mathcal{E}$. Let $G(\mathcal{E})$ denote the undirected covering median graph of $\mathcal{D}(\mathcal{E})$ and $\overrightarrow{G}(\mathcal{E}) = (G(\mathcal{E}), o)$ denote the directed graph of $\mathcal{D}(\mathcal{E})$. We restate the main theorem of this section that characterize the trace event structures $\mathcal{E}$ for which the MSO theories of their graphs $G(\mathcal{E})$ and $\overrightarrow{G}(\mathcal{E})$ are decidable.

THEOREM 2.12. *For a trace-regular event structure $\mathcal{E} = (E, \leq, \#, \lambda)$, the following conditions are equivalent:*

(1) $\mathrm{MSO}(\overrightarrow{G}(\mathcal{E}))$ *is decidable;*
(2) $\mathrm{MSO}_1(G(\mathcal{E}))$ *is decidable;*
(3) $\mathrm{MSO}_2(G(\mathcal{E}))$ *is decidable:*
(4) $G(\mathcal{E})$ *has finite treewidth;*
(5) the clusters of $G(\mathcal{E})$ *have bounded diameter;*
(6) $\overrightarrow{G}(\mathcal{E})$ *is context-free.*

Similarly to a question about the decidability of the MSO theory of graphs of domains of trace-regular event structures (i.e., of domains of event structure unfoldings of net systems), one can ask a similar question about the decidability of the MSO theory for the graphs (1-skeletons) of the universal covers of the special cube complexes X_N of net systems N. In this case, the following result holds:

PROPOSITION 2.48. *Let $N = (S, \Sigma, F, m_0)$ be a net system, X_N be the special cube complex of N, and let $\overrightarrow{G}(\widetilde{\mathbf{X}}_N)$ be the 1-skeleton of the directed labeled universal cover of X_N. Then the following conditions are equivalent:*

(1) $\mathrm{MSO}(\overrightarrow{G}(\widetilde{\mathbf{X}}_N))$ *is decidable;*
(2) $\mathrm{MSO}_2(G(\widetilde{\mathbf{X}}_N))$ *is decidable:*
(3) $G(\widetilde{\mathbf{X}}_N)$ *has finite treewidth;*
(4) $\overrightarrow{G}(\widetilde{\mathbf{X}}_N)$ *is context-free.*

The proof of this result essentially follows from the result by Kuske and Lohrey [**154**] that the decidability of the MSO theory of a directed graph $\overrightarrow{G}$ of bounded degree whose automorphism group $\mathrm{Aut}(\overrightarrow{G})$ has only finitely many orbits on $\overrightarrow{G}$ is equivalent to the fact that $\overrightarrow{G}$ is context-free and to the fact that its undirected support has finite treewidth. This result cannot be applied to prove Theorem 2.12 because $\mathrm{Aut}(\overrightarrow{G}(\mathcal{E}))$ may have an infinite number of orbits.

To relate the MSO theory of the graph of the domain of a trace event structure with the MSO theory of the event structure, we introduce the notion of the *hairing* $\dot{\mathcal{E}} = (\dot{E}, \dot{\leq}, \dot{\#})$ of an event structure $\mathcal{E} = (E, \leq, \#)$. To obtain $\dot{\mathcal{E}}$, we add a *hair event* e_c for each configuration c of $\mathcal{E}$, i.e., $\dot{E} = E \cup E_C$ where $E_C = \{e_c : c \in \mathcal{D}(\mathcal{E})\}$. For any hair event e_c and any event $e \in \dot{E}$, we set $e \dot{\leq} e_c$ if $e \in c$ and $e \dot{\#} e_c$ otherwise. Suppose additionally that $\mathcal{E}$ is trace-regular and let λ be a trace labeling of $\mathcal{E}$ with a trace alphabet $M = (\Sigma, I)$. Let h be a letter that does not belong to Σ and consider the trace alphabet $\dot{M} = (\Sigma \cup \{h\}, I)$ (note that since I is not modified, $(h, a) \notin I$ for every $a \in \Sigma$). Let $\dot{\lambda}$ be the labeling of $\dot{\mathcal{E}}$ extending λ by setting $\dot{\lambda}(e_c) = h$ for any $e_c \in E_C$. The labeled event structure obtained in this way is trace-regular:

PROPOSITION 2.49. *For a trace-regular event structure $\mathcal{E} = (E, \leq, \#, \lambda)$, the hairing $\dot{\mathcal{E}} = (\dot{E}, \dot{\leq}, \dot{\#}, \dot{\lambda})$ is also a trace-regular event structure.*

By the definition of $\dot{\mathcal{E}}$, the directed graph $\overrightarrow{G}(\dot{\mathcal{E}})$ of its domain $\mathcal{D}(\dot{\mathcal{E}})$ is obtained from the directed graph $\overrightarrow{G}(\mathcal{E})$ of the domain $\mathcal{D}(\mathcal{E})$ of $\mathcal{E}$ by adding an outgoing arc $\overrightarrow{vw_v}$ to each vertex v of $\overrightarrow{G}(\mathcal{E})$. In a similar way, we can define the *hairing* $\dot{G}$ (respectively $\dot{X}$) of any directed graph G (respectively, any directed NPC complex X) by adding for each vertex v, a new vertex v' and an arc $\overrightarrow{vv'}$. Observe that each new vertex v' has in-degree 1 and out-degree 0. With this definition, the hairing $\dot{\overrightarrow{G}}(\mathcal{E})$ of the directed graph $\overrightarrow{G}(\dot{\mathcal{E}})$ of the domain $\mathcal{D}(\dot{\mathcal{E}})$ of $\mathcal{E}$ coincides with the directed graph $\overrightarrow{G}(\dot{\mathcal{E}})$ of the domain $\mathcal{D}(\dot{\mathcal{E}})$ of the hairing $\dot{\mathcal{E}}$ of $\mathcal{E}$. Given a poset $\mathcal{D} = (D, \prec)$, we define the *hairing* of $\mathcal{D}$ as the poset $\dot{\mathcal{D}} = (\dot{D}, \dot{\prec})$ such that the Hasse diagram of $\dot{\mathcal{D}}$ is the hairing of the Hasse diagram of $\mathcal{D}$. With this definition, the domain $\mathcal{D}(\dot{\mathcal{E}})$ of $\mathcal{E}$ coincides with the domain $\mathcal{D}(\dot{\mathcal{E}})$ of the hairing $\dot{\mathcal{E}}$ of $\mathcal{E}$.

Note that the hairing $\dot{\widetilde{X}}$ of the universal cover $\widetilde{X}$ of a directed NPC complex X coincides with the universal cover $\widetilde{\dot{X}}$ of the hairing $\dot{X}$ of X. When an event structure $\mathcal{E}$ is strongly regular, there exists a finite directed NPC complex X such that $\mathcal{D}(\mathcal{E}) = \mathcal{F}_{\widetilde{o}}(\widetilde{v}, \widetilde{X}^{(1)})$. In this case, we have:

$$\mathcal{D}(\dot{\mathcal{E}}) = \dot{\mathcal{D}}(\mathcal{E}) = \dot{\mathcal{D}}_{\widetilde{o}}(\widetilde{v}, \widetilde{X}^{(1)}) = \mathcal{F}_{\widetilde{o}}(\widetilde{v}, \dot{\widetilde{X}}^{(1)}) = \mathcal{F}_{\widetilde{o}}(\widetilde{v}, \widetilde{\dot{X}}^{(1)}). \tag{2.1}$$

We can also define the *hairing* $\dot{N} = (\dot{S}, \dot{\Sigma}, \dot{F}, \dot{m}_0)$ of a net system $N = (S, \Sigma, F, m_0)$ as follows. First, for each transition $a \in \Sigma$, we add a place p_a such that ${}^\bullet p_a = p_a^\bullet = \{a\}$ and such that p_a contains a token in the initial configuration. Then, we add a transition h such that ${}^\bullet h = \{p_a : a \in \Sigma\}$ and $h^\bullet = \varnothing$. In other words, $\dot{S} = S \cup \{p_a : a \in \Sigma\}$, $\dot{\Sigma} = \Sigma \cup \{h\}$, $\dot{F} = F \cup \{(p_a, a), (a, p_a), (p_a, h) : a \in \Sigma\}$, and $\dot{m}_0 = m_0 \cup \{p_a : a \in \Sigma\}$.

PROPOSITION 2.50. *For a net system N, there is an isomorphism between the special cube complex $X_{\dot{N}}$ of the hairing $\dot{N}$ of N and the hairing $\dot{X}_N$ of the special cube complex X_N of N that maps the initial marking m_0 of N to the initial marking $\dot{m}_0$ of $\dot{N}$. Consequently, the event structure unfolding $\mathcal{E}_{\dot{N}}$ of the hairing $\dot{N}$ of N is isomorphic to the hairing $\dot{\mathcal{E}}_N$ of the event structure unfolding $\mathcal{E}_N$ of N.*

Note that Proposition 2.49 follows from Proposition 2.50 and Thiagarajan's Theorem 2.16. However, there is a simpler proof of this result that does not rely on the involved construction of a net system from a trace-regular event structure used in the proof of Theorem 2.16.

Notice that the hair events of $\dot{\mathcal{E}}$ introduce a lot of conflicting events in $\dot{\mathcal{E}}$, and we use them to encode vertex variables as event variables in order to prove the following result:

THEOREM 2.51. *For a trace-regular event structure $\mathcal{E} = (E, \leq, \#, \lambda)$, $\mathrm{MSO}(\dot{\mathcal{E}})$ is decidable if and only if $\mathrm{MSO}(\overrightarrow{G}(\mathcal{E}))$ is decidable. In particular, $\mathrm{MSO}(\dot{\mathcal{E}})$ is decidable if and only if $G(\mathcal{E})$ has finite treewidth.*

Since $\mathrm{MSO}(\mathcal{E})$ is a fragment of $\mathrm{MSO}(\dot{\mathcal{E}})$, we obtain the following corollary of Theorem 2.51:

COROLLARY 2.52. *For any trace-regular event structure $\mathcal{E} = (E, \leq, \#, \lambda)$, if $G(\mathcal{E})$ has finite treewidth, then $\mathrm{MSO}(\mathcal{E})$ is decidable.*

7.2. Treewidth and Context-free Graphs. Let $G = (V, E)$ be a simple graph, not necessarily finite. A *tree decomposition* [**211**] of G is a pair (T, f), where $T = (V(T), E(T))$ is a tree and $f : V(T) \to 2^V \setminus \{\varnothing\}$ is a function such that the following holds:

(i) $\bigcup_{t \in V(T)} f(t) = V$,
(ii) for every edge $uv \in E$ of G there exists $t \in V(T)$ such that $u, v \in f(t)$,
(iii) if $t', t'' \in V(T)$ and t lies on the unique path of T from t' to t'', then $f(t') \cap f(t'') \subseteq f(t)$.

The supremum in $\mathbb{N} \cup \{\infty\}$ of the cardinalities $|f(t)|, t \in V(T)$, is called the *width* of the tree decomposition (T, f). The graph G has *treewidth* $\leq b$ if there exists a tree decomposition of G of width $\leq b$. A graph G has *bounded* (or *finite*) *treewidth* if it has treewidth $\leq b$ for some $b \in \mathbb{N}$. The treewidth represents how close a graph is to a tree from a *combinatorial* point of view.

A graph H is a *minor* of a graph G if a graph isomorphic to H can be obtained from a subgraph G' of G by contracting edges. Equivalently, H is a minor of a connected graph G if G contains a subgraph G' such that there exists a partition of vertices of G' into connected subgraphs $\mathcal{P} = \{P_1, \ldots, P_t\}$ and a bijection $f : V(H) \to \mathcal{P}$ such that if $uv \in E(H)$ then there exists an edge of G' running between the subgraphs $f(u)$ and $f(v)$ of $\mathcal{P}$ (i.e., after contracting each subgraph $P_i \in \mathcal{P}$ into a single vertex we will obtain a graph containing H as a spanning subgraph). Treewidth does not increase when taking a minor.

Since the treewidth of an $n \times n$ square grid is n, the treewidth of a graph G is always greater than or equal to the size of the largest square grid minor of G. In the other direction, the grid minor theorem by Robertson and Seymour [**212**] shows that there exists a function f such that the treewidth is at most $f(r)$ where r is the size of the largest square grid minor of G:

THEOREM 2.53 ([**212**]). *A graph G has bounded treewidth if and only if the square grid minors of G have bounded size.*

Let G be an edge-labeled graph of uniformly bounded degree and v_0 be an arbitrary root (basepoint) of G. Let $S(v_0, k) = \{x \in V : d_G(v_0, x) = k\}$ denote the sphere of radius k centered at v_0. A connected component Υ of the subgraph of G induced by $V \setminus S(v_0, k)$ is called an *end* of G. The vertices of $\Upsilon \cap S(v_0, k+1)$ are called *frontier points* and this set is denoted by $C(\Upsilon)$ [**178**] and called a *cluster*. There exists a bijection between the ends and the clusters: each end contains a unique cluster and conversely, for a cluster C, the unique end $\Upsilon(C)$ containing C consists of the union of all principal filters of the vertices $v \in C$ (with respect to the basepoint order).

Let $\Phi(G)$ and $\mathcal{C}(G)$ denote the set of all ends and all clusters of G, respectively. An *end-isomorphism* between two ends Υ and Υ' of G is a label-preserving mapping f between Υ and Υ' such that f is a graph isomorphism and f maps $C(\Upsilon)$ to $C(\Upsilon')$. Then G is called a *context-free graph* [**178**] if $\Phi(G)$ has only finitely many isomorphism classes under end-isomorphisms. Since G has uniformly bounded degree, each cluster $C(\Upsilon)$ is finite. Moreover, from the definition of context-free graphs follows that a context-free graph G has only finitely many isomorphism classes of clusters, thus there exists a constant $\delta < \infty$ such that the diameter of any cluster of G is bounded by δ. By [**84**, Proposition 12] any graph G whose diameters of clusters is uniformly bounded by δ is δ-hyperbolic (in fact, G is quasi-isometric to a tree). Note that the converse is not true (see the 1-skeleton of the square complex $\widetilde{Z}$ described in Section 8).

7.3. Some Results from MSO Theory. In this subsection, we recall some results from MSO theory of undirected graphs, directed labeled graphs, latices and posets, and event structures. These results either will be used below or are related to our work.

Among the MSO theories of various discrete structures, the MSO theory of undirected graphs is probably the most complete, with various and deep applications. For a comprehensive account of this theory, see the book by Courcelle and Engelfriet [**97**]. Let $G = (V, E)$ be an undirected and unlabeled graph. The MSO logic as introduced in Subsection 1.5 only allow quantifications over subsets of vertices of G. This theory is usually denoted by $\mathrm{MSO}_1(G)$. In order to allow also quantifications over subsets of edges, an extended representation of a graph is used. This is the relational structure $G^e = (V \cup E, \mathrm{inc})$, where

$$\mathrm{inc} = \{(e, v) \in E \times V : \exists u \in V \text{ such that } e \in \{uv, vu\}\}.$$

The MSO theory of this relational structure G^e is usually denoted by $\mathrm{MSO}_2(G)$. Seese [**225**] proved the following fundamental result about MSO_2 decidability:

THEOREM 2.54 ([**225**]). *If* $\mathrm{MSO}_2(G)$ *is decidable, then* G *has finite treewidth.*

The converse of Seese's theorem is not true: one can construct trees with undecidable MSO_2 theory. On the other, Courcelle [**95**] proved that for any natural integer k the class of all graphs of treewidth at most k has a decidable MSO_2 theory.

If $\mathrm{MSO}_2(G)$ is decidable, then $\mathrm{MSO}_1(G)$ is also decidable. Again, the reverse implication is not true. However, Courcelle [**96**] proved that the converse holds for graphs with bounded degrees:

THEOREM 2.55 ([**96**]). *If* G *is a graph with uniformly bounded degrees and* $\mathrm{MSO}_1(G)$ *is decidable, then* $\mathrm{MSO}_2(G)$ *is also decidable.*

Now, consider labeled directed graphs. Let Σ be a finite alphabet. A Σ-*labeled directed graph* is a relational structure $\overrightarrow{G} = (V, (E_a)_{a\in\Sigma})$, where V is the set of vertices and $E_a \subseteq V \times V$ is the set of a-labeled directed edges. Denote by $\mathrm{MSO}(\overrightarrow{G})$ the MSO theory of this relational structure. We associate to $\overrightarrow{G}$ the unlabeled graph $G = (V, \bigcup_{a\in\Sigma}\{uv : u \neq v, (u,v) \in E_a \text{ or } (v,u) \in E_a\})$.

Müller and Schupp [**178**] proved the following fundamental theorem about Σ-labeled pointed context-free graphs of bounded degree (and directed according to the basepoint order):

THEOREM 2.56 ([**178**]). *If* $\overrightarrow{G}$ *is a context-free graph, then* $\mathrm{MSO}(\overrightarrow{G})$ *is decidable.*

7.4. Grids. In this section we need to consider several types of square grids, which characterize different properties of event structures and their graphs. In this subsection, we will introduce some notational order between these notions and relate some of them. Recall that the infinite *square grid* Γ is the graph whose vertices correspond to the points in the plane with nonnegative integer coordinates and two vertices are connected by an edge whenever the corresponding points are at distance 1. The $n \times n$ square grid Γ_n is the subgraph of Γ whose vertices are all vertices of Γ with x- and y-coordinates in the range $0, \ldots, n$. Γ and Γ_n can be viewed as directed graphs with respect to the basepoint order with respect to the corner $(0,0)$. As we noticed above, Γ is the domain of the event structure consisting of two pairwise disjoint sets $X = \{x_0, x_1, x_2, \ldots\}, Y = \{y_0, y_1, y_2, \ldots\}$ of events, such that $x_0 < x_1 < x_2 < \cdots$ and $y_0 < y_1 < y_2 < \cdots$, and all events of X are concurrent with all events of Y. This event structure is conflict-free and trace-regular. Below, if not specified, by Λ we denote either of the grids Γ or Γ_n. A *directed grid* $\overrightarrow{\Lambda}$ is a grid Λ with basepoint orientation with respect to the origin $(0,0)$.

By Theorem 2.53, the treewidth of a graph is characterized by square grid minors. We will say that a square grid Λ is a *grid minor* of a graph G if Λ is a minor of G.

By Lemma 2.27, the hyperbolicity of a median graph (event domain or 1-skeleton of a CAT(0) cube complex) is characterized by isometrically embedded square grids. We will say that a square grid Λ is an *isometric grid* of a median graph $G = (V, E)$ if there exists an isometric embedding of Λ in G, i.e., a map $f : V(\Lambda) \to V$ such that $d_\Lambda(x, y) = d_G(x, y)$ for any two vertices $x, y \in V(\Lambda)$. An event structure characterization of isometric grids is provided below.

A stronger version of isometric grid is the notion of a flat grid. We will say that an isometric grid Λ is a *flat grid* of a median graph G if for any two vertices x, y of Λ at distance 2, any common neighbor z of x and y in G belongs to the grid Λ. Since any locally-convex connected subgraph of G is convex, any flat grid is a convex (and thus gated) subgraph of G. If G is the 1-skeleton of a 2-dimensional cube complex, then any isometric grid is flat. If Λ is a flat grid of the graph $G(\mathcal{E})$ of an event domain $\mathcal{D}(\mathcal{E})$, then there are two disjoint subsets $X = \{x_0, x_1, x_2, \ldots\}, Y = \{y_0, y_1, y_2, \ldots\}$ of events of

$\mathcal{E}$ such that $x_0 \lessdot x_1 \lessdot x_2 \lessdot \cdots$ and $y_0 \lessdot y_1 \lessdot y_2 \lessdot \cdots$, and all events of X are concurrent with all events of Y.

The minor of a graph is defined by contracting edges. Minors are also implicitly used in the theory of event structures, namely, when the event structure $\mathcal{E}\backslash c$ rooted at a configuration c was defined (this notion was essential in the definition of regularity). The domain of $\mathcal{E}\backslash c$ is the principal filter $\mathcal{F}(c)$ of c. $\mathcal{F}(c)$ is a convex subgraph of $G(\mathcal{E})$, thus $\mathcal{F}(c)$ is the intersection of all halfspaces containing $\mathcal{F}(c)$. Therefore, $\mathcal{F}(c)$ can be obtained from the median graph $G(\mathcal{E})$ of the event structure $\mathcal{E}$ by contracting all hyperplanes which do not intersect $\mathcal{F}(c)$.

Given a median graph G and a hyperplane H of its CAT(0) cube complex, the graph G' is obtained by *hyperplane-contraction* of G with respect to H if G' is obtained from G by simultaneously contracting all edges of G dual to H. Clearly, G' is also a median graph. We will say that a median graph G' is a *strong-minor* of a median graph G if G' can be obtained from G by hyperplane-contraction of a set of hyperplanes of G.

Finally recall the event structure $\mathcal{E}_{TY} = (E, \leq, \#)$ occurring in the definition of grid-free event structures. Recall that E consists of three pairwise disjoint sets X, Y, Z satisfying the following conditions:

- $X = \{x_0, x_1, x_2, \ldots\}$ is an infinite set of events with $x_0 < x_1 < x_2 < \cdots$.
- $Y = \{y_0, y_1, y_2, \ldots\}$ is an infinite set of events with $y_0 < y_1 < y_2 < \cdots$.
- $X \times Y \subseteq \|$.
- There exists an injective mapping $g : X \times Y \to Z$ satisfying: if $g(x_i, y_j) = z$ then $x_i < z$ and $y_j < z$. Furthermore, if $i' > i$ then $x_{i'} \not< z$ and of $j' > j$ then $y_{j'} \not< z$.

The domain of $\mathcal{E}_{TY}$ contains the infinite square grid Λ as a strong-minor. This grid corresponds to the events defined by the sets X and Y and is obtained by contracting all hyperplanes corresponding to the events in $E \setminus (X \cup Y \cup Z)$. On the other hand, the events from Z correspond to the hairs attached to the grid Λ in the definition of the hairing of an event structure. However, the relationship between the events of Z or the events of Z and a part of events of $X \cup Y$ is not specified, thus one cannot say more about the structure of the domain of $\mathcal{E}_{TY}$.

We continue with relationships between different types of grids. We start with isometric grids and hyperbolicity.

LEMMA 2.57. *Let $\mathcal{E} = (E, \leq, \#)$ be an event structure of bounded degree. The directed median graph $\overrightarrow{G}(\mathcal{E})$ contains arbitrarily large isometric square grids if and only if E contains two infinite disjoint conflict-free sets of events $X = \{x_0, x_1, \ldots, x_n, \ldots\}, Y = \{y_0, y_1, \ldots, y_n, \ldots\}$ such that $x_i \parallel y_j$ for any two events $x_i \in X, y_j \in Y$.*

An immediate consequence of the previous lemma is the following:

PROPOSITION 2.58. *If the graph $G(\mathcal{E})$ of an event structure $\mathcal{E}$ of bounded degree is hyperbolic, then $\mathcal{E}$ is grid-free.*

7.5. Proof of Theorem 2.12. Since for a Σ-labeled directed graph $\overrightarrow{G}$, the decidability of MSO($\overrightarrow{G}$) implies the decidability of $\mathrm{MSO}_1(G)$, (1) $\Rightarrow$ (2). Since the degrees of vertices of $G(\mathcal{E})$ are uniformly bounded, the implication (2) $\Rightarrow$ (3) follows from Courcelle's Theorem 2.55 [**96**]. The implication (3) $\Rightarrow$ (4) is a particular case of Seese's Theorem 2.54 [**225**]. Finally, the implication (6) $\Rightarrow$ (1) follows from the Müller and Schupp Theorem 2.56 [**178**] that the MSO theory of context-free graphs is decidable. The implication (5) $\Rightarrow$ (6) follows from [**14**, Proposition 4.4] and the fact that trace event structures are recognizable by trace automata (In [**52**], we presented an alternative proof based on the geometric properties of median graphs). It remains to establish the implication (4) $\Rightarrow$ (5).

To establish (4) $\Rightarrow$ (5), we show that if $G(\mathcal{E})$ has clusters of arbitrarily large diameters, then for any n, one can construct in $G(\mathcal{E})$ an $n \times n$ square grid as a minor. This construction uses the properties of clusters and proceeds level-by-level starting from v_0. Namely, we construct larger and larger half square grids containing v_0.

7.6. Proof of Theorem 2.51. The proof of Theorem 2.51 is based on Theorem 2.12 and Propositions 2.59 and 2.60.

PROPOSITION 2.59. *For a trace-regular event structure* $\mathcal{E} = (E, \leq, \#, \lambda)$*, if* $\mathrm{MSO}(\dot{\mathcal{E}})$ *is decidable, then* $\mathrm{MSO}(\overrightarrow{G}(\mathcal{E}))$ *is decidable.*

SKETCH OF THE PROOF. The idea is to transform inductively any formula in $\mathrm{MSO}(\overrightarrow{G}(\mathcal{E}))$ into a formula in $\mathrm{MSO}(\dot{\mathcal{E}})$ where the variables representing the vertices of $\overrightarrow{G}(\mathcal{E})$ will be replaced by variables representing the corresponding hair events of $\dot{\mathcal{E}}$. □

PROPOSITION 2.60. *For a trace-regular event structure* $\mathcal{E} = (E, \leq, \#, \lambda)$*, if* $\mathrm{MSO}(\overrightarrow{G}(\mathcal{E}))$ *is decidable, then* $\mathrm{MSO}(\mathcal{E})$ *is decidable.*

SKETCH OF THE PROOF. Given a formula in $\mathrm{MSO}(\mathcal{E})$, we first transform it into another formula of $\mathrm{MSO}(\mathcal{E})$ where the atomic formulas are of the type $e \in X$, $e_1 = e_2$, $R_a(e)$ for $a \in \Sigma$, and $e_1 \lessdot e_2$. Then, we transform the formula into another formula of $\mathrm{MSO}(\mathcal{E})$ in which each event variable (respectively, each set variable) has a label $a \in \Sigma$, i.e., it can be interpreted only by an event labeled by a (respectively, by a subset of events labeled by a). We can then transform inductively the latter formula into a formula in $\mathrm{MSO}(\overrightarrow{G}(\mathcal{E}))$ where each event variable e is replaced by a second order variable representing a set of vertices S. The idea of this transformation is that an event variable e can be interpreted in $\mathcal{E}$ by an event f if and only if the set S can be interpreted in $\overrightarrow{G}(\mathcal{E})$ by the set of sources of precisely those edges which are dual to the hyperplane $\mathcal{H}_f$. Similarly, a set of events will be represented by the set of sources of the edges dual to the corresponding hyperplanes. □

The "if" implication of Theorem 2.51 is the content of Proposition 2.59. To prove the converse implication, consider a trace-regular event structure $\mathcal{E} = (E, \leq, \#, \lambda)$, such that $\mathrm{MSO}(\overrightarrow{G}(\mathcal{E}))$ is decidable. By Theorem 2.12, $G(\mathcal{E})$ has finite treewidth. Obviously, this implies that $G(\dot{\mathcal{E}})$ has also finite treewidth. By Theorem 2.12, $\mathrm{MSO}(\overrightarrow{G}(\dot{\mathcal{E}}))$ is decidable, and thus, by Proposition 2.60, $\mathrm{MSO}(\dot{\mathcal{E}})$ is decidable.

REMARK 2.61. Notice that the converse of Proposition 2.60 is not true: the MSO theory of trace conflict-free event structures is decidable [**166**], however the graphs of their domains may have infinite treewidth and thus an undecidable MSO theory. For example, the event structure $\mathcal{E} = (E, \leq, \#)$ consisting of two pairwise disjoint sets $X = \{x_0, x_1, x_2, \ldots\}, Y = \{y_0, y_1, y_2, \ldots\}$ of events, such that $x_0 < x_1 < x_2 < \cdots$ and $y_0 < y_1 < y_2 < \cdots$, and all events of X are concurrent with all events of Y, is conflict-free and trace-regular, but its domain $\mathcal{D}(\mathcal{E})$ is the infinite square grid.

8. Counterexamples to Thiagarajan's Conjecture on the decidability of the MSO logic of trace-regular event structures

In this section, we use the general results obtained in Section 7 to construct a counterexample to Thiagarajan's Conjecture 2.6, establishing Theorem 2.9. In view of Theorem 2.51, it suffices to find a trace-regular event structure $\mathcal{E}$ whose graph $G(\mathcal{E})$ has unbounded treewidth (i.e., it contains arbitrarily large square grid minors) and whose hairing $\dot{\mathcal{E}}$ is grid-free (as an event structure). To build such an example, as in Section 6, we start by constructing a finite NPC square complex. Namely, we consider an NPC square complex Z with one vertex, four edges, and three squares, and we show that Z is virtually special. This implies that the principal filter of the universal cover $\widetilde{Z}$ of Z is the domain $\mathcal{D}(\mathcal{E}_Z)$ of a trace-regular event structure (i.e., $\mathcal{E}_Z$ is the event structure unfolding of a net system N_Z). We prove that the median graph $G(\mathcal{E}_Z)$ of the domain

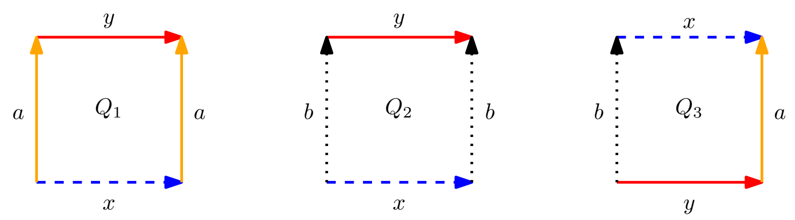


FIGURE 2.9. The three squares defining the VH-complex Z

has unbounded treewidth. On the other hand, to prove that $\dot{\mathcal{E}}_Z$ is grid-free we show that the graph $G(\mathcal{E}_Z)$ of the domain has bounded hyperbolicity (this correspond to bounded isometric square grids).

Badouel et al. [**14**, pp. 144–146] described a trace-regular event structure that has a domain that is not context-free. Using the results of Section 7, we show that the hairing of this event structure is also a counterexample to Conjecture 2.6.

8.1. The First Counterexample. The square complex Z consists of three squares Q_1, Q_2, Q_3, one vertex v_0, and four edges, colored and directed as in Figure 2.9. The four edges of Z are colored orange (color a), black (color b), blue (color x), and red (color y) as indicated in the figure. Since Z is a VH-complex, Z is nonpositively curved. Let $\widetilde{\mathbf{Z}} = (\widetilde{Z}, \widetilde{o}, \widetilde{c})$ denote the directed and colored universal cover of Z. Pick any vertex $\widetilde{v}_0$ of $\widetilde{\mathbf{Z}}$ ($\widetilde{v}_0$ is a lift of v_0) and let $\mathcal{E}_Z$ denote the event structure whose domain is the principal filter $\mathcal{D}_Z = (\mathcal{F}_{\widetilde{o}}(\widetilde{v}_0, \widetilde{Z}^{(1)}), \prec_{\widetilde{o}})$ of $(\widetilde{Z}, \widetilde{o})$. Let also $\vec{G}(\mathcal{E}_Z)$ and $G(\mathcal{E}_Z)$ denote the directed and the undirected 1-skeletons of $\mathcal{D}_Z$. Finally, denote by $\dot{\mathcal{E}}_Z$ the hairing of $\mathcal{E}_Z$.

First we investigate the properties of the complexes Z and $\widetilde{\mathbf{Z}}$, of the graphs $\vec{G}_Z$ and G_Z, and of the event structure $\mathcal{E}_Z$. First, even if Z is not special, we show that it is virtually special:

LEMMA 2.62. *The NPC square complex Z is virtually special. Consequently, the event structures $\mathcal{E}_Z$ and $\dot{\mathcal{E}}_Z$ are trace-regular.*

PROOF. Let Z' be the square complex represented in Figure 2.10. As in Figure 2.4, one has to merge the left and right sides, as well as the lower and the upper sides. Consider the map φ sending all vertices of Z' to the unique vertex of Z, and each edge of Z' to the unique edge of Z with the same color.

The complex Z' has 8 vertices, 32 edges, and 24 squares. In Z', a 4-cycle is the boundary of a square if opposite edges have the same label (and direction) and if the colors of the boundary of this square correspond to the colors of the boundary of one of the three squares of Z. In the figure, the number (2 or 4) in the middle of each 4-cycle represent the number of squares of Z' on the vertices of this 4-cycle. This implies that φ is a covering map from Z' to Z.

Observe that two edges are dual to the same hyperplane of Z' if and only if they have the same label. Using this, it is easy to check that Z' is special. Consequently, Z is virtually special. By Theorem 2.23 and Proposition 2.24, for any vertex $\widetilde{v} \in \widetilde{Z}'$, $\mathcal{F}(\widetilde{v}, \widetilde{Z}')$ is the domain of a trace-regular event structure $\mathcal{E}_{Z'}$ (one can show that $\mathcal{E}_{Z'}$ is independent of the choice of the basepoint $\widetilde{v}$). The fact that $\dot{\mathcal{E}}_Z$ is trace regular follows from Proposition 2.49. □

REMARK 2.63. Observe that Z' coincides with the special cube complex X_{N^*} of the net system N^* from Examples 2.15 and 2.32. Consequently, $\mathcal{E}_Z$ coincides with the event structure unfolding $\mathcal{E}_{N^*}$ of N^*. To obtain a net system $\dot{N}^*$ corresponding to $\dot{\mathcal{E}}_Z$, one can use Proposition 2.50. However in this case, it is enough to add a single transition h to N^* such that ${}^\bullet h = \{C_1, C_2, C_3, C_4\}$ and $h^\bullet = \varnothing$. In the resulting $\dot{N}^*$, for any firing

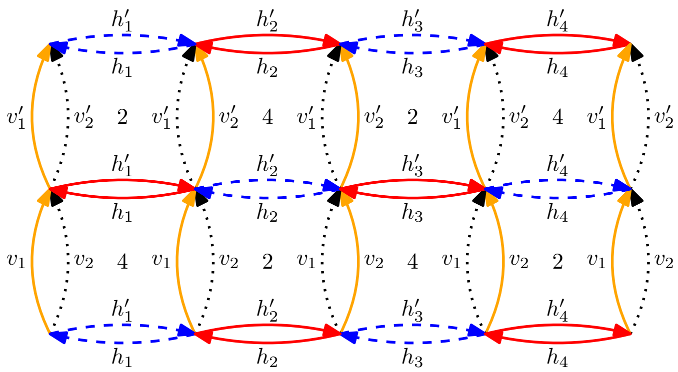


FIGURE 2.10. A finite special cover Z' of the complex Z.

sequence σ of N^*, σh is a firing sequence of $\dot{N}^*$ and no transition can be fired once h has been fired. Using this property, with a proof similar to the proof of Proposition 2.50, one can show that $\dot{\mathcal{E}}_Z$ and $\mathcal{E}_{\dot{N}^*}$ are isomorphic.

The next lemma follows from the fact that with the tile set defining Z one cannot tile a 3×3-square, implying that $G(\mathcal{E}_Z)$ is hyperbolic by Lemma 2.27, and thus that $\mathcal{E}_Z$ and $\dot{\mathcal{E}}_Z$ are grid-free by Proposition 2.58.

LEMMA 2.64. *The graph $G(\mathcal{E}_Z)$ is hyperbolic. Consequently, the event structures $\mathcal{E}_Z$ and $\dot{\mathcal{E}}_Z$ are grid-free.*

REMARK 2.65. Since $\dot{\mathcal{E}}_Z = \mathcal{E}_{\dot{N}^*}$, we can also establish that $\dot{\mathcal{E}}_Z$ is grid-free by considering the net system $\dot{N}^*$ (as suggested by one of the referees of this paper). Using the symmetries of $\dot{N}^*$ and some case analysis, one can show that there exist no reachable marking m of $\dot{N}^*$ and firing sequences σ, σ' such that $m \xrightarrow{\sigma} m$, $m \xrightarrow{\sigma'} m$, and $(a, a') \in I$ for any transition a and a' appearing respectively in σ and σ'. By [**241**, Corollary 5], we conclude that the net system $\dot{N}^*$ is grid-free.

The most technical part of the proof is the following lemma:

LEMMA 2.66. *The graph $G(\mathcal{E}_Z)$ has infinite treewidth, i.e., the directed graph $\overrightarrow{G}(\mathcal{E}_Z)$ is not context-free. Consequently, the theories* $\mathrm{MSO}(\overrightarrow{G}(\mathcal{E}_Z))$, $\mathrm{MSO}_2(G(\mathcal{E}_Z))$, *and* $\mathrm{MSO}(\dot{\mathcal{E}}_Z)$ *are undecidable.*

Similarly to the proof of implication (4)⇒(5) of Theorem 2.12, we construct arbitrarily large half-grids minors rooted at the origin of the domain.

PROOF. We will denote by $z_{i,j}, i, j \geq 0$, the vertices of the half-grid and by $Z_{i,j}, i, j \geq 0$, the connected subgraph of $G(\mathcal{E}_Z)$ which will be mapped (contracted) to $z_{i,j}$. The subgraphs $Z_{i,j}$ are also paths laying in two consecutive spheres $S(\widetilde{v}_0, k-1) \cup S(\widetilde{v}_0, k)$.

For this we use the fact that $\overrightarrow{G}(\mathcal{E}_Z)$ is the graph of the principal filter $\mathcal{D}_Z = (\mathcal{F}_{\widetilde{o}}(\widetilde{v}_0, \widetilde{Z}^{(1)}), \prec_{\widetilde{o}})$ of the universal cover $(\widetilde{Z}, \widetilde{o})$ of Z (here $\widetilde{v}_0$ is an arbitrary but fixed lift of v_0). Since Z has one vertex v_0, all vertices $\widetilde{v}$ of $\overrightarrow{G}(\mathcal{E}_Z)$ are lifts of v_0. Analogously to v_0, each such vertex $\widetilde{v}$ is incident to four outgoing and to four incoming colored edges in $\widetilde{Z}$. However, in the graph $\overrightarrow{G}(\mathcal{E}_Z)$ of the domain, each vertex $\widetilde{v}$ has at most two incoming edges (otherwise, there exists a 3-cube in the interval $I(\widetilde{v}_0, \widetilde{v})$, but this is impossible since $\widetilde{Z}$ is 2-dimensional). The four outgoing edges define three squares Q_1, Q_2, Q_3 having $\widetilde{v}$ as the source (for an illustration, see the first row in Figure 2.11). One can see that there exists an infinite directed path P_a with $\widetilde{v}_0$ as the origin and in which all edges have color orange (color a). Analogously, there exists an infinite directed path P_y with $\widetilde{v}_0$ as the origin and in which all edges have color red (color y). Let $P_a = (\widetilde{u}_0 = \widetilde{v}_0, \widetilde{u}_1, \widetilde{u}_2, \ldots)$

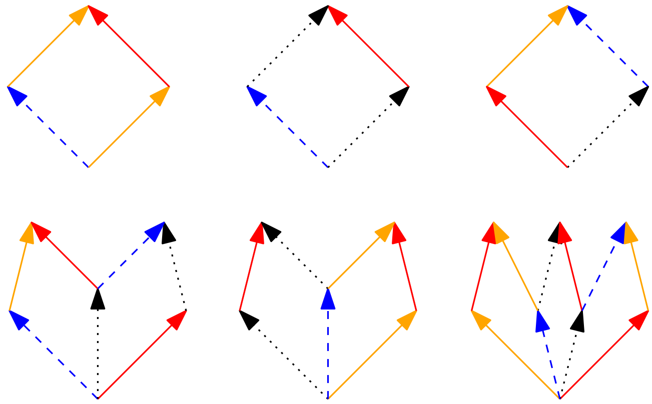


FIGURE 2.11. To the proof of Lemma 2.66

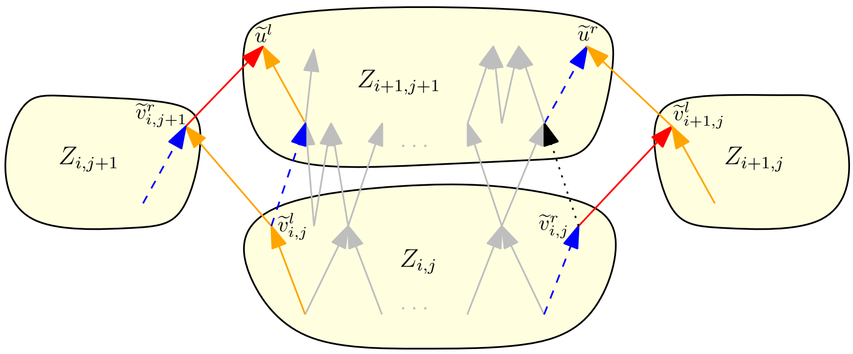


FIGURE 2.12. Construction of the path $Z_{i+1,j+1}$.

and $P_y = (\widetilde{v}_0, \widetilde{v}_1, \widetilde{v}_2 \dots)$ and observe that P_a (respectively, P_y) is a shortest path from $\widetilde{u}_0 = \widetilde{v}_0$ to $\widetilde{u}_i$ (respectively, $\widetilde{v}_i$) for every i.

One can show that for any vertex $\widetilde{v}$, any two outgoing edges of $\widetilde{v}$ can be connected by a path of finite length in the next levels, as indicated in Figure 2.11.

For each k, we construct iteratively a simple path $P_k = P(\widetilde{u}_k, \widetilde{v}_k) = (\widetilde{u}_k = \widetilde{p}_{k,1}, \widetilde{q}_{k,1}, \dots, \widetilde{p}_{k,\ell-1}, \widetilde{q}_{k,\ell-1}, \widetilde{p}_{k,\ell} = \widetilde{v}_k)$ such that $\overrightarrow{\widetilde{q}_{k,1}\widetilde{p}_{k,1}}$ is colored orange (color a), $\overrightarrow{\widetilde{q}_{k,\ell-1}\widetilde{p}_{k,\ell}}$ is colored red (color y), and for each i, $\widetilde{p}_{k,i} \in S(\widetilde{v}_0, k)$ and $\widetilde{q}_{k,i} \in S(\widetilde{v}_0, k-1)$.

Let $P_1 = (\widetilde{u}_1, \widetilde{v}_0, \widetilde{v}_1)$ and suppose that the simple path $P_k = P(\widetilde{u}_k, \widetilde{v}_k)$ has been defined. We define the path $P_{k+1} = P(\widetilde{u}_{k+1}, \widetilde{v}_{k+1})$ in two steps. First, let P'_{k+1} be the path obtained by concatenating the paths obtained by applying the rules of Figure 2.11 to each vertex $q_{k,i}$ of $P_k \cap S(\widetilde{v}_0, k)$ and its two outgoing edges in P_k. Note that the first edges of P_k and P'_{k+1} are consecutive edges in a square Q of $\overrightarrow{G}(\mathcal{E}_Z)$. Since the first edge of P_k is orange (color a), necessarily $Q = Q_1$ and the first edge of P'_{k+1} is red (color y). Analogously, the last edges of P_k and P'_{k+1} are consecutive edges in a square Q' of $\overrightarrow{G}(\mathcal{E}_Z)$. Since the last edge of P_k is red (color y), then necessarily $Q' = Q_3$ and the last edge of P'_{k+1} is orange (color a).

LEMMA 2.67. *P'_{k+1} is a simple path.*

The path $P_{k+1} = P(\widetilde{u}_{k+1}, \widetilde{v}_{k+1})$ is obtained from P'_{k+1} by concatenating the orange (color a) edge $\overrightarrow{\widetilde{u}_k\widetilde{u}_{k+1}}$ at the beginning of P'_{k+1} and the red (color y) edge $\overrightarrow{\widetilde{v}_k\widetilde{v}_{k+1}}$ at the end of P'_{k+1}. We show that P_{k+1} is also a simple path.

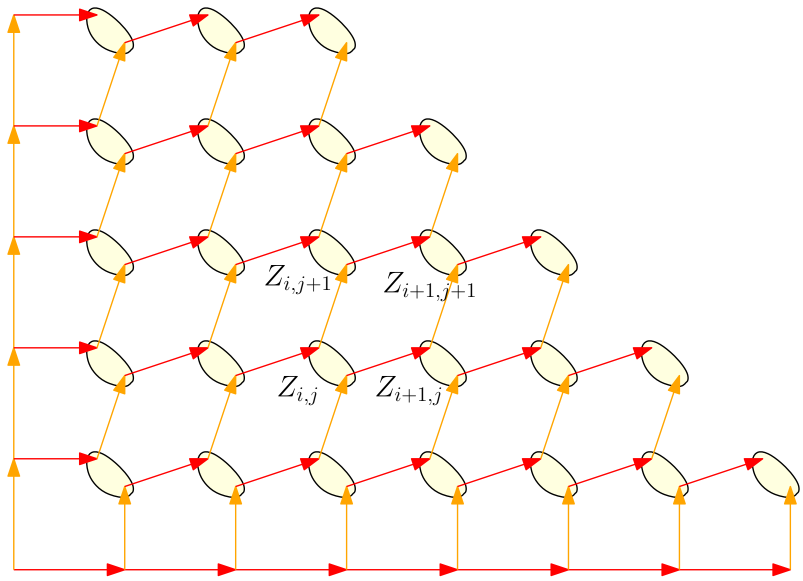


FIGURE 2.13. Part of the half-grid resulting from the contraction of the paths $Z_{i,j}$.

Now, for each k, we construct iteratively the paths $Z_{i,j}$ with $i+j=k$ by selecting subpaths of $P(\widetilde{u}_k, \widetilde{v}_k)$. We require that the paths $Z_{i,j}$ satisfy the following properties (See Figures 2.12 and 2.13):

(1) $Z_{0,j} = \{\widetilde{u}_j\}$ and $Z_{i,0} = \{\widetilde{v}_i\}$ for each $i, j \in \{0, \dots, n\}$;
(2) for each k, if $i+j=k$, then $Z_{i,j}$ is a subpath of P_k;
(3) for each i, j with $i+j=k-1$, the last vertex $\widetilde{v}^r_{i,j+1}$ of the path $Z_{i,j+1}$ appears in P_k before the first vertex $\widetilde{v}^l_{i+1,j}$ of $Z_{i+1,j}$;
(4) each $Z_{i,j}$ with $i, j \geq 1$ has its two end-vertices in $S(\widetilde{v}_0, k)$ and its first edge is orange (color a) and its last edge is blue (color x);
(5) for each pair (i, j) with $i+j=k$, the leftmost vertex $\widetilde{v}^l_{i,j}$ of the path $Z_{i,j}$ is adjacent to the rightmost vertex $\widetilde{v}^r_{i,j+1}$ of the path $Z_{i,j+1}$ by an orange (color a) edge belonging to P_{k+1} and the rightmost vertex $\widetilde{v}^r_{i,j}$ of $Z_{i,j}$ is adjacent to the leftmost vertex $\widetilde{v}^l_{i+1,j}$ of the path $Z_{i+1,j}$ by a red (color y) edge belonging to P_{k+1};
(6) any two distinct paths $Z_{i,j}$ and $Z_{i',j'}$ are disjoint.

The half grid minor of $G(\mathcal{E}_Z)$ then appears when we contract each path $Z_{i,j}$ to a vertex $z_{i,j}$, concluding the proof of the lemma. □

Consequently, by Lemma 2.64, the event structure $\dot{\mathcal{E}}_Z$ is grid-free and by Lemma 2.66, MSO($\dot{\mathcal{E}}_Z$) is undecidable. This concludes the proof of Theorem 2.9.

REMARK 2.68. By construction, the event structure $\dot{\mathcal{E}}_Z$ is strongly regular, and $\dot{\mathcal{E}}_Z$ is hyperbolic by Lemma 2.64. However, $\dot{\mathcal{E}}_Z$ is not strongly regular hyperbolic because $\widetilde{Z}$ (and thus $\dot{\widetilde{Z}}$) is not hyperbolic. Indeed, in $\widetilde{Z}$, it is possible to build an infinite grid by repeating the pattern described in Figure 2.14. Due to the orientation of the edges of this grid, it is easy to see that this grid cannot appear in any principal filter of $(\widetilde{Z}, \widetilde{o})$. Consequently, $\widetilde{Z}$ is not hyperbolic, but any principal filter of $\widetilde{Z}$ is hyperbolic.

8.2. Another Counterexample to Conjecture 2.6. Another counterexample to Conjecture 2.6 can be derived from the hairing $\dot{\mathcal{E}}_{BDR}$ of the trace-regular event structure $\mathcal{E}_{BDR}$ described by Badouel et al. [**14**, pp. 144–146 and Figures 5–9]. The domain of $\mathcal{E}_{BDR}$ is a plane graph defined recursively as a tiling of the quarterplane with origin v_0 by tiles consisting of two squares sharing an edge (see Figure 2.15, left). Namely, we

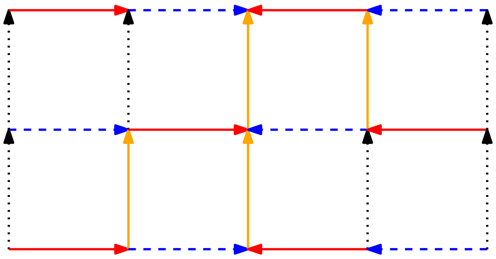


FIGURE 2.14. Part of a infinite grid in $\widetilde{Z}$

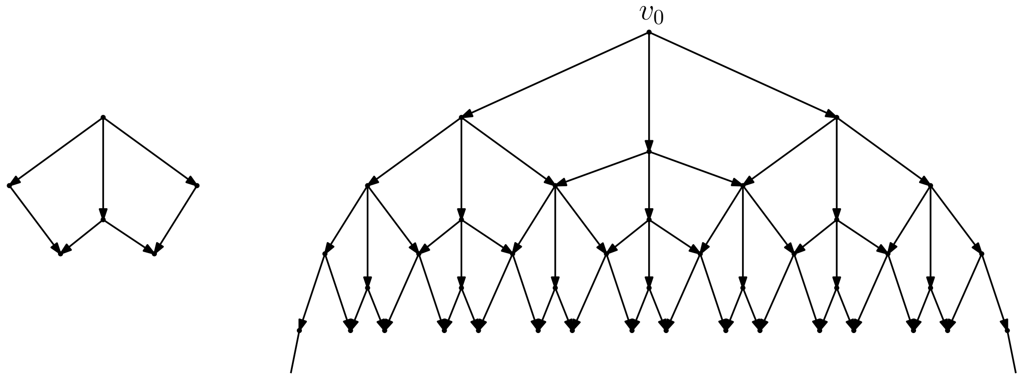


FIGURE 2.15. The tile that we recursively insert to build the domain of $\mathcal{E}_{BDR}$ and the first four steps of the construction.

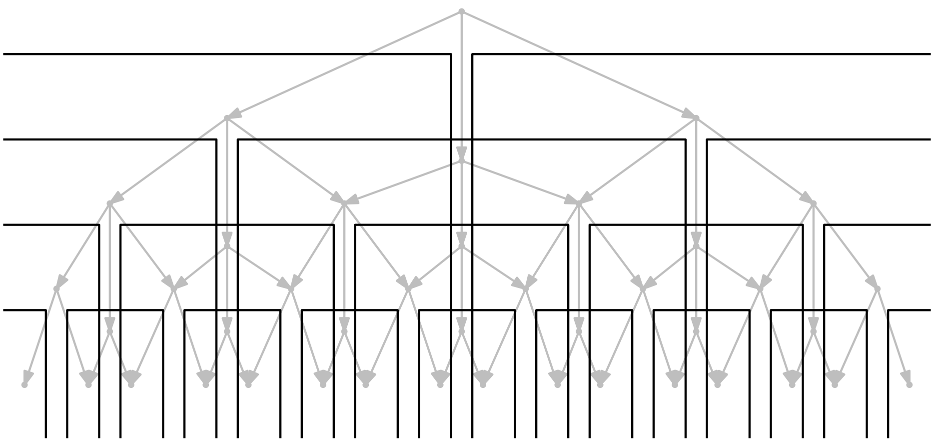


FIGURE 2.16. The hyperplanes obtained during the first four steps of the construction of the domain of $\mathcal{E}_{BDR}$.

start with two infinite directed paths with common origin v_0, and at each step, we insert the tile in each free angle (see Figure 2.15, right for the tiling obtained after the first four steps). As observed in [**14**], the hyperplanes of $G(\mathcal{E}_{BDR})$ can be represented by an arrangement of axis-parallel pseudolines in the plane (see Figure 2.16).

Badouel et al. [**14**] showed that the directed graph $\overrightarrow{G}(\mathcal{E}_{BDR})$ is not context-free. Indeed, for each k, there is a unique level k cluster coinciding with the sphere $S(v_0, k)$ of radius k and the diameters of spheres increase together with their radius. By Theorem 2.12, this shows that the graph $G(\mathcal{E}_{BDR})$ has infinite treewidth. On the other hand one can easily show that the planar graph $G(\mathcal{E}_{BDR})$ has bounded hyperbolicity.

Finally, the fact that $\mathcal{E}_{BDR}$ admits a regular nice labeling was established in [**14**]. Badouel et al. showed that the domain of $\mathcal{E}_{BDR}$ is the domain of a finite trace automaton.

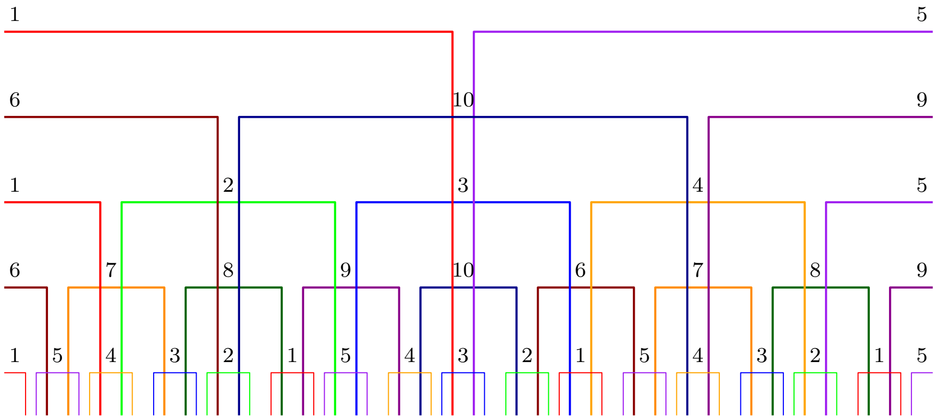


FIGURE 2.17. The trace labeling of the events (hyperplanes) of $\mathcal{E}_{BDR}$ obtained during the first five steps of the construction.

Using the result of Schmitt [**223**] (or the more general one of Morin [**176**]), this implies that $\mathcal{E}_{BDR}$ is a trace-regular event structure.

The labeling of the events (hyperplanes) of $\mathcal{E}_{BDR}$ is given in Figure 2.17 for the events obtained during first five steps of the construction. The idea is that the events constructed at step $4i+1$ are labeled consecutively from left to right $1, 5, 4, 3, 2, 1, \ldots, 1, 5$, those constructed at step $4i+2$ are labeled $6, 10, 9, 8, 7, 6, \ldots, 10, 9$, those constructed at step $4i+3$ are labeled $1, 2, 3, 4, 5, \ldots, 4, 5$, and those constructed at step $4i+4$ are labeled $6, 7, 8, 9, 10, \ldots, 8, 9$. A tedious check of the construction shows that this labeling gives 40 types of labeled principal filters[1].

Consequently, $\mathcal{E}_{BDR}$ is a grid-free trace-regular event structure whose graph $G(\mathcal{E}_{BDR})$ has infinite treewidth. By Theorem 2.51, the MSO theory MSO$(\dot{\mathcal{E}}_{BDR})$ of the hairing of $\mathcal{E}_{BDR}$ is undecidable.

REMARK 2.69. By Corollary 2.34, the domain of $\mathcal{E}_{BDR}$ is the principal filter of the universal cover of some finite (virtually) special cube complex. However, we do not even have an explicit construction of a small NPC square complex X_{BDR} such that the domain of $\mathcal{E}_{BDR}$ is a principal filter of the universal cover of X_{BDR}. To produce such a NPC square complex X_{BDR}, one can use the result of Schmitt [**223**] (or Morin [**176**]) to find a trace-regular labeling of $\mathcal{E}_{BDR}$, then the result of Thiagarajan [**239**, **240**] to construct a net system N_{BDR} such that its event structure unfolding $\mathcal{E}_{N_{BDR}}$ is $\mathcal{E}_{BDR}$, and finally Theorem 2.31 to construct a finite special cube complex X_{BDR} from N_{BDR}. The first two steps of this approach significantly increase the number of labels used to label the events of $\mathcal{E}_{BDR}$ and it is not clear how to avoid this combinatorial explosion.

9. Conclusion

9.1. Regular versus Strongly Regular Event Structures. In view of Proposition 2.21, any strongly regular event structure is regular. One can ask if the converse holds :

QUESTION 2.70. *Is any regular event structure strongly regular?*

REMARK 2.71. In view of Corollary 2.34, if the answer to Question 2.70 is negative, this would provide automatically counterexamples to Thiagarajan's Conjecture 2.3. The counterexamples provided above are of a different kind since they are strongly regular event structures that are not trace-regular.

[1]In [**14**], only 20 types of labeled principal filters are mentioned

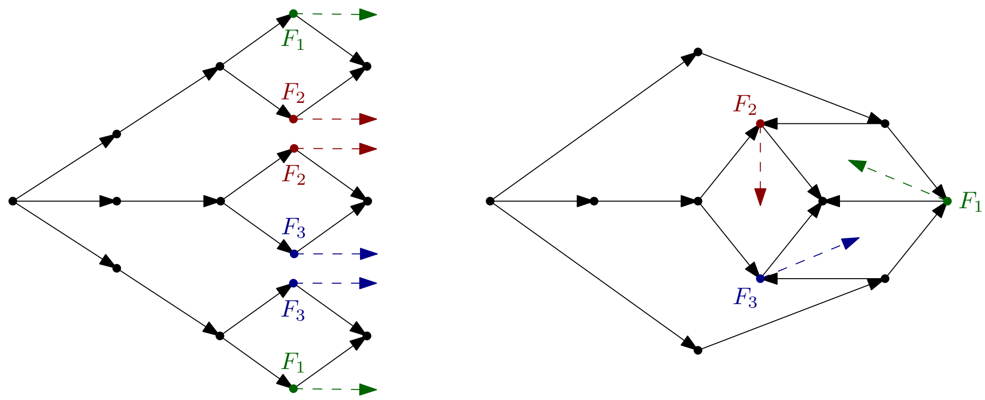


FIGURE 2.18. If we factorize the domain on the left over the equivalence classes of futures, we obtain the square complex on the right that is not an NPC square complex

A natural way to derive a finite directed NPC complex from the domain $\mathcal{D}$ of a regular event structure $\mathcal{E}$ is to factorize $\mathcal{D}$ over all equivalence classes of futures (i.e., to identify in a single vertex all configurations having the same principal filter up to isomorphism). Unfortunately, this construction does not preserve the non-positive curvature of $\mathcal{D}$. For example, consider a domain $\mathcal{D}$ as described on the left of Figure 2.18. In the figure, only a part of the domain is described: one has to imagine that the dashed arrows lead to the remaining part of the domain with the assumption that two nodes that have the same label have isomorphic principal filters. When we factorize the domain $\mathcal{D}$ over the equivalence classes of futures, we obtain the square complex on the left of Figure 2.18. Note that this square complex is not an NPC square complex as it contains three squares that intersect in a vertex and that pairwise intersect on edges and these three squares do not belong to a 3-cube.

This phenomenon does not arise if we consider VH-complexes and isomorphisms that preserve vertical and horizontal edges. More formally, the domain $\mathcal{D} = \mathcal{D}(\mathcal{E})$ of an event structure $\mathcal{E}$ is a *VH-domain* if $\mathcal{D}$ is a VH-complex. In this case, $\mathcal{E}$ is called a *VH-event structure* and the events of $\mathcal{E}$ are partitioned into vertical and horizontal events. A VH-event structure $\mathcal{E}$ is *VH-regular* if $\mathcal{E}$ has finite degree and has a finite number of principal filters up to isomorphism preserving vertical and horizontal events. In this case, the domain $\mathcal{D}(\mathcal{E})$ is called a *regular VH-domain*.

Even in this case, we do not know how to define formally a directed NPC square complex according to the factorization mentioned above such that the original domain is a principal filter of the universal cover of this complex.

QUESTION 2.72. *Does any regular VH-domain occur as a principal filter of the universal cover of some finite directed VH-complex?*

9.2. Hyperbolic Event Domains. There are several natural reasons to investigate hyperbolic event domains. Similarly to CAT(0) and NPC spaces, Gromov hyperbolicity is defined by a metric condition. However, similarly to the CAT(0) property, the hyperbolicity of a CAT(0) cube complex can be expressed in purely combinatorial way, by requiring that all isometric square grids have bounded size. Theorem 2.29 establishes that Thiagarajan's conjecture is true for strongly hyperbolic regular event structures. We conjecture that this result can be generalized in the following way:

CONJECTURE 2.73. *Any strongly regular event structure with a hyperbolic domain admits a trace-regular labeling.*

Conjecture 2.4 was positively solved by Badouel et al. [**14**] for context-free domains, which are particular hyperbolic domains. The following conjecture generalizes Theorem 2.29, the results of [**14**] in the case of event structures considered in this paper, and Conjecture 2.73.

CONJECTURE 2.74. *Any regular event structure with a hyperbolic domain admits a trace-regular labeling.*

9.3. Recognizing Trace-Regular Event Structures and Related Undecidability Questions. We think that the relationship between the existence of aperiodic tile sets and the nonexistence of regular nice labelings of the associated event structures may help to prove some undecidability results. We conjecture that one cannot decide if a regular event structure satisfies Thiagarajan's conjecture:

CONJECTURE 2.75. *There does not exist an algorithm that, given a strongly regular event domain $\mathcal{D}$, can determine whether or not $\mathcal{D}$ admits a regular nice labeling.*

The intuition behind is that one can use Lukkarilla's construction [**164**] to prove this conjecture. As in the proof of undecidability of the classical tiling problem [**27**,**213**], the undecidability proof of Lukkarila is based on a reduction from the *Turing machine halting problem.* More precisely, for any Turing machine $\mathcal{M}$, Lukkarila constructs a 4-way deterministic tile set $T_{\mathcal{M}}$ such that either $T_{\mathcal{M}}$ is an aperiodic tile set (this corresponds to the case when the Turing machine $\mathcal{M}$ does not halt), or $T_{\mathcal{M}}$ does not tile the plane (this corresponds to the case when the Turing machine $\mathcal{M}$ halts). In the first case, by Theorem 2.46, the domain $(\widetilde{W(T_{\mathcal{M}})}_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ does not admit a regular nice labeling. In the second case, by Lemma 2.45, $(\widetilde{W(T_{\mathcal{M}})}_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ is a strongly regular domain that is hyperbolic. Consequently, if Conjecture 2.73 was true, $(\widetilde{W(T_{\mathcal{M}})}_{\widetilde{v}}, \prec_{\widetilde{o}^*})$ would admit a regular nice labeling. This would prove Conjecture 2.75.

Another possible way to prove Conjecture 2.75 would be to anwser the following question in a positive way and use Theorem 2.10.

QUESTION 2.76. *Given a 4-way deterministic tile set T such that there is no valid tiling with the tiles of T, is it true that the VH-complex $W(T)$ is virtually special?*

Note that if there was a positive answer to this question, this would answer a question of Agol [**5**, Question 3] and confirm the following conjecture of Bridson and Wilton [**43**]:

CONJECTURE 2.77 ([**43**, Conjecture 1.2]). *There does not exist an algorithm that, given a finite NPC square complex Y, can determine whether or not Y is virtually special.*

Indeed, in Lukkarila's construction, if the Turing machine $\mathcal{M}$ does not halt, then by Theorem 2.46 $W(T_{\mathcal{M}})$ is not virtually special. On the other hand, if the Turing machine $\mathcal{M}$ halts, then if the answer to Question 2.76 was positive, $W(T_{\mathcal{M}})$ would be virtually special.

9.4. Trace-Regular Event Structures with a Decidable MSO Theory. Even if Theorem 2.13 and Corollary 2.14 give partial answers, the initial fundamental question (Question 2.5) about the characterization of trace-regular event structures that have a decidable MSO theory remains open in general, and even in some very specific cases.

For example, we do not know if the hairing operation is necessary in order to obtain grid-free trace-regular event structures with undecidable MSO theories. In particular, we wonder whether MSO($\mathcal{E}_Z$) and MSO($\mathcal{E}_{BDR}$) are decidable. If this is not the case, this would provide counterexamples to Conjecture 2.6 that are not based on encoding MSO formulas over the domain by MSO formulas over the hair events.

We have shown that there exists domains that are hyperbolic and not context free. However, as mentioned in Remark 2.68, even if the domain of $\dot{\mathcal{E}}_Z$ is hyperbolic, $\widetilde{Z}$ itself is not hyperbolic. This leads to the following open question:

QUESTION 2.78. *Can one construct a finite directed special complex X such that $\widetilde{X}$ is hyperbolic and some principal filter of $\widetilde{X}$ is not context-free?*

In particular, if one consider the trace-regular event structure $\mathcal{E}_{BDR}$ of Badouel et al., we do not have an explicit construction of a finite (virtually) special complex X_{BDR} such that the domain of $\mathcal{E}_{BDR}$ is the domain is a principal filter of the universal cover $\widetilde{X}_{BDR}$ of X_{BDR} (see Remark 2.69). One can ask whether there exists such an X_{BDR} that has a hyperbolic universal cover $\widetilde{X}_{BDR}$.

CHAPTER 3

Hyperbolicity

There exist three classical models of geometries: the usual Euclidean geometry, the hyperbolic geometry, and the elliptic geometry. In these different geometries, the curvature behaves differently (it is 0 in Euclidean geometry, negative in hyperbolic geometry, and positive in elliptic geometry) and this leads to different global geometric and topological properties. In his visionary paper [**128**], Gromov defined CAT(κ) geodesic spaces using a simple 4-point axiom that generalize the three classical geometries. For example, the CAT(0) spaces represent a far reaching common generalization of Euclidean and hyperbolic geometries. In this case, the 4-point axiom just states that the geodesic triangles in the space are thinner than in the Euclidean plane. Another revolutionary concept of Gromov [**128**] is the notion of δ-hyperbolic space. Again defined by a 4-point condition, δ-hyperbolic spaces generalize hyperbolic geometry. CAT(0) and δ-hyperbolicity had a huge impact on the development of geometric group theory [**7, 28, 42, 90, 112, 116**].

The notion of Gromov-hyperbolicity (i.e., δ-hyperbolicity for a finite δ) plays an important roles in the geometry of metric spaces, geometric group theory, and more recently in graph theory and networks theory. Hyperbolicity can be defined in several completely different ways: via the 4-point conditions, via slim triangles, via thin triangles, via linear isoperimetric inequality, via exponential divergence of geodesics, etc. Hyperbolic geodesic spaces, infinite hyperbolic graphs as well as hyperbolic groups (i.e., groups acting geometrically on a hyperbolic graph/space) have deep and interesting asymptotic and structural properties. For example, in hyperbolic groups, the word problem can be solved in linear time [**7, 128**] while it is undecidable in general [**187**]. This is due to the fact that hyperbolic groups and hyperbolic graphs can be characterized by a linear isoperimetric inequality [**42, 128**], and thus the classical Dehn method can be applied efficiently [**42**]. In fact, hyperbolic groups are biautomatic [**48**].

Many classes of important groups and/or graphs/complexes occuring in geometric group theory are known to be hyperbolic: free groups, fundamental groups of compact Riemannian manifolds with negative curvature, some small cancellation groups, curve complexes [**167**], etc. For example, 7-systolic complexes and CAT(0) cube complexes without 2×2 grids are 1-hyperbolic. In fact, systolic or CAT(0) cube complexes without infinite isometric triangular or square grids respectively are hyperbolic.

When considering finite graphs, any graph is hyperbolic for some δ. Therefore, one can define the hyperbolicity of a graph G as the smallest δ such that G is δ-hyperbolic. It can be viewed as a local measure of how close G is to a tree: the smaller the hyperbolicity is, the closer the metrics of its 4-point subspaces are close to tree-metrics. It turns out that many real-world graphs are tree-like from a metric point of view [**1, 2, 36**] or have small hyperbolicity [**149, 180, 227**]. This is due to the fact that many of these graphs (including Internet application networks, web networks, collaboration networks, social networks, biological networks, and others) possess certain geometric and topological characteristics. Hence, for many applications, including the design of efficient algorithms (cf., e.g., [**36, 61, 84–87, 100, 115, 246**]), it is important to design efficient algorithms to compute or approximate the hyperbolicity of a graph, as well as to solve optimization problems on δ-hyperbolic graphs.

I discovered the world of hyperbolic graphs (and then groups) in a unexpected way by considering cop and robber games on graphs. The classical cop and robber game is a pursuit-evasion game played on finite undirected graphs G where the cop attempts to capture the robber. The two players move alternatively, starting with the cop where a move is to slide along an edge of G or to stay at the same vertex. Nowakowski and Winkler [**188**], and Quilliot [**208**] show that the graphs in which the cop always win (the *cop-win* graphs) are exactly the dismantlable graphs (that were already considered in Section 4 of Chapter 1). In [**59**], we considered a generalization of this classical cop and robber game where the players have speeds: at its turm, a player with speed s can traverse at most s edges. Generalizing the results of [**188**,**208**], we prove that a cop with speed s' always captures a robber with speed s in a graph G (i.e., G is (s, s')-cop-win) if and only if G is (s, s')-dismantlable. Rephrasing a result of [**87**], any δ-hyperbolic graph is $(2r, r+2\delta)$-dismantlable, i.e., δ-hyperbolic graphs are $(2r, r+2\delta)$-cop-win. In [**59**], we conjectured that graphs where a cop with speed s' can always capture a faster robber with speed $s > s'$ are always δ-hyperbolic with δ depending only on s. In [**59**], we proved that this conjecture is true for bridged graphs and Helly graphs.

In [**61**], we prove this conjecture. Namely, we showed that:

THEOREM 3.1. *If a finite graph G is (s, s')-cop-win with $0 < s' < s$, then G is δ-hyperbolic with $\delta = 64s^2$.*

Surprisingly, the proof of this theorem uses the characterization of hyperbolic graphs via the linear isoperimetric inequality [**7**,**42**,**128**]. As a nice byproduct of this new characterization of hyperbolic graphs via cop and robber games, we designed an algorithm approximation the hyperbolicity of a graph with n vertices in optimal $O(n^2)$ time (assuming the graph is given by its distance matrix). Its approximation factor is 1569. This large value is the theoretical guarantee we established and it is mainly due to the sequential use of several definitions of hyperbolicity. Nevertheless, the algorithm is simple to implement and once a breadth-first-search tree has been computed, only local operations are executed.

Finding an approximation algorithm with the same time complexity and a better approximation factor was a natural question, and a small approximation factor can lead to interesting applications. In [**53**, **54**], we designed such an algorithm:

THEOREM 3.2. *Given a graph G with n vertices described by its distance matrix, one can compute an 8-approximation (with an additive constant 1) of the hyperbolicity $\delta(G)$ of G in $O(n^2)$ time.*

The algorithm construct a BFS-tree T from a root r and computes an approximation of the thinness of G by considering only geodesic triangles between r and any pair of vertices x, y where the considered geodesics between r and x and r and y are the path of T.

The results of this chapter are based on the papers [**53**, **54**], [**56**, Section 9], [**59**], and [**61**].

1. Gromov-hyperbolicity and its Relatives

1.1. δ-hyperbolic Metric Spaces. Let (X, d) be a metric space and $w \in X$. The *Gromov product*[1] of $y, z \in X$ with respect to w is $(y|z)_w = \frac{1}{2}(d(y, w) + d(z, w) - d(y, z))$. A metric space (X, d) is *δ-hyperbolic* [**128**] for $\delta \geq 0$ if $(x|y)_w \geq \min\{(x|z)_w, (y|z)_w\} - \delta$ for all $w, x, y, z \in X$. Equivalently, (X, d) is δ-hyperbolic if for any $u, v, x, y \in X$, the two largest of the sums $d(u, v) + d(x, y)$, $d(u, x) + d(v, y)$, $d(u, y) + d(v, x)$ differ by at most $2\delta \geq 0$. A metric space (X, d) is said to be *δ-hyperbolic with respect to a basepoint w* if $(x|y)_w \geq \min\{(x|z)_w, (y|z)_w\} - \delta$ for all $x, y, z \in X$.

[1]Informally, $(y|z)_w$ can be viewed as half the detour you make, when going over w to get from y to z.

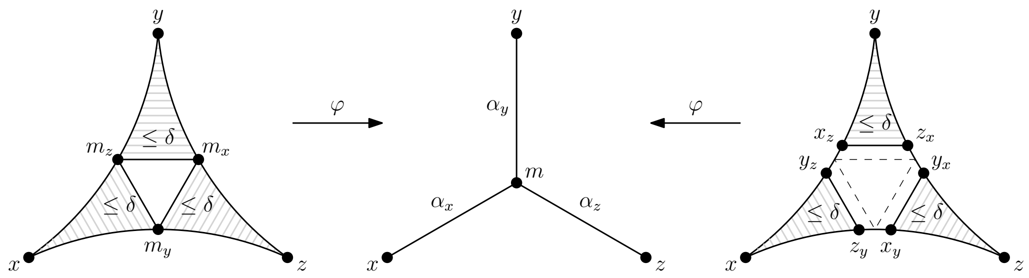


FIGURE 3.1. Insize and thinness in geodesic spaces and graphs.

Let (X, d) be a metric space. An (x, y)-*geodesic* is a (continuous) map $\gamma : [0, d(x, y)] \to X$ from the segment $[0, d(x, y)]$ of $\mathbb{R}^1$ to X such that $\gamma(0) = x, \gamma(d(x, y)) = y$, and $d(\gamma(s), \gamma(t)) = |s-t|$ for all $s, t \in [0, d(x, y)]$. A *geodesic segment* with endpoints x and y is the image of the map γ (when it is clear from the context, by a geodesic we mean a geodesic segment and we denote it by $[x, y]$). A metric space (X, d) is *geodesic* if every pair of points in X can be joined by a geodesic. A *real tree* (or an $\mathbb{R}$-*tree*) [**42**, p.186] is a geodesic metric space (T, d) such that

(1) there is a unique geodesic $[x, y]$ joining each pair of points $x, y \in T$;
(2) if $[y, x] \cap [x, z] = \{x\}$, then $[y, x] \cup [x, z] = [y, z]$.

Let (X, d) be a geodesic metric space. A *geodesic triangle* $\Delta(x, y, z)$ with $x, y, z \in X$ is the union $[x, y] \cup [x, z] \cup [y, z]$ of three geodesics connecting these points. A geodesic triangle $\Delta(x, y, z)$ is called δ-*slim* if for any point u on the side $[x, y]$ the distance from u to $[x, z] \cup [z, y]$ is at most δ. Let m_x be the point of $[y, z]$ located at distance $\alpha_y := (x|z)_y$ from y. Then, m_x is located at distance $\alpha_z := (y|x)_z$ from z because $\alpha_y + \alpha_z = d(y, z)$. Analogously, define the points $m_y \in [x, z]$ and $m_z \in [x, y]$ both located at distance $\alpha_x := (y|z)_x$ from x; see Figure 3.1 for an illustration. We define a tripod $T(x, y, z)$ consisting of three solid segments $[x, m], [y, m]$, and $[z, m]$ of lengths α_x, α_y, and α_z, respectively. The function mapping the vertices x, y, z of $\Delta(x, y, z)$ to the respective leaves of $T(x, y, z)$ extends uniquely to a function $\varphi : \Delta(x, y, z) \to T(x, y, z)$ such that the restriction of φ on each side of $\Delta(x, y, z)$ is an isometry. This function maps the points m_x, m_y, and m_z to the center m of $T(x, y, z)$. Any other point of $T(x, y, z)$ is the image of exactly two points of $\Delta(x, y, z)$. A geodesic triangle $\Delta(x, y, z)$ is called δ-*thin* if for all points $u, v \in \Delta(x, y, z)$, $\varphi(u) = \varphi(v)$ implies $d(u, v) \leq \delta$. The *insize* of $\Delta(x, y, z)$ is the diameter of the preimage $\{m_x, m_y, m_z\}$ of the center m of the tripod $T(x, y, z)$. Below, we remind that the hyperbolicity of a geodesic space can be approximated by the maximum thinness and slimness of its geodesic triangles.

For a geodesic metric space (X, d), one can define the following parameters:

- *hyperbolicity* $\delta(X) = \min\{\delta : X \text{ is } \delta\text{-hyperbolic}\}$,
- *pointed hyperbolicity* $\delta_w(X) = \min\{\delta : X \text{ is } \delta\text{-hyperbolic with respect to a basepoint } w\}$,
- *slimness* $\varsigma(X) = \min\{\delta : \text{any geodesic triangle of } X \text{ is } \delta\text{-slim}\}$,
- *thinness* $\tau(X) = \min\{\delta : \text{any geodesic triangle of } X \text{ is } \delta\text{-thin}\}$,
- *insize* $\iota(X) = \min\{\delta : \text{the insize of any geodesic triangle of } X \text{ is at most } \delta\}$.

PROPOSITION 3.3 ([**7**, **42**, **125**, **128**, **232**]). *For a geodesic metric space (X, d) and any point $w \in X$,*

- $\delta(X) \leq 2\delta_w(X)$,
- $\delta(X) \leq \iota(X) = \tau(X) \leq 4\delta(X)$,
- $\varsigma(X) \leq \tau(X) \leq 4\varsigma(X)$,
- $\delta(X) \leq 2\varsigma(X) \leq 6\delta(X)$.

Due to Proposition 3.3, a geodesic metric space (X, d) is called *hyperbolic* if one of the numbers $\delta(X), \delta_w(X), \varsigma(X), \tau(X), \iota(X)$ (and thus all) is finite. Notice also that a geodesic metric space (X, d) is 0-hyperbolic if and only if (X, d) is a real tree [**42**, p.399] (and in this case, $\varsigma(X) = \tau(X) = \iota(X) = \delta(X) = 0$).

1.2. Hyperbolicity of Graphs. Let $[x, y]$ denote a shortest path connecting vertices x and y in G; we call $[x, y]$ a *geodesic* between x and y. The *interval* $I(u, v) = \{x \in V : d(u, x) + d(x, v) = d(u, v)\}$ consists of all vertices on (u, v)-geodesics. There is a strong analogy between the metric properties of graphs and geodesic metric spaces, due to their uniform local structure. Any graph $G = (V, E)$ gives rise to a geodesic space (X_G, d) (into which G isometrically embeds) obtained by replacing each edge xy of G by a segment isometric to $[0, 1]$ with ends at x and y. X_G is called a *metric graph*. Conversely, by [**42**, Proposition 8.45], any geodesic metric space (X, d) is (3,1)-quasi-isometric to a graph $G = (V, E)$. This graph G is constructed in the following way: let V be an open maximal $\frac{1}{3}$-packing of X, i.e., $d(x, y) > \frac{1}{3}$ for any $x, y \in V$ (that exists by Zorn's lemma). Then two points $x, y \in V$ are adjacent in G if and only if $d(x, y) \leq 1$. Since hyperbolicity is preserved (up to a constant factor) by quasi-isometries, this reduces the computation of hyperbolicity for geodesic spaces to the case of graphs.

The notions of geodesic triangles, insize, δ-slim and δ-thin triangles can also be defined in case of graphs with the single difference that for graphs, the center of the tripod is not necessarily the image of any vertex on the sides of $\Delta(x, y, z)$. For graphs, we "discretize" the notion of δ-thin triangles in the following way. We say that a geodesic triangle $\Delta(x, y, z)$ of a graph G is *δ-thin* if for any $v \in \{x, y, z\}$ and vertices $a \in [v, u]$ and $b \in [v, w]$ ($u, w \in \{x, y, z\}$, and u, v, w are pairwise distinct), $d(v, a) = d(v, b) \leq (u|w)_v$ implies $d(a, b) \leq \delta$. A graph G is *δ-thin*, if all geodesic triangles in G are δ-thin. Given a geodesic triangle $\Delta(x, y, z) := [x, y] \cup [x, z] \cup [y, z]$ in G, let x_y and y_x be the vertices of $[z, x]$ and $[z, y]$, respectively, both at distance $\lfloor (x|y)_z \rfloor$ from z. Similarly, one can define vertices x_z, z_x and vertices y_z, z_y; see Figure 3.1. The *insize* of $\Delta(x, y, z)$ is defined as $\max\{d(y_z, z_y), d(x_y, y_x), d(x_z, z_x)\}$. An interval $I(x, y)$ is said to be κ-thin if $d(a, b) \leq \kappa$ for all $a, b \in I(x, y)$ with $d(x, a) = d(x, b)$. The smallest κ for which all intervals of G are κ-thin is called the *interval thinness* of G and denoted by $\kappa(G)$. Denote also by $\delta(G)$, $\delta_w(G)$, $\varsigma(G)$, $\tau(G)$, and $\iota(G)$ respectively the hyperbolicity, the pointed hyperbolicity with respect to a basepoint w, the slimness, the thinness, and the insize of a graph G.

We will need the following inequalities between $\varsigma(G)$, $\tau(G)$, $\iota(G)$, and $\delta(G)$. This proposition is the counterpart Proposition 3.3 for graphs.

PROPOSITION 3.4. *For any graph G and any vertex w of G,*

- $\delta(G) \leq 2\delta_w(G)$,
- $\delta(G) - \frac{1}{2} \leq \iota(G) = \tau(G) \leq 4\delta(G)$,
- $\varsigma(G) \leq \tau(G) \leq 4\varsigma(G)$,
- $\delta(G) - \frac{1}{2} \leq 2\varsigma(G) \leq 6\delta(G) + 1$,
- $\kappa(G) \leq \min\{\tau(G), 2\delta(G), 2\varsigma(G)\}$.

REMARK 3.5. In general, the converse of the inequality $\kappa(G) \leq 2\delta(G)$ from Proposition 3.4 does not hold: for odd cycles C_{2k+1}, $\kappa(C_{2k+1}) = 0$ while $\delta(C_{2k+1})$ increases with k. However, the following result holds. If G is a graph, denote by G' the graph obtained by subdividing all edges of G once. Papasoglu [**199**] showed that if G' has κ-thin intervals, then G is $f(\kappa)$-hyperbolic for some function f (which may be exponential).

2. Characterizing Hyperbolic Graphs via the Cop and Robber Game

In this section, we outline the proof of Theorem 3.1.

A (non-necessarily finite) graph $G = (V, E)$ is called (s, s')-*dismantlable* if the vertex set of G admits a well-order $\preceq$ such that for each vertex v of G there exists another vertex

u with $u \preceq v$ such that $B_s(v,G) \cap X_v \subseteq B_{s'}(u,G)$, where $X_v := \{w \in V : w \preceq v\}$. In the following, if $B_s(v,G) \cap X_v \subseteq B_{s'}(u,G)$, then we will say that v *is eliminated* by u or that u *eliminates* v. From the definition immediately follows that if G is (s,s')-dismantlable, then G is also (s,s'')-dismantlable for any $s'' > s'$ (with the same dismantling order). In the case of finite graphs, the following result holds (if $s = s' = 1$, this is the classical characterization of cop-win graphs by Nowakowski, Winkler [**188**] and Quilliot [**208**]). Its proof follows the lines of the proof of the classical case, however, it is technically more involved.

THEOREM 3.6. *For any $s, s' \in \mathbb{N} \cup \{\infty\}$, $s' \leq s$, a finite graph G is (s,s')-cop-win if and only if G is (s,s')-dismantlable.*

In fact, one can define two kinds of cop and robber games with speed depending on whether the robber is allowed to go through the position of the cope when it moves or not. This leads to two different characterizations of (s,s')-cop-win graphs. In the statement of the theorem, we consider the game where the robber can traverse the position of the cop. In [**59**], we considered the other variant of the game, but the proofs of both cases are similar.

Rephrasing a result of [**87**], δ-hyperbolic graphs are $(2r, r+2\delta)$-dismantlable.

PROPOSITION 3.7. *For a δ-hyperbolic graph G and any integer $r \geq \delta$, any breadth-first search order $\preceq$ is a $(2r, r+2\delta)$-dismantling order of G.*

Since any (s,s')-dismantlable graph is $(s,s-1)$-dismantlable when $s' < s$, setting $s' = s-1$ in the following Theorem 3.8, and combining it with Theorem 3.6, we get Theorem 3.1.

THEOREM 3.8. *If a graph G is (s,s')-dismantlable with $0 < s' < s$, then G is δ-hyperbolic with $\delta = 16(s+s')\left\lceil \frac{s+s'}{s-s'} \right\rceil + \frac{1}{2} \leq 32\frac{s(s+s')}{s-s'} + \frac{1}{2}$.*

Even if Theorem 3.6 holds only for finite graphs, (s,s')-dismantlability is defined for arbitrary graphs Theorem 3.8 hold for arbitrary graphs. The proof of Theorem 3.8 uses an improved characterization of hyperbolicity via linear isoperimetric inequality and the fact that all (s,s')-dismantlable graphs satisfy such an inequality.

2.1. Linear Isoperimetric Inequality. Now, we recall the definition of hyperbolicity via the linear isoperimetric inequality. Although this (combinatorial) definition of hyperbolicity is given for geodesic metric spaces, it is quite common to approximate the metric space by a graph via a quasi-isometric embedding and to define N-fillings for the resulting graph (see for example, [**42**, pp. 414–417]). Here, we directly give the definitions in the setting of graphs.

In a graph $G = (V,E)$, a *loop* c is a sequence of vertices $(v_0, v_1, v_2, \ldots, v_{n-2}, v_{n-1}, v_0)$ such that for each $0 \leq i \leq n-1$, either $v_i = v_{i+1}$, or $v_i v_{i+1} \in E$; n is called the *length* $\ell(c)$ of c. A *simple cycle* $c = (v_0, v_1, v_2, \ldots, v_{n-2}, v_{n-1}, v_0)$ is a loop such that for all $0 \leq i < j \leq n-1$, $v_i \neq v_j$.

A *non-expansive map* Φ from a graph $G = (V,E)$ to a graph $G' = (V',E')$ is a function $\Phi\colon V \to V'$ such that for all $v, w \in V$, if $vw \in E$ then either $\Phi(v) = \Phi(w)$ or $\Phi(v)\Phi(w) \in E'$. Note that a map Φ from G to G' is non-expansive if and only if for all vertices v, w of G, $d_{G'}(\Phi(v), \Phi(w)) \leq d_G(v,w)$.

For an integer $N > 0$ and a loop $c = (v_0, v_1, v_2, \ldots, v_{n-2}, v_{n-1}, v_0)$ in a graph G, an *N-filling* (D, Φ) of c consists of a 2-connected planar graph D and a non-expansive map Φ from D to G such that the following conditions hold for an example):

(1) the external face of D is a simple cycle $(v'_0, v'_1, \ldots, v'_{n-1}, v'_0)$ such that $\Phi(v'_i) = v_i$ for all $0 \leq i \leq n-1$,
(2) every internal face of D has at most $2N$ edges.

The *N-area* $\mathrm{Area}_N(c)$ of c is the minimum number of faces in an N-filling of c. A graph G satisfies a *linear isoperimetric inequality* if there exists an $N > 0$ such that any loop c of G has an N-filling and $\mathrm{Area}_N(c)$ is linear in the length of c (i.e., there exists a positive integer K such that $\mathrm{Area}_N(c) \leq K \cdot \ell(c)$). The following result of Gromov [**128**] proven in [**7, 38, 42, 189**] is the basic ingredient of our proof:

THEOREM 3.9 (Gromov). *If a graph G is δ-hyperbolic, then any edge-loop of G admits a 16δ-filling of linear area. Conversely, if a graph G satisfies the linear isoperimetric inequality $\mathrm{Area}_N(c) \leq K \cdot \ell(c)$ for some integers N and K, then G is δ-hyperbolic, where $\delta \leq 108K^2N^3 + 9KN^2$.*

All proofs of Theorem 3.9 available in the litterature do not care about the dependencies of the hyperbolicity on the constants K and N. For our purposes, we need the best possible dependencies on these parameters. Therefore, we revisited Theorem 3.9. Namely, we extended this result to the case of rational K and improved its statement by showing that the hyperbolicity of G is quadratic (and not cubic) in N.

PROPOSITION 3.10. *For a graph G and constants $K \in \mathbb{Q}$ and $N \in \mathbb{N}$ such that $2KN$ is a positive integer, if for every cycle c of G, $\mathrm{Area}_N(c) \leq \lceil Kl(c) \rceil$, then the geodesic triangles of G are $16KN^2$-slim and G is $(32KN^2 + \frac{1}{2})$-hyperbolic.*

2.2. Proof of Theorem 3.8. We start by establishing a property satisfied by loops in (s, s')-dismantlable graphs.

LEMMA 3.11. *If a graph G is (s, s')-dismantlable with $s' < s$ and $c = (v_0, v_1, \ldots, v_{n-1}, v_0)$ is a loop of G of length $n > 2(s+s')$, then c contains two vertices $x = v_p, y = v_q$ with $q - p = 2s \mod n$ such that $d(x, y) \leq 2s'$.*

The proof considers the vertex v of c that is the largest for a (s, s')-dismantling order $\preceq$. Then the vertices x and y that are at distance s from v on the loop c are both at distance at most s' from u where u is a vertex of G that eliminates v in $\preceq$.

The following proposition shows that loops of G satisfy a linear isoperimetric inequality.

PROPOSITION 3.12. *If a graph G is (s, s')-dismantlable with $s' < s$ and c is a loop of G, then $\mathrm{Area}_{s+s'}(c) \leq \left\lceil \frac{\ell(c)}{2(s-s')} \right\rceil$.*

PROOF. Let $c = (v_0, v_1, \ldots, v_{n-1}, v_0)$ be a loop of G. To prove that $\mathrm{Area}_{s+s'}(c) \leq \left\lceil \frac{\ell(c)}{2(s-s')} \right\rceil$, it suffices to show that there exists a 2-connected planar graph D and a non-expansive map Φ from D to G such that

(F1) D has at most $\left\lceil \frac{\ell(c)}{2(s-s')} \right\rceil$ faces,
(F2) all internal faces of D have length at most $2(s + s')$,
(F3) the external face of D is a simple cycle $(v'_0, v'_1, \ldots, v'_{n-1}, v'_0)$ such that $\Phi(v'_i) = v_i$ for all $0 \leq i \leq n - 1$.

The image of each face of D will be a loop of G of length at most $2(s + s')$.

We proceed by induction on the length $n := \ell(c)$ of c. If $n \leq 2(s+s')$, let D consists of a single face bounded by a simple cycle $(v'_0, v'_1, \ldots, v'_{n-1}, v'_0)$ of length n and for each i, let $\Phi(v'_i) = v_i$. This shows that $\mathrm{Area}_{s+s'}(c) = 1$.

Now, suppose that $n > 2(s + s')$. By Lemma 3.11 there exist two vertices $x = v_p, y = v_q$ of c with $q - p = 2s \mod n$ and $d(x, y) \leq 2s'$. Suppose without loss of generality that $q = p + 2s$. Let $P' = (x = v_p, v_{p+1}, \ldots, v_{q-1}, v_q = y)$ and $P'' = (x = v_p, v_{p-1}, \ldots, v_0, v_{n-1}, \ldots, v_{q+1}, v_q = y)$ be the two (x, y)-paths constituting c. If $x = y$, let $P = (x, y)$; if $x \neq y$, let $P = (x = w_0, w_1, \ldots, w_k = y)$ be any shortest path in G between x and y. Note that $\ell(P) \leq 2s' < 2s = \ell(P')$.

Let c_0 be the loop obtained as the concatenation of the paths P from x to y and P' from y to x. Since $\ell(P') = 2s$ and $\ell(P) \leq 2s'$, we have $\ell(c_0) \leq 2s + 2s'$. Let c_1 be

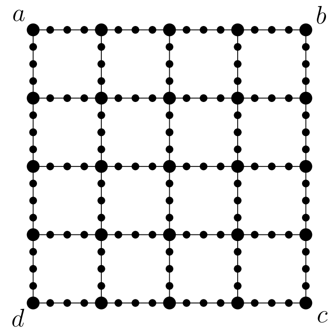


FIGURE 3.2. The graph described in Remark 3.13 when $N = 4$.

the loop obtained as the concatenation of the paths P'' from y to x and P from x to y. Note that $\ell(c_1) = \ell(P) + \ell(P'') \le \ell(P) + \ell(c) - \ell(P') \le \ell(c) - (2s - 2s') < \ell(c)$.

By induction assumption, c_1 admits an $(s + s')$-filling (D_1, Φ_1) satisfying the conditions (F1),(F2), and (F3). Note that the external face of D_1 is bounded by a cycle $(v'_p = x' = w'_0, w'_1, \ldots, w'_k = y' = v'_q, v'_{q+1}, \ldots, v_{n-1}, v_0, \ldots, v'_{p-1}, v'_p)$ such that $\Phi_1(v'_i) = v_i$ for all $i \in [0,p] \cup [q, n-1]$ and $\Phi_1(w'_i) = w_i$ for all $0 \le i \le k$.

Consider the planar graph D obtained from D_1 by adding $q - p - 1$ new vertices forming a path $(x' = v'_p, v'_{p+1}, \ldots, v'_q = y')$ from x' to y' on the external face of D_1 such that the external face of D is bounded by the cycle $(v'_0, v'_1, \ldots, v'_{n-1}, v'_0)$. Let Φ be the non-expansive map defined by $\Phi(v') = \Phi_1(v')$ for every $v \in V(D_1)$ and $\Phi(v'_i) = v_i$ for every $p + 1 \le i \le q - 1$. Clearly, D_1 is a 2-connected planar graph and for each $0 \le i \le n - 1$ we have $\Phi(v'_i) = v_i$. The planar graph D has one more internal face than D_1 that is bounded by the cycle $(x' = v'_p, v'_{p+1}, \ldots, v'_q = y' = w'_k, w'_{k-1}, \ldots, w'_1, w'_0 = x')$. This cycle has the same length as c_0 and is thus bounded by $2(s + s')$. Consequently, (D, Φ) satisfies the conditions (F2) and (F3).

It remains to show that the $(s + s')$-filling (D, Φ) of c satisfies (F1). Since $\ell(c_1) \le \ell(c) - 2(s - s')$, by induction assumption, we obtain

$$\mathrm{Area}_{s+s'}(c) \le \mathrm{Area}_{s+s'}(c_1) + 1 \le \left\lceil \frac{\ell(c_1)}{2(s - s')} \right\rceil + 1 \le \left\lceil \frac{\ell(c) - 2(s - s')}{2(s - s')} \right\rceil + 1 = \left\lceil \frac{\ell(c)}{2(s - s')} \right\rceil,$$

yielding the desired inequality. □

The assertion of Theorem 3.8 follows from Propositions 3.10 and 3.12 by setting $N := s + s'$ and $K := \frac{1}{2N} \cdot \left\lceil \frac{N}{(s-s')} \right\rceil \ge \frac{1}{2(s-s')}$.

REMARK 3.13. The dependence of δ, N in Proposition 3.10 is the "best possible" in the following sense. There are graphs G_N ($N \in \mathbb{N}$) which satisfy $\mathrm{Area}_N(c) \le \lceil l(c) \rceil$ and which are not δ-hyperbolic for $\delta = o(N^2)$ (so δ in general grows quadratically in N). Indeed, take G_N to be a planar square $N \times N$ grid subdivided into squares of side-length N (see Figure 3.2 for an example with $N = 4$). Then clearly for every cycle c, $\mathrm{Area}_{4N}(c) \le \frac{1}{4} \lceil l(c) \rceil$. Consider now the four corners a, b, c, d of the grid (see Figure 3.2); we have $d(a, c) + d(b, d) = 4N^2 > 2N^2 = d(a, b) + d(c, d) = d(a, d) + d(b, c)$ and thus $\delta \ge N^2$.

2.3. An Algorithmic Consequence of Theorem 3.8. We now describe a fast $O(n^2)$ time algorithm for constant-factor approximation of the hyperbolicity $\delta(G)$ of a graph G with n vertices and m edges, assuming that its distance-matrix has already been computed. Our algorithm is very simple and can be used as a practical heuristic to approximate the hyperbolicity of graphs.

The hyperbolicity $\delta(G)$ of a graph G is an integer or a half-integer belonging to the list $\{0, \frac{1}{2}, 1, \frac{3}{2}, 2, \dots n-1, \frac{2n-1}{2}, n\}$. Since 0-hyperbolic spaces are tree-metrics, 0-hyperbolic graphs can be recognized in $O(n^2)$ time (in fact, they are exactly the block graphs [**21**]). Consequently, we assume in the following that $\delta(G) \geq \frac{1}{2}$.

Before presenting the general algorithm (Algorithm 3.2), we describe an auxiliary algorithm (Algorithm 3.1) that for a parameter α either ensures that G is $(784\alpha + \frac{1}{2})$-hyperbolic or that G is not $\frac{\alpha}{2}$-hyperbolic. Algorithm 3.1 is based on Theorem 3.8 and Proposition 3.7.

Algorithm 3.1: Approximated-Hyperbolicity(G, α)

Construct a BFS-order $\preceq$ starting from an arbitrary vertex v_0;
For each $v \in V$, let $f_\alpha(v)$ be the vertex at distance $\min\{2\alpha, d(v, v_0)\}$ from v on the path of the BFS-tree from v to v_0;
for each $v \in V$ **do**
  if $B_{4\alpha}(v) \cap X_v \not\subseteq B_{3\alpha}(f_\alpha(v))$ **then return** NO;
return YES

First, suppose that Algorithm 3.1 returns YES. This means that the BFS-order $\preceq$ is a $(4\alpha, 3\alpha)$-dismantling order of the vertices of G. Consequently, from Theorem 3.8, G is $(784\alpha + \frac{1}{2})$-hyperbolic. Now, suppose that the algorithm returns NO. This means that there exists a vertex v such that $B_{4\alpha}(v) \cap X_v \not\subseteq B_{3\alpha}(f_\alpha(v))$. From Proposition 3.7 with $r = 2\alpha$, this implies that G is not $\frac{\alpha}{2}$-hyperbolic and thus $\delta(G) > \frac{\alpha}{2}$.

Algorithm 3.2 efficiently computes the smallest integer α for which the Algorithm 3.1 returns the answer YES, i.e, the smallest integer α for which the inclusion $B_{4\alpha}(v) \cap X_v \subseteq B_{3\alpha}(f_\alpha(v))$ holds for all vertices v of G. Similarly to Algorithm 3.1, we assume that we have constructed a BFS-order $\preceq$ of the vertices of G starting from an arbitrary but fixed vertex v_0. Suppose that for each vertex v, $p(v)$ denotes the parent of v in the BFS-tree corresponding to $\preceq$ (with the convention that $p(v_0) = v_0$). As in Algorithm 3.1, for each vertex v and for each value of α, let $f_\alpha(v)$ be the vertex at distance $\min\{2\alpha, d(v, v_0)\}$ from v located on the path of the BFS-tree from v to v_0. Note that $f_{\alpha+1}(v) = p(p(f_\alpha(v)))$.

We start with a lemma ensuring that during the execution of the algorithm, we do not have to completely recompute the balls $B_{3\alpha}(f_\alpha(v))$ each time we modify α.

LEMMA 3.14. *If $\alpha' \leq \alpha$, then $B_{3\alpha'}(f_{\alpha'}(v)) \subseteq B_{3\alpha}(f_\alpha(v))$ for any vertex v of G.*

Algorithm 3.2 can be viewed as a "sieve of n stacks" and works as follows. In the preprocessing step, for each vertex v of G, we sort the vertices of G according to their distances to v and successively insert them in a stack $L(v)$ (so that v is the head of $L(v)$). Starting with $\alpha = 1$, for each vertex v of G, we compute $f_\alpha(v)$ and as long as the current head u of $L(v)$ is in $B_{4\alpha}(v)$ and is such that $v \preceq u$ or $d(u, f_\alpha(v)) \leq 3\alpha$, we pop u from $L(v)$. The idea is that none of those popped elements can be a witness for $B_{4\alpha}(v) \cap X_v \not\subseteq B_{3\alpha}(f_\alpha(v))$. If there exists a vertex v which is at distance at most 4α from the head u of its stack $L(v)$, then we have found a witness showing that $B_{4\alpha}(v) \cap X_v \not\subseteq B_{3\alpha}(f_\alpha(v))$. In this case, by Proposition 3.7, we know that G is not $\frac{\alpha}{2}$-hyperbolic. Thus, we increment α by 1 and start a new iteration. By Lemma 3.14, the vertices removed from the stacks do not have to be reconsidered. Otherwise, if each v is at distance $> 4\alpha$ from the current head of $L(v)$, then Algorithm 3.2 returns the current α as the least value for which the Algorithm 3.1 returns the answer YES.

PROPOSITION 3.15. *There exists a constant-factor approximation algorithm to approximate the hyperbolicity $\delta(G)$ of a graph G with n vertices running in $O(n^2)$ time if G is given by its distance-matrix. The algorithm returns a 1569-approximation of $\delta(G)$.*

PROOF. By the previous discussion, Algorithm 3.2 returns the smallest α such that Algorithm 3.1 returns the answer YES. Consequently, G is $(784\alpha + \frac{1}{2})$-hyperbolic and

Algorithm 3.2: Approximated-Hyperbolicity-via-Dismantling(G)

```
Construct a BFS-order ⪯ starting from an arbitrary vertex v_0;
For each v ∈ V, let p(v) be the parent of v in the BFS-tree;
For each v ∈ V, let L(v) be a stack containing all vertices of G sorted
  (increasingly) by their distance to v;
done ← false;
α ← 0;
for each v ∈ V do f_α(v) ← v;
while not done do
    done ← true;
    α ← α + 1;
    for each v ∈ V do
        f_α(v) ← p(p(f_α(v)));
        repeat
            u ← pop(L(v))
        until d(u,v) > 4α or (u ⪯ v and d(u, f_α(v)) > 3α);
        if d(u,v) ≤ 4α then done ← false;
        push(u, L(v));
return α
```

not $\frac{\alpha-1}{2}$-hyperbolic, i.e., $\delta(G) \leq 784\alpha + \frac{1}{2} \leq 1568\delta(G) + \frac{1}{2} \leq 1569\delta(G)$. This gives a 1569-approximation of the hyperbolicity $\delta(G)$ of G.

As to the complexity, first note that computing the BFS-order $\preceq$ and the value of $p(v)$ for each $v \in V$ can be done in time $O(n^2)$ from the distance-matrix of G (this can be done in time linear in the number of edges of G if we are also given the adjacency list of G). Since $|V| = n$ and all the pairwise distances are integers between 0 and n, one can construct each stack $L(v)$ in time $O(n)$ using a counting sort algorithm. Thus, the preprocessing step requires total $O(n^2)$ time. Since during the execution of the algorithm we always have $\alpha \leq 2\delta(G) \leq 2n$, α is incremented at most $2n$ times. Since for each $v \in V$, once a vertex w is popped from $L(v)$, w is no longer used for v at subsequent iterations, there are at most $O(n^2)$ pop operations. Therefore Algorithm 3.2 terminates in time $O(n^2)$. □

Observe that once the BFS-tree has been computed, around each vertex, one consider only a ball of radius $O(\delta(G))$: this is because of the "locality" of the characterization of δ-hyperbolicity via (s, s')-dismantlability.

This locality phenomenon is not so surprising since a local-to-global characterization of hyperbolicity is also available for geodesic spaces:

Theorem 3.16 ([**101**, **128**]). *Given $\delta > 0$, let $R = 10^5\delta$ and $\delta' = 200\delta$. Let (X, d) be a simply connected geodesic metric space in which each loop of length $< 100\delta$ is null-homotopic inside a ball of diameter $< 200\delta$. If every ball $B_R(x_0)$ of X is δ-hyperbolic, then X is δ'-hyperbolic.*

Gromov [**128**] and Papasoglu [**200**] gave an algorithm to recognize Cayley graphs of hyperbolic groups and estimate the hyperbolicity constant δ. The algorithm is based on the theorem that hyperbolicity "propagates", i.e. if balls of an appropriate fixed radius are hyperbolic for a given δ then the whole space is δ'-hyperbolic for some $\delta' > \delta$ (see [**128**], 6.6.F).

3. Hyperbolicity of Weakly Modular Graphs

Weakly modular graphs do not have bounded hyperbolicity in general. However, we are able to show that they satisfy a quadratic isoperimetric inequality. Moreover, we characterize δ-hyperbolic weakly modular graphs by forbidding large isometric square grids and large metric triangles. We also prove that $(s, s-1)$-dismantlable weakly modular graphs are $O(s)$-hyperbolic, improving the quadratic bound we have in the general case.

Since general weakly modular graphs are not δ-hyperbolic, they do not satisfy a linear isoperimetric inequality (for any value of N and K). However, we can show that they satisfy a quadratic isoperimetric inequality. In this case, when considering an N-filling (D, Φ) of a loop c of G, we can assume that the faces of D are either triangles or squares. The following theorem is a refinement of the fact that the triangle-square complex of a weakly modular graph is simply connected and the proof uses the same ideas.

THEOREM 3.17. *In a weakly modular graph G, for any loop c, we have* $\text{Area}_2(c) = \text{Area}_{\triangle\square}(c) \leq 2\ell(c)^2$.

Three vertices u, v, w of a graph G form a *metric triangle uvw* if the intervals $I(u, v), I(u, w)$, and $I(v, w)$ pairwise intersect only in the common end-vertices, i.e., $I(u, v) \cap I(u, w) = \{u\}$, $I(u, v) \cap I(v, w) = \{v\}$, and $I(u, w) \cap I(v, w) = \{w\}$. If $d(v_1, v_2) = d(v_2, v_3) = d(v_3, v_1) = k$, then this metric triangle is called *equilateral* of *size* k. In a weakly modular graph, every metric triangle is equilateral [**80**]. Observe that in a median graph G, all metric triangles of G are reduced to a point and that in a Helly graph G, all metric triangles have sides of length at most 1.

Generalizing Lemma 2.27, we showed that in weakly modular graphs, hyperbolicity can be characterized by the sizes of the isometric square grids and of the metric triangles:

THEOREM 3.18. *For a weakly modular graph G the following are equivalent:*

(i) *there exists δ such that G is δ–hyperbolic;*
(ii) *there exist μ, ν such that the metric triangles of G have sides of length at most μ and G does not contain isometric square grids of side ν.*

More precisely, in a weakly modular graph G, every isometric square grid of G is of side at most $\delta(G)$ and every metric triangle of G is of side at most $4\delta(G)$. Conversely, if the metric triangles of G have sides of length at most μ and isometric square grids of G have sides of length at most ν, then we showed that $\kappa(G) \leq 2\nu + \mu$, and the following result of [**84**] implies that $\delta(G) \leq 32\nu + 20\mu$.

PROPOSITION 3.19 ([**84**]). *If G is a graph in which all the metric triangles of G have sides of length at most μ, then $\delta(G) \leq (16\kappa(G) + 4\mu)$.*

When considering $(s, s-1)$-dismantlable graphs that are weakly modular, we are able to obtain stronger results than in the general case: namely, we show that for any $s' < s$, if a weakly modular graph G is (s, s')-dismantlable, then G is $O(s)$-hyperbolic.

THEOREM 3.20. *If G is an (s, s')-dismantlable weakly modular graph with $s' < s$, then G is $184s$-hyperbolic.*

To prove Theorem 3.20, we say that a cycle c of G is *s-geodesically covered* if there exists a set $\mathcal{P} = \{P_0, P_1, \ldots, P_{n-1}\}$ of geodesics of G such that:

(i) each P_i is a subpath of c,
(ii) each edge of c is contained in a geodesic of $\mathcal{P}$,
(iii) if P_i and P_j are not consecutive (modulo n), then P_i and P_j are edge-disjoint, and
(iv) if P_i and P_j are consecutive (i.e., $j = i+1 \mod n$), then $P_i \cap P_j$ is a path of length $\geq 2s$.

We prove that if a graph G is $(s, s-1)$-dismantlable, then G does not contain s-geodesically covered cycles by considering the last vertex in a potential s-geodesically covered cycle. Using this, we can show that in a $(s, s-1)$-dismantlable weakly modular graph G, the sides of the metric triangles of G are of length at most $6s$ and that $\kappa(G) \leq 10s$. Theorem 3.20 follows then from Proposition 3.19.

As a corollary of Theorem 3.20, reusing the same ideas as in Section 2.3, for any weakly modular graph with n vertices given by its distance-matrix, one can compute in $O(n^2)$ time a value δ' such that $\delta(G) \leq \delta' \leq 736\delta(G) + 368$.

Using the more precise characterization of δ-hyperbolic weakly modular graph given in Theorem 3.18, we can also obtain a local-to-global condition for hyperbolicity for weakly modular graphs, analogous to Theorem 3.16.

Proposition 3.21. *If G is a weakly modular graph such that every ball $B_{4\delta'+1}(v)$ of G is δ'-hyperbolic, then $\delta(G) \leq 112\delta'$.*

Proof. Let ν and μ be respectively the largest sizes of the sides of an isometric square grid Γ and of a metric triangle T of G. Pick a vertex v in of Γ and note that if $\delta' + 1 \leq \nu$, then there exists a $(\delta'+1) \times (\delta'+1)$-isometric square grid in $B_{4\delta'+1}(v)$ and thus $B_{4\delta'+1}(v)$ is not δ'-hyperbolic, a contradiction.

If T is the metric triangle uvw, then all vertices of $I(u, v) \cup I(u, w) \cup I(v, w)$ are at distance at most ν from v [**80**]. If $4\delta' + 1 \leq \mu$, then $B_{4\delta'+1}(v)$ contains a metric triangle with sides of length $(4\delta' + 1)$ and thus $B_{4\delta'+1}(v)$ is not δ'-hyperbolic, a contradiction.

Consequently, $\nu \leq \delta'$ and $\mu \leq 4\delta'$ and therefore, $\delta(G) \leq 32\nu + 20\mu \leq 112\delta'$. □

4. A Fast Factor 8 Approximation Algorithm for Hyperbolicity

As mentioned earlier, computing exactly or approximatively the hyperbolicity of a finite graph is important in the analysis of many real-world networks. For an n-vertex graph G, the definition of hyperbolicity directly implies a simple brute-force $O(n^4)$ algorithm to compute $\delta(G)$. This running time is too slow for computing the hyperbolicity of large graphs that occur in applications [**1, 36, 37, 122**]. On the theoretical side, it was shown that relying on matrix multiplication results, one can improve the upper bound on time-complexity to $O(n^{3.69})$ [**122**]. Moreover, roughly quadratic lower bounds are known [**37, 93, 122**]. In practice, however, the best known algorithm still has an $O(n^4)$-time worst-case bound but uses several clever tricks when compared to the brute-force algorithm [**36**]. Based on empirical studies, an $O(mn)$ running time is claimed, where m is the number of edges in the graph. Furthermore, there are heuristics for computing the hyperbolicity of a given graph [**91**], and there are investigations whether one can compute hyperbolicity in linear time when some graph parameters take small values [**94, 120**].

Perhaps, it is interesting to notice that the first algorithms for testing graph hyperbolicity were designed for Cayley graphs of finitely generated groups (see Theorem 3.16 above). For other algorithms deciding if the Cayley graph of a finitely generated group is hyperbolic, see [**38, 200**]. However, similar methods do not help when dealing with arbitrary graphs.

By Proposition 3.4, if the four-point condition in the definition of hyperbolicity holds for a fixed basepoint w and any triplet x, y, v of X, then the metric space (X, d) is 2δ-hyperbolic. This provides a factor 2 approximation of hyperbolicity of a metric space on n points running in cubic $O(n^3)$ time. Using fast algorithms for computing (max,min)-matrix products, it was noticed in [**122**] that this 2-approximation of hyperbolicity can be implemented in $O(n^{2.69})$ time. In the same paper, it was shown that any algorithm computing the hyperbolicity for a fixed basepoint in time $O(n^{2.05})$ would provide an algorithm for (max, min)-matrix multiplication faster than the existing ones. In [**113**], approximation algorithms are given to compute a $(1 + \epsilon)$-approximation in $O(\epsilon^{-1} n^{3.38})$ time and a $(2 + \epsilon)$-approximation in $O(\epsilon^{-1} n^{2.38})$ time.

The algorithm presented in the previous section and in [**61**] was the first constant-factor approximation algorithm for hyperbolicity of G running in optimal $O(n^2)$ time (when the graph is given by its distance matrix). However, since its approximatio ratio is huge, its practical interest is rather limited. Therefore, the question of designing fast and (theoretically certified) accurate algorithms for approximating graph hyperbolicity is still an important and open question.

In this section, we tackle this open question and propose a very simple (and thus practical) factor 8 algorithm for approximating the hyperbolicity $\delta(G)$ of an n-vertex graph G running in optimal $O(n^2)$ time, establishing Theorem 3.2. As in several previous algorithms, we assume that the input is the distance matrix D of the graph G.

4.1. The Approximation Algorithm. In order to design our algorithm, we introduce a new parameter of a graph G. This parameter depends on an arbitrary fixed BFS-tree of G. It can be computed efficiently and it provides constant-factor approximations for $\delta(G)$, $\varsigma(G)$, and $\tau(G)$.

Consider a graph $G = (V, E)$ and an arbitrary BFS-tree T rooted at some vertex or point w. Denote by x_y the vertex of $[w, x]_T$ at distance $\lfloor (x|y)_w \rfloor$from w and by y_x the vertex of $[w, y]_T$ at distance $\lfloor (x|y)_w \rfloor$ from w. Let $\rho_{w,T}(G) := \max\{d(x_y, y_x) : x, y \in X\}$. In some sense, $\rho_{w,T}(G)$ can be seen as the insize of G with respect to w and T. For this reason, we call $\rho_{w,T}(G)$ the *rooted insize* of G with respect to w and T. The differences between $\rho_{w,T}(G)$ and $\iota(G)$ are that we consider only geodesic triangles $\Delta(w, x, y)$ containing w where the geodesics $[w, x]$ and $[w, y]$ belong to T, and we consider only $d(x_y, y_x)$, instead of $\max\{d(x_y, y_x), d(x_w, w_x), d(y_w, w_y)\}$. Using T, we can also define the thinness of G with respect to w and T: let $\mu_{w,T}(G) = \sup\big\{d(x', y') : \exists x, y \text{ such that } x' \in [w, x]_T, y' \in [w, y]_T \text{ and } d(w, x') = d(w, y') \le (x|y)_w\big\}$. Similarly to Proposition 3.4 establishing that $\iota(G) = \tau(G)$, we can show that these two definitions give rise to the same value.

Proposition 3.22. *For any graph G and any BFS-tree T rooted at a vertex w, $\rho_{w,T}(G) = \mu_{w,T}(G)$.*

The next theorem is the crucial metric property that we use in our algorithm. It establishes that $2\rho_{w,T}(G)$ provides an 8-approximation of the hyperbolicity of $\delta(G)$ (it holds also for infinite graphs).

Theorem 3.23. *Given a graph G and a BFS-tree T rooted at w, $\delta(G) \le 2\rho_{w,T}(G) + 1 \le 8\delta(G) + 1$.*

Proof. Let $\rho := \rho_{w,T}(G)$, $\delta := \delta(G)$, and $\delta_w := \delta_w(G)$. By Gromov's Proposition 3.4, $\delta \le 2\delta_w$. We proceed in two steps. In the first step, we show that $\rho \le 4\delta$. In the second step, we prove that $\delta_w \le \rho + \frac{1}{2}$. Hence, combining both steps we obtain $\delta \le 2\delta_w \le 2\rho + 1 \le 8\delta + 1$.

The first step follows from Proposition 3.4 and from the inequality $\rho \le \iota(G) = \tau(G)$. To prove that $\delta_w \le \rho + 1/2$, for any quadruplet x, y, z, w containing w, we show the four-point condition $d(x, z) + d(y, w) \le \max\{d(x, y) + d(z, w), d(y, z) + d(x, w)\} + (2\rho + 1)$. Assume without loss of generality that $d(x, z) + d(y, w) \ge \max\{d(x, y) + d(z, w), d(y, z) + d(x, w)\}$ and that $d(w, x_y) = d(w, y_x) \le d(w, y_z) = d(w, z_y)$. Since y_x, y_z belong to the shortest path $[w, y]$ of T (that is also a shortest path of G), we have $d(y_x, y_z) = d(y, y_x) - d(y, y_z)$. From the definition of ρ, we also have $d(x_y, y_x) \le \rho$ and $d(y_z, z_y) \le \rho$.

Consequently, by the definition of x_y, y_x, y_z, z_y and by the triangle inequality, we get

$$\begin{aligned} d(y,w)+d(x,z) &\leq d(y,w)+d(x,x_y)+d(x_y,y_x)+d(y_x,y_z)+d(y_z,z_y)+d(z_y,z) \\ &\leq (d(y,y_z)+d(y_z,w))+d(x,x_y)+\rho+d(y_x,y_z)+\rho+d(z_y,z) \\ &= d(y,y_z)+d(w,z_y)+d(x,x_y)+d(y_x,y_z)+d(z_y,z)+2\rho \\ &= d(y,y_z)+d(x,x_y)+(d(y,y_x)-d(y,y_z))+(d(w,z_y)+d(z_y,z))+2\rho \\ &= d(y,y_z)+d(x,x_y)+d(y,y_x)-d(y,y_z)+d(w,z)+2\rho \\ &\leq d(x,y)+1+d(w,z)+2\rho, \end{aligned}$$

the last inequality following from the definition of x_y and y_x in graphs (in the case of geodesic metric spaces, we have $d(x,x_y)+d(y,y_x)=d(x,y)$). This establishes the four-point condition for w,x,y,z and proves that $\delta_w \leq \rho+1/2$. □

We present now a simple self-contained algorithm for computing the rooted insize $\rho_{w,T}(G)$ in $O(n^2)$ time when $G=(V,E)$ is a graph with n vertices, establishing Theorem 3.2.

For any non-negative integer r, let $x(r)$ be the unique vertex of $[w,x]_T$ at distance r from w if $r < d(w,x)$ and the vertex x if $r \geq d(w,x)$. First, we compute in $O(n^2)$ time a table M with rows indexed by V, columns indexed by $\{1,\ldots,n\}$, and such that $M(x,r)$ is the identifier of the vertex $x(r)$ of $[w,x]_T$ located at distance r from w. To compute this table, we explore the tree T starting from w. Let x be the current vertex and r its distance to the root w. For every vertex y in the subtree of T rooted at x, we set $M(y,r):=x$. Assuming that the table M and the distance matrix $D:=(d(u,v):u,v\in X)$ between the vertices of G are available, we can compute $x_y=M(x,\lfloor (x|y)_w\rfloor)$, $y_x=M(y,\lfloor (x|y)_w\rfloor)$ and $d(x_y,y_x)$ in constant time for each pair of vertices x,y, and thus $\rho_{w,T}(G)=\max\{d(x_y,y_x):x,y\in V\}$ can be computed in $O(n^2)$ time.

When the graph G is given by its adjacency list, one can compute its distance-matrix in $O(\min(mn,n^{2.38}))$ time and then use the algorithm described above. However, we explain in the next proposition how to obtain an 8-approximation of $\delta(G)$ in $O(mn)$ time using only linear space.

PROPOSITION 3.24. *For any graph G with n vertices and m edges that is given by its adjacency list, one can compute an 8-approximation (with an additive constant 1) of the hyperbolicity $\delta(G)$ of G in $O(mn)$ time and in linear $O(n+m)$ space.*

PROOF. Fix a vertex w and compute a BFS-tree T of G rooted at w. Note that at the same time, we can compute the value $d(w,x)$ for each $x\in V$.

For each vertex x, consider the map $P_x:\{0,\ldots,d(w,x)\}\to V$ such that for each $0\leq i\leq d(w,x)$, $P_x(i)$ is the unique vertex on the path from w to x in T at distance i from w. For every vertex x, consider the map $Q_x:V\to\mathbb{N}\cup\{\infty\}$ such that for each $y\in V$, $Q_x(y)=d(y,P_x(i))$ if $i=d(w,y)\leq d(w,x)$ and $Q_x(y)=\infty$ otherwise.

We perform a depth first traversal of T starting at w and consider every vertex x in this order. Initially, $P_x=P_w$ can be trivially computed in constant time and $Q_x=Q_w$ can be initialized in $O(n)$ time. During the depth first traversal of T, each time we go up or down, P_x can be updated in constant time. Assume now that a vertex x is fixed. In $O(n+m)$ time and space, we compute $d(x,y)$ for every $y\in V$ by performing a BFS of G from x. Moreover, each time we modify x, for each y, we can update $Q_x(y)$ in constant time by setting $Q_x(y):=\infty$ if $d(w,y)>d(w,x)$, setting $Q_x(y):=d(x,y)$ if $d(w,y)=d(w,x)$, and keeping the previous value if $d(w,u)<d(w,x)$.

We perform a depth first traversal of T from w and consider every vertex y in this order. As for P_x, we can update P_y in constant time at each step. Since $d(w,x),d(w,y)$, and $d(x,y)$ are available, one can compute $(x|y)_w$ in constant time. Therefore, in constant time, we can find $y_x=P_y(\lfloor (x|y)_w\rfloor)$ using P_y and compute $d(x_y,y_x)=Q_x(y_x)$ using Q_x.

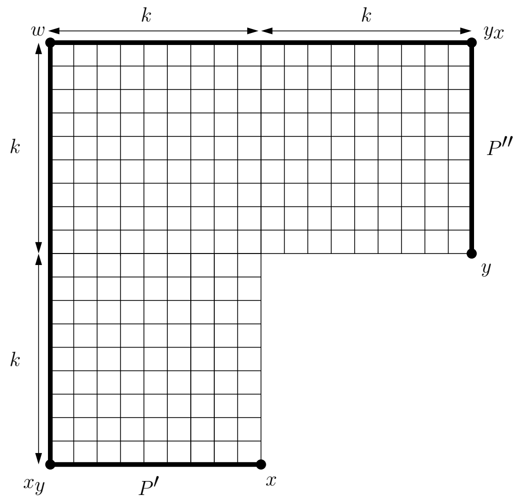


FIGURE 3.3. In H_k, $\rho_{w,T}(H_k) = d(x_y, y_x) = 4k = 4\delta(H_k)$, showing that the inequality $\rho_{w,T}(G) \leq 4\delta$ is tight in the proof of Theorem 3.23.

Consequently, for each x, we compute $\max\{d(x_y, y_x) : y \in V\}$ in $O(m)$ time and therefore, we compute $\rho_{w,T}(G)$ in $O(mn)$ time. At each step, we only need to store the distances from all vertices to w and to the current vertex x, as well as arrays representing the maps P_x, Q_x, and P_y. This can be done in linear space. □

REMARK 3.25. If we are given the distance-matrix D of G, we can use the algorithm described in the proof of Proposition 3.24 to avoid using the $O(n^2)$ space occupied by table M in the first algorithm. In this case, since the distance-matrix D of G is available, we do not need to perform a BFS for each vertex x and the algorithm computes $\rho_{w,T}(G)$ in $O(n^2)$ time.

The following result shows that the bounds in Theorem 3.23 are optimal.

PROPOSITION 3.26. *For any positive integer k, there exists a graph H_k, a vertex w, and a BFS-tree T rooted at w such that $\delta(H_k) = k$ and $\rho_{w,T}(H_k) = 4k$.*

For any positive integer k, there exists a graph G_k, a vertex w, and a BFS-tree T rooted at w such that $\rho_{w,T}(G_k) \leq 2k$ and $\delta(G_k) = 4k$.

PROOF. The graph H_k is the $2k \times 2k$ square grid from which we removed the vertices of the rightmost and downmost $(k-1) \times (k-1)$ square (see Figure 3.3, left). The graph H_k is a median graph and therefore its hyperbolicity is the size of a largest isometrically embedded square subgrid ([**84**, **129**]). The largest square subgrid of H_k has size k, thus $\delta(H_k) = k$.

Let w be the leftmost upmost vertex of H_k. Let x be the downmost rightmost vertex of H_k and y be the rightmost downmost vertex of H_k. Then $d(x, y) = 2k$ and $d(x, w) = d(y, w) = 3k$. Let P' and P'' be the shortest paths between w and x and w and y, respectively, running on the boundary of H_k. Let T be any BFS-tree rooted at w and containing the shortest paths P' and P''. The vertices $x_y \in P'$ and $y_x \in P''$ are located at distance $(x|y)_w = \frac{1}{2}(d(w, x) + d(w, y) - d(x, y)) = 2k$ from w. Thus x_y is the leftmost downmost vertex and y_x is the rightmost upmost vertex. Hence $\rho_{w,T}(H_k) \geq d(x_y, y_x) = 4k$. Since the diameter of H_k is $4k$, we conclude that $\rho_{w,T}(H_k) = 4k = 4\delta(H_k)$.

Let G_k be the $4k \times 4k$ square grid and note that $\delta(G_k) = 4k$. Let w be the center of G_k. We suppose that G_k is isometrically embedded in the ℓ_1-plane in such a way that w is mapped to the origin of coordinates $(0, 0)$ and the four corners of G_k are mapped

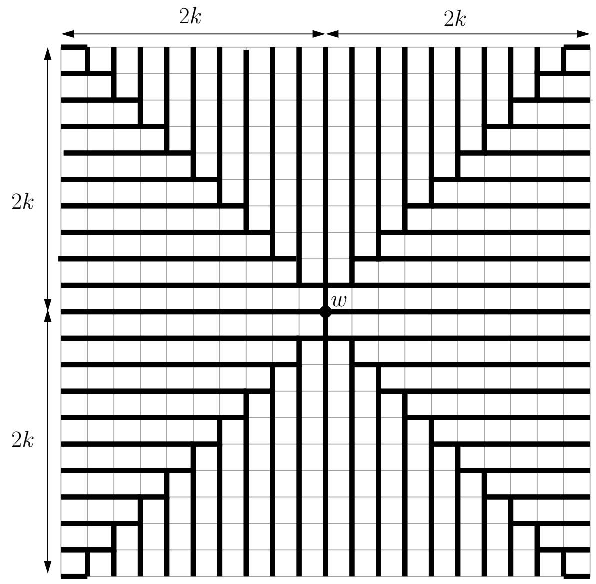


FIGURE 3.4. In G_k, $\rho_{w,T}(G_k) \leq 2k = \frac{1}{2}\delta(G_k)$, showing that (up to an additive factor of 1) the inequality $\delta \leq 2\rho_{w,T}(G) + 1$ is tight in the proof of Theorem 3.23.

to the points with coordinates $(2k, 2k), (-2k, 2k), (-2k, -2k), (2k, -2k)$, We build the BFS-tree T of G_k as follows. First we connect w to each of the corners of G_k by a shortest zigzagging path (see Figure 3.4). For each $i \leq i \leq k$, we add a vertical path from (i, i) to $(i, 2k)$, from $(i, -i)$ to $(i, -2k)$, from $(-i, i)$ to $(-i, 2k)$, and from $(-i, -i)$ to $(-i, -2k)$. Similarly, for each $i \leq i \leq k$, we add a horizontal path from (i, i) to $(2k, i)$, from $(i, -i)$ to $(2k, -i)$, from $(-i, i)$ to $(-2k, i)$, and from $(-i, -i)$ to $(-2k, -i)$. For any vertex $v = (i, j)$, the shortest path of G_k connecting w to v in T has the following structure: it consists of a subpath of one of the zigzagging paths until this path arrives to the vertical or horizontal line containing v and then it continues along this line until v. By case analysis, we can show that $\rho_{w,T}(G_k) \leq 2k = \frac{1}{2}\delta(G_k)$.

□

If instead of knowing the distance-matrix D, we only know the distances between the vertices of G up to an additive error k, then we can define a parameter $\widehat{\rho}_{w,T}(G)$ in a similar way as $\rho_{w,T}(G)$ is defined and show that $2\widehat{\rho}_{w,T}(G) + k + 1$ is an 8-approximation of $\delta(G)$ with an additive error of $3k + 1$.

PROPOSITION 3.27. *Given a graph G with n vertices, a BFS-tree T rooted at a vertex w, and a matrix $\widehat{D}$ such that $d(x, y) \leq \widehat{D}(x, y) \leq d(x, y) + k$, we can compute in time $O(n^2)$ a value $\widehat{\rho}_{w,T}(G)$ such that $\delta(G) \leq 2\widehat{\rho}_{w,T}(G) + k + 1 \leq 8\delta(G) + 3k + 1$.*

4.2. Fast Approximation and Exact Computation of Thinness, Slimness, and Insize. Using Proposition 3.4, Theorem 3.23, and Proposition 3.24, we get the following corollary.

COROLLARY 3.28. *For a graph G with n vertices and a BFS-tree T rooted at a vertex w, $\tau(G) \leq 8\rho_{w,T}(G) + 4 \leq 8\tau(G) + 4$ and $\varsigma(G) \leq 6\rho_{w,T}(G) + 3 \leq 24\varsigma(G) + 3$. Consequently, an 8-approximation (with additive surplus 4) of the thinness $\tau(G)$ and a 24-approximation (with additive surplus 3) of the slimness $\varsigma(G)$ can be found in $O(n^2)$ time for any graph G given by its distance matrix.*

In fact, $\rho_{w,T}(G)$ gives us a 7-approximation of the thinness $\tau(G)$ of G.

THEOREM 3.29. *Given a graph G and a BFS-tree T rooted at w, $\tau(G) \leq 7\rho_{w,T}(G) + 4 \leq 7\tau(G) + 4$. Consequently, a 7-approximation (with an additive constant 4) of the thinness $\tau(G)$ of G can be computed in $O(n^2)$ time for any graph G given by its distance matrix.*

The fact that $\rho_{w,T}(G)$ gives a better approximation for $\tau(G)$ than for $\delta(G)$ is not surprising since $\rho_{w,T}(G)$ can be viewed as the pointed thinness of G. However, the proof of Theorem 3.29 is much more involved than the proof of Theorem 3.23 and several cases have to be considered.

In [**54**], we also provide exact algorithms for computing the slimness $\varsigma(G)$, the thinness $\tau(G)$, and the insize $\iota(G)$ of a given graph G.

THEOREM 3.30. *For a graph $G = (V, E)$ with n vertices and m edges, the following holds:*

(1) the thinness $\tau(G)$ and the insize $\iota(G)$ of G can be computed in $O(n^2 m)$ time;
(2) the slimness $\varsigma(G)$ of G can be computed in $\widehat{O}(n^2 m + n^4/\log^3 n)$ time combinatorially and in $O(n^{3.273})$ time using matrix multiplication.

Both algorithms use $O(n^2)$ space.

When the graph is dense (i.e., $m = \Omega(n^2)$), the time complexity of our algorithms is of the same order of magnitude as the best-known algorithms for computing $\delta(G)$ in practice (see [**36**]), but when the graph is not so dense (i.e., $m = o(n^2)$), our algorithms run in $o(n^4)$ time. In contrast to this result, the existing algorithms for computing $\delta(G)$ exactly are not sensitive to the density of the input.

4.3. Geodesic Spanning Trees. A *geodesic spanning tree rooted at a point w* (a *GS-tree* for short) of a geodesic space (X, d) is a union of geodesics $\Gamma_w := \bigcup_{x \in X} \gamma_{w,x}$ with one end at w such that $y \in \gamma_{w,x}$ implies that $\gamma_{w,y} \subseteq \gamma_{w,x}$. When a GS-tree Γ_w of geodesic metric space (X, d) is given, one can define the parameter $\rho_{w,\Gamma_w}(X)$ as in the case of graphs (without ceilings and taking a sup instead of a max). Then, the analogue of Theorem 3.23 also holds in this case.

THEOREM 3.31. *Given a geodesic metric space X and a GS-tree Γ_w rooted at w, $\delta(G) \leq 2\rho_{w,\Gamma_w}(X) \leq 8\delta(G)$.*

Therefore, the question of the existence of geodesic spanning trees naturally arise. We show that they exist in all complete geodesic metric spaces (recall that a metric space (X, d) is called *complete* if every Cauchy sequence of X has a limit in X):

THEOREM 3.32. *For any complete geodesic metric space (X, d) and for any basepoint w, one can define a geodesic spanning tree $\Gamma_w = \bigcup_{x \in X} \gamma_{w,x}$ rooted at w.*

For finite graphs this is well-known and simple, and such trees can be constructed in various ways, for example via Breadth-First-Search. The existence of BFS-trees in infinite graphs has been established by Polat [**203**, Lemma 3.6]. However for geodesic spaces this result seems to be new (and not completely trivial).

Given a point $w \in X$, the next proposition defines a geodesic $\gamma_{w,x}$ for each point $x \in X$ such that $\Gamma_w = \bigcup_{x \in X} \gamma_{w,x}$ is a geodesic spanning tree of (X, d) rooted at w.

PROPOSITION 3.33. *For any complete geodesic metric space (X, d), for any pair of points $x, y \in X$ one can define an (x, y)-geodesic $\gamma_{x,y}$ such that for all $x, y \in X$ and for all $u, v \in \gamma_{x,y}$, we have $\gamma_{u,v} \subseteq \gamma_{x,y}$.*

PROOF. Let $\preceq$ be a well-order on X. For any $x, y \in X$ we define inductively two sets $P^{\prec v}_{x,y}$ and $P^{v}_{x,y}$ for any $v \in X$:

$$P^{\prec v}_{x,y} = \{x, y\} \cup \bigcup_{u \prec v} P^{u}_{x,y},$$

$$P^{v}_{x,y} = \begin{cases} P^{\prec v}_{x,y} \cup \{v\} & \text{if there is an } (x,y)\text{-geodesic } \gamma \text{ with } P^{\prec v}_{x,y} \cup \{v\} \subseteq \gamma, \\ P^{\prec v}_{x,y} & \text{otherwise.} \end{cases}$$

We set $P_{x,y} = \bigcup_{u \in X} P^{u}_{x,y}$.

CLAIM 3.34. *For all $x, y \in X$ and for any $v \in X$,*

(1) there exists an (x, y)-geodesic $\gamma^{\prec v}_{x,y}$ such that $P^{\prec v}_{x,y} \subseteq \gamma^{\prec v}_{x,y}$,
(2) there exists an (x, y)-geodesic $\gamma^{v}_{x,y}$ such that $P^{v}_{x,y} \subseteq \gamma^{v}_{x,y}$,
(3) there exists an (x, y)-geodesic $\gamma_{x,y}$ such that $P_{x,y} \subseteq \gamma_{x,y}$.

Using this claim, we can show that $P_{x,y}$ is an (x, y)-geodesic and that for each $u, v \in P_{x,y}$, the geodesic $P_{u,v}$ is included in $P_{x,y}$, establishing the proposition. □

If Γ_w is a geodesic spanning tree of X, then X is the union of the images $[w, x]$ of the geodesics of $\gamma_{w,x} \in \Gamma_w$ and one can show that there exists a real tree $T = (X, d_T)$ such that any $\gamma_{w,x} \in \Gamma_w$ is the (w, x)-geodesic of T.

REMARK 3.35. The proof of Theorem 3.32 of the existence of GS-trees is completely different from the proof of Polat [**203**] of the existence of BFS-trees in arbitrary graphs. The proof of [**203**], as the usual BFS-tree construction in finite graphs, constructs an increasing sequence of trees that span vertices at larger and larger distances from the root. In other words, from an arbitrary well-ordering of the set V of vertices of G, Polat [**203**] constructs a well-ordering of V that is consistent with the distances to the root.

When considering arbitrary geodesic metric spaces, a well-ordering consistent with the distances to the basepoint w does not always exist; consider for example the segment $[0, 1]$ with $w = 0$.

5. Conclusion

Our characterization of hyperbolicity via cop and robber games allows to derive an $O(n^2)$-time algorithm to compute an approximation of the hyperbolicity $\delta(G)$ of a graph G with n vertices. Unfortunately, the approximation factor of this algorithm is huge and even if the algorithm is very simple, it would be of little use in practice. However, the bound on the approximation factor is obtained via successive approximations since we use different definitions of the hyperbolicity in the proof. In particular, the core of our characterization of hyperbolicity via cops and robber games is based on the linear isoperimetric inequality that is satisfied by δ-hyperbolic graph. It would be interesting to find an alternative proof of Theorem 3.8 that do not rely on the isoperimetric inequality and that would lead to a better bound on the hyperbolicity.

One interesting property of the algorithm based on the cop and robber game is that once the BFS-tree has been computed, one only need to consider a ball of radius $O(\delta(G))$ around each vertex to compute an approximation of $\delta(G)$. In fact, given a graph endowed with a BFS-tree and a parameter α, there exists a local algorithm in the sense of Peleg [**202**] that can check whether G is 1569α-hyperbolic, or G is not α-hyperbolic. Note that the algorithm described in Section 4 does not have this property since we consider all pairs x, y of vertices. Observe that since locally a cycle and a tree look similar, one cannot hope for a really local algorithm (that does not use the BFS-tree, or some other initial labeling of the network). Note also that from our algorithm, we cannot directly get a local algorithm that computes an approximation of $\delta(G)$ even if we are given a graph G endowed with a BFS-tree. Indeed in our algorithm, we check the balls of radius 4α around all vertices for increasing values of α until all of them satisfy some criteria. But if only one of the ball does not satisfy this criteria, we have to check again all the balls for larger value of α. It would be interesting to know whether there exists a local algorithm where the complexity depends only on the hyperbolicity $\delta(G)$ that starting from a graph G endowed with a BFS-tree can compute an approximation of the hyperbolicity $\delta(G)$ of G. One can imagine that each node computes a value and that the maximum value computed gives an approximation of $\delta(G)$.

CHAPTER 4

Representation Maps for Maximum and Ample classes

One of the oldest open question in computational machine learning is the important compression conjecture by Littlestone and Warmuth [**163**]/Floyd and Warmuth [**119**]. It asserts that any concept class C of VC-dimension d admits a sample compression scheme of size $O(d)$.

Combinatorially, a concept class is just a set family. Graph theoretically, a concept class C can be viewed as a subgraph $G(C)$ of the hypercube induced by the $\{0,1\}$-vectors encoding the concepts of C. Geometrically and topologically, a concept class C can be viewed as the cube complex $X(C)$ of $G(C)$. This graph-theoretical and geometrico-topological point of view on concept classes enable to use results from geometry in machine learning.

In this chapter, we design unlabeled sample compression scheme for maximum classes and characterize such schemes for ample classes (a.k.a. lopsided or extremal) via representation maps and unique sink orientations. We also construct an example of a maximum class of dimension 3 without corners. This refutes several previous works in machine learning from the past 11 years. In particular, it implies that all previous constructions of optimal unlabeled sample compression schemes for maximum classes are erroneous.

The Sauer-Shelah-Perles Lemma [**222**, **228**, **245**] is arguably the most basic fact in VC theory; it asserts that any class $C \subseteq \{0,1\}^n$ satisfies $|C| \leq \binom{n}{\leq d}$, where $d =$ VC-dim(C). A beautiful generalization of Sauer-Shelah-Perles's inequality asserts that $|C| \leq |\overline{X}(C)|$, where $\overline{X}(C)$ is the family of subsets that are shattered by C.[1] The latter inequality is a part of the Sandwich Lemma [**10**, **35**, **110**, **195**], which also provides a lower bound for $|C|$ (and thus "sandwiches" $|C|$) in terms of the number of its *strongly shattered subsets* (see Section 1). A class C is called *maximum/ample* if the Sauer-Shelah-Perles/Sandwich upper bounds are tight (respectively). Every maximum class is ample, but not vice versa.

Maximum classes were studied mostly in discrete geometry and machine learning, e.g. [**118**, **119**, **123**, **155**, **249**]. The history of ample classes is more interesting as they were discovered independently by several works in disparate contexts [**10**, **19**, **35**, **110**, **159**, **173**, **250**]. Consequently, they received different names such as lopsided classes [**159**], extremal classes [**35**, **173**], and ample classes [**19**, **110**]. Lawrence [**159**] was the first to define them for the investigation of the possible sign patterns realized by points of a convex set of $\mathbb{R}^d$. Interestingly, Lawrence's definition of these classes does not use the notion of shattering nor the Sandwich Lemma. In this context, these classes were discovered by Bollobás and Radcliffe [**35**] and Bandelt et al. [**19**], and the equivalence between the two definitions appears in [**19**]. Ample classes admit a multitude of combinatorial and geometric characterizations [**19**, **35**, **159**] and comprise many natural examples arising from discrete geometry, combinatorics, graph theory, and geometry of groups [**19**, **159**].

Main Results. A *corner* in an ample class C is any concept $c \in C$ that belongs to a unique maximal cube of C (equivalently, c is a corner if $C \setminus \{c\}$ is also ample). A sequence of corner removals leading to a single concept is called a *corner peeling*; corner

[1]Note that this inequality indeed implies the Sauer-Shelah-Perles Lemma, since $|\overline{X}(C)| \leq \binom{n}{\leq d}$.

peeling is a strong version of *collapsibility*. Wiedemann [**250**] and independently Chepoi (unpublished, 1996) asked whether every ample class has a corner. The machine learning community studied this question independently in the context of *sample compression schemes* for maximum classes: Rubinstein and Rubinstein [**216**] showed that corner peelings lead to optimal *unlabeled sample compression schemes (USCS)*.

The following result refutes this conjecture.

THEOREM 4.1. *There exists a maximum class C_H of VC-dimension 3 without any corner.*

The crux of the proof is an equivalence between corner peelings and partial shellings of the cross-polytope. This equivalence translates the question whether corners always exist to the question whether partial shellings can always be extended. The latter was an open question in Ziegler's book on polytopes [**261**], and was resolved in Hall's PhD thesis where he presented an interesting counterexample [**135**]. The ample class resulting from Hall's construction yields a maximum class without corners.

Sample compression is a powerful technique to derive generalization bounds in statistical learning. Littlestone and Warmuth [**163**] introduced it and asked if every class of VC-dimension $d < \infty$ has a sample compression scheme of a finite size (see Section 1 for a definition). This question was later precised by Floyd and Warmuth [**119, 247**] to *whether a sample compression scheme of size $O(d)$ exists.* The first question was recently resolved by [**175**] who exhibited an $\exp(d)$ sample compression. The second question however remains one of the oldest open problems in machine learning (for more background we refer the reader to [**174**] and the books [**226, 251**]).

Rubinstein and Rubinstein [**216**, Theorem 16] showed that the existence of a corner peeling for a maximum class C implies a *representation map* for C (see Section 2 for a definition), which is known to yield an optimal unlabeled sample compression scheme of size VC-dim(C) [**155**].[2] They claim, using an interesting topological approach, that maximum classes admit corner peelings. Unfortunately, Theorem 4.1 shows that this does not hold.

While our Theorem 4.1 rules out the program of deriving representation maps from corner peelings, we provide an alternative derivation of representation maps for maximum classes and therefore also of an optimal unlabeled sample compression scheme for them.

THEOREM 4.2. *Any maximum class $C \subseteq 2^U$ of VC-dimension d admits a representation map, and consequently, an unlabeled compression scheme of size d.*

We next turn to construction of representation maps for ample classes. We present a local-to-global characterization of such maps via *unique sink orientations.*

THEOREM 4.3. *For an ample class C, map $r : C \to 2^U$ is a representation map if and only if r is the out-map of a unique sink orientation*

An orientation of the edges of a cube B is a *unique sink orientation (USO)* if any subcube $B' \subseteq B$ has a unique sink. Szabó and Welzl [**236**] showed that any USO of B leads to a representation map for B. We extend this bijection to ample classes C by proving that representation maps are equivalent to orientations o of C such that (i) o is a USO on each subcube $B \subseteq C$, and (ii) for each $c \in C$ the edges outgoing from c belong to a subcube $B \subseteq C$. We further show that any ample class admits orientations satisfying each one of those conditions. However, *the question whether all ample classes admit representation maps remains open.*

The results of this chapter are based on the paper [**57, 58**].

[2] Pálvölgyi and Tardos [**196**] recently exhibited a (non-ample) class C with no USCS of size VC-dim(C).

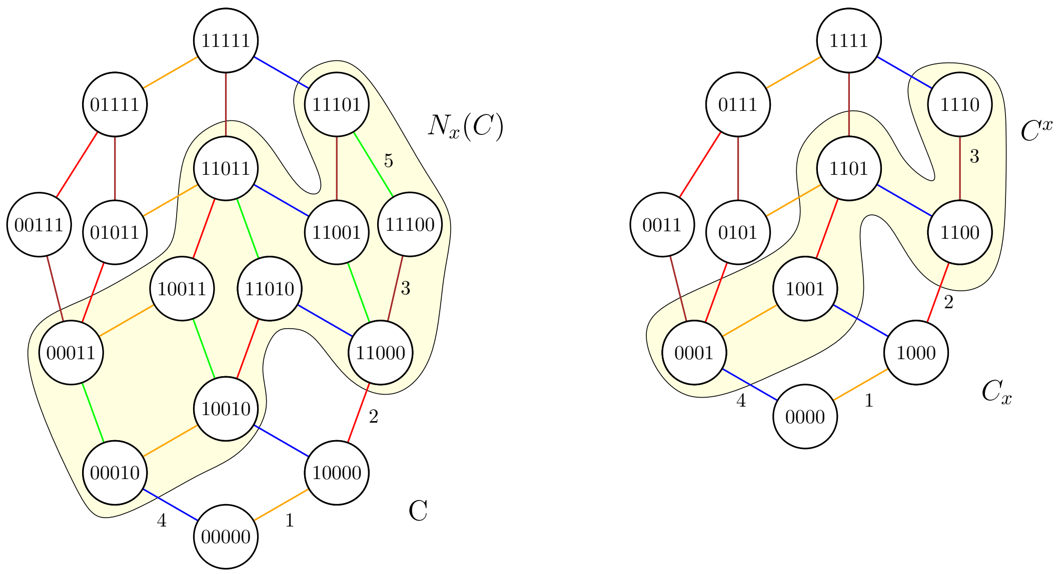


FIGURE 4.1. A 2-dimensional maximum class $C \subseteq 2^{\{1,2,3,4,5\}}$ on the left and the restriction C_x for $x = 5$ on the right. The reduction C^x corresponds to the restriction of the carrier $N_x(C)$.

1. Concept Classes and (Unlabeled) Sample Compression Schemes

A *concept class* C is a set of subsets (concepts) of a finite ground set U which is called the *domain* of C and denoted $\mathrm{dom}(C)$. We sometimes treat the concepts as characteristic functions rather than subsets. The *support* (or *dimension set*) $\mathrm{supp}(C)$ of C is the set $\{x \in U : x \in c' \setminus c'' \text{ for some } c', c'' \in C\}$. $C^* := 2^U \setminus C$ is the *complement* of C. The *restriction* of C on $Y \subseteq U$ is the class $C|Y = \{c \cap Y : c \in C\}$ whose domain is Y. We use C_Y as shorthand for $C|(U \setminus Y)$; in particular, we write C_x for $C_{\{x\}}$ (see Figure 4.1 for an example), and c_x for $c|(U \setminus \{x\})$ for $c \in C$ (note that $c_x \in C_x$). A class $B \subseteq 2^U$ is a *cube* if there exists $Y \subseteq U$ such that $B|Y = 2^Y$ and B_Y contains a single concept (denoted by $\mathrm{tag}(B)$). Note that $\mathrm{supp}(B) = Y$ and therefore we say that B is a Y*-cube*; $|Y|$ is called the *dimension* $\dim(B)$ of B. Two cubes B, B' with the same support are called *parallel cubes*. A cube B is *maximal* if there is no cube B' such that $B \subsetneq B'$.

Let Q_n denote the n-dimensional cube where $n = |U|$; $c, c' \in Q_n$ are called adjacent if the symmetric difference $c\Delta c'$ is of size 1. The *1-inclusion graph* of C is the subgraph $G(C)$ of Q_n induced by the vertex-set C when the concepts of C are identified with the corresponding vertices of Q_n. Any cube $B \subseteq C$ is called *a cube of* C. The *cube complex* of C is the set $Q(C) = \{B : B \text{ is a cube of } C\}$. The *dimension* of $Q(C)$ is $\dim(Q(C)) := \max_{B \in Q(C)} \dim(B)$. A concept $c \in C$ is called a *corner* of C if c belongs to a unique maximal cube of C. The *reduction* C^Y of C to $Y \subseteq U$ is the class $C^Y := \{\mathrm{tag}(B) : B \in Q(C) \text{ and } \mathrm{supp}(B) = Y\}$ whose domain is $U \setminus Y$. When $x \in U$ we denote $C^{\{x\}}$ by C^x and call it *the* x*-hyperplane of* C (see Figure 4.1 for an example). Note that a concept c belongs to C^x if and only if c and $c \cup \{x\}$ both belong to C. The union of all cubes of C having x in their support is called the *carrier* of C^x and is denoted by $N_x(C)$. If $c \in N_x(C)$, we also denote $c|U \setminus \{x\}$ by c^x (note that $c^x \in C^x$).

A class C is *connected* if the graph $G(C)$ is connected. Let $d_{G(C)}(c, c')$ denote the distance between c and c' in $G(C)$. Note that $d_{Q_n}(c, c') =: d(c, c')$ coincides with the Hamming distance $|c\Delta c'|$. Let $B(c, c') = \{t \subseteq U : d(c, t) + d(t, c') = d(c, c')\}$ be the *interval* between c and c' in Q_n. A class C is called *isometric* if $d(c, c') = d_{G(C)}(c, c')$ for any $c, c' \in C$ and *weakly isometric* if $d(c, c') = d_{G(C)}(c, c')$ if $d(c, c') \leq 2$. Any path connecting two concepts tag(B) and tag(B') of C^Y inside C^Y can be lifted to a path of Y-cubes connecting B and B' in C; such a path of cubes is called a *gallery*.

A class C *shatters* $Y \subseteq U$ if $C|Y = 2^Y$. C *strongly shatters* Y if C contains a Y-cube. Let $\overline{X}(C)$, $\underline{X}(C)$ denote the simplicial complexes $\overline{X}(C) = \{Y : C \text{ shatters } Y\}$, $\underline{X}(C) = \{Y : C \text{ strongly shatters } Y\}$. Note that $\underline{X}(C) \subseteq \overline{X}(C)$. The *VC-dimension* VC-dim(C) of C is the size of a largest set C shatters. The *Sandwich Lemma* asserts that $|\underline{X}(C)| \leq |C| \leq |\overline{X}(C)|$.

A *labeled sample* is a set $s = \{(x_1, y_1), \dots, (x_m, y_m)\}$, where $x_i \in U$ and $y_i \in \{0, 1\}$. An *unlabeled sample* is a set $\{x_1, \dots, x_m\}$, where $x_i \in U$. Given a labeled sample $s = \{(x_1, y_1), \dots, (x_m, y_m)\}$, the unlabeled sample $\{x_1, \dots, x_m\}$ is the domain of s and is denoted by dom(s). A sample s is *realizable* by a concept $c : U \to \{0, 1\}$ if $c(x_i) = y_i$ for every i, and s is realizable by a concept class C if it is realizable by some $c \in C$.

A *sample compression scheme* for a concept class C is best viewed as a protocol between a *compressor* and a *reconstructor*. The compressor gets a realizable sample s from which it picks a small subsample s'. The compressor sends s' to the reconstructor. Based on s', the reconstructor outputs a concept c that needs to be consistent with the entire input sample s. A sample compression scheme has size k if for every realizable input sample s the size of the compressed subsample s' is at most k. An *unlabeled (sample) compression scheme* (USCS) is a sample compression scheme in which the compressed subsample s' is unlabeled. So, the compressor removes the labels before sending the subsample to the reconstructor.

2. Ample and Maximum Classes

We briefly review the main characterizations and basic geometric examples of ample and maximum classes. The next theorem summarizes the main characterizations of ample classes:

THEOREM 4.4 ([**19, 35, 159**]). *The following are equivalent for a class C:*

(1) C is ample;
(2) C^* is ample;
(3) $\underline{X}(C) = \overline{X}(C)$;
(4) $|\underline{X}(C)| = |C|$;
(5) $|\overline{X}(C)| = |C|$;
(6) $C \cap B$ is ample for any cube B;
(7) $(C^Y)_Z = (C_Z)^Y$ for all partitions $U = Y \cup Z$;
(8) for all partitions $U = Y \cup Z$, either $Y \in \underline{X}(C)$ or $Z \in \underline{X}(C^*)$.

Condition (3) leads to a simple definition of ampleness: C is ample if whenever $Y \subseteq U$ is shattered by C, then there is a Y-subcube of C. Thus, if C is ample we will write $X(C)$ instead of $\underline{X}(C) = \overline{X}(C)$. A *representation map* for an ample class C is a bijection $r : C \to X(C)$ satisfying the *non-clashing condition*: $c|(r(c) \cup r(c')) \neq c'|(r(c) \cup r(c'))$, for all $c, c' \in C, c \neq c'$. We continue with metric and recursive characterizations of ample classes:

THEOREM 4.5 ([**19**]). *The following are equivalent for a class C:*

(1) C is ample;
(2) C^Y is connected for all $Y \subseteq U$;
(3) C^Y is isometric for all $Y \subseteq U$;
(4) C is isometric, and both C_x and C^x are ample for all $x \in U$;
(5) C is connected and all hyperplanes C^x are ample.

COROLLARY 4.6. *Two maximal cubes of an ample class C have different supports.*

Indeed, if B and B' are two d-cubes with the same support, by Theorem 4.5(2) B and B' can be connected in C by a gallery, and thus B is contained in a $d+1$-cube. Therefore, B and B' cannot be maximal.

The Sandwich Lemma and Theorem 4.4(5) imply that maximum classes are ample. Basic examples of maximum classes are concept classes derived from hyperplane arrangements in $\mathbb{R}^n$, ball arrangements in $\mathbb{R}^n$, and unions of n intervals in $\mathbb{R}$. The following theorem summarizes some characterizations of maximum classes provided in [**118**, **119**, **123**, **249**]:

THEOREM 4.7. *The following are equivalent for a class C:*
(1) C is maximum;
(2) C_Y is maximum for all $Y \subseteq U$;
(3) C_x and C^x are maximum for all $x \in U$;
(4) C^* is maximum.

We continue with some important geometric examples of ample classes.

1. Simplicial complexes. Every simplicial complex S (viewed as a set system closed under taking subsets) is ample since $\overline{X}(S) = \underline{X}(S)$.

2. Realizable ample classes. Let $K \subseteq \mathbb{R}^n$ be a convex set. Let $C(K) := \{\text{sign}(v) : v \in K, v_i \neq 0, \forall i \leq n\}$, where $\text{sign}(v) \in \{\pm 1\}^n$ is the sign pattern of v. Lawrence [**159**] showed that $C(K)$ is ample, and called ample classes representable in this manner *realizable.*

3. Median classes. A class C is called *median* if for every three concepts c_1, c_2, c_3 of C their *median* $m(c_1, c_2, c_3) := (c_1 \cap c_2) \cup (c_1 \cap c_3) \cup (c_2 \cap c_3)$ also belongs to C. Observe that a class C is median if and only if its 1-inclusion graph $G(C)$ is a median graph. Median classes are ample by [**19**, Proposition 2]. Due to their relationships with other discrete structures, median classes are one of the most important examples of ample classes. Median classes are equivalent to finite median graphs (a well-studied class in metric graph theory, see [**18**]), to CAT(0) cube complexes, i.e., cube complexes of global nonpositive curvature (central objects in geometric group theory, see [**128**, **219**]), and to the domains of event structures (a basic model in concurrency theory [**183**, **253**]).

4. Convex geometries and conditional antimatroids. Let C be a class such that (i) $\varnothing \in C$ and (ii) $c, c' \in C$ implies that $c \cap c' \in C$. Call $x \in c \in C$ *extremal* if $c \setminus \{x\} \in C$. We say that $c \in C$ is *generated* by $s \subseteq c$ if c is the smallest member of C containing s. A class C satisfying (i) and (ii) with the additional property that every member c of C is generated by its extremal points is called a *conditional antimatroid* [**19**, Section 3]. If $U \in C$, then we obtain the well-known structure of a *convex geometry* (called also an *antimatroid*) [**114**]. By [**19**, Proposition 1], conditional antimatroids C are ample since $\underline{X}(C)$ coincides with the sets of extremal points and $\overline{X}(C)$ coincides with the set of all minimal generating sets of sets from C. Convex geometries comprise many examples from geometry, ordered sets, and graphs; see the foundational paper [**114**]. For example, a *realizable convex geometry* is a convex geometry $C \subseteq U$ such that U can be realized as a set of $\mathbb{R}^n$ and $c \in C$ if and only if c is the intersection of a convex set of $\mathbb{R}^n$ with U.

We continue with two examples of conditional antimatroids.

EXAMPLE 4.8. Closer to usual examples from machine learning, let U be a finite set of points in $\mathbb{R}^n$, no two points sharing the same coordinate, and let the concept class C_Π consist of all intersections of axis-parallel boxes of $\mathbb{R}^n$ with U. Then C_Π is a convex geometry: for each $c \in C_\Pi$, $\text{ex}(c)$ consists of all points of c minimizing or maximizing one of the n coordinates. Clearly, for any $p \in \text{ex}(c)$, there exists a box Π such that $\Pi \cap U = c \setminus \{p\}$.

EXAMPLE 4.9. A *partial linear space* is a pair (P, L) consisting of a finite set P whose elements are called *points* and a family L of subsets of P, whose elements are called *lines*, such that any line contains at least two points and any two points belong to at most one line. The projective plane (any pair of points belong to a common line and any two lines intersect in exactly one point) is a standard example, but partial linear spaces comprise many more examples. The concept class $L \subseteq 2^P$ has VC-dimension at most 2 because any two points belong to at most one line. Now, for each line $\ell \in L$ fix an arbitrary total order π_ℓ of its points. Let L^* consist of all subsets of points that belong to a common line ℓ and define an interval of π_ℓ. Then L^* is still a concept class of VC-dimension 2. Moreover, L^* is a conditional antimatroid: if $c \in L^*$ and c is an interval of the line ℓ, then $\mathrm{ex}(c)$ consists of the two end-points of c on ℓ.

5. Ample Classes from Graph Orientations. Kozma and Moran [**153**] used the sandwich lemma to derive several properties of graph orientations. They also presented two examples of ample classes related to distances and flows in networks (see also [**159**, p.157] for another example of a similar nature). Let $G = (V, E)$ be an undirected simple graph and let o^* be a fixed reference orientation of E. To an arbitrary orientation o of E associate a concept $c_o \subseteq E$ consisting of all edges which are oriented in the same way by o and by o^*. It is proven in [**153**, Theorem 26] that if each edge of G has a non-negative capacity, a source s and a sink t are fixed, then for any flow-value $w \in \mathbb{R}^+$, the set C_w^{flow} of all orientations of G for which there exists an (s,t)-flow of value at least w is an ample class. An analogous result was obtained if instead of the flow between s and t one consider the distance between those two nodes.

3. Corner Peelings and Partial Shellings

In this section, we prove that corner peelings of ample classes are equivalent to isometric orderings of C as well as to partial shellings of the cross-polytope. This equivalence, combined with a result by Hall [**135**] yields a maximum class with VC dimension 3 without corners (Theorem 4.1). Let $C_< := (c_1, \ldots, c_m)$ be an ordering of the concepts in C. For any $1 \leq i \leq m$, let $C_i := \{c_1, \ldots, c_i\}$ denote the *i'th level set*. The ordering $C_<$ is called:

(1) an *ample* ordering if every level set C_i is ample;
(2) a *corner peeling* if every c_i is a *corner* of C_i;
(3) an *isometric* ordering if every level set C_i is isometric;
(4) a *weakly isometric* ordering if every level set C_i is weakly isometric.

Proposition 4.10. *The following are equivalent for an ordering $C_<$ of an isometric class C:*

(1) $C_<$ is ample;
(2) $C_<$ is a corner peeling;
(3) $C_<$ is isometric;
(4) $C_<$ is weakly isometric.

A concept class C is *dismantlable* if it admits an ordering satisfying any of the equivalent conditions (1)–(4) in Proposition 4.10. Isometric orderings of Q_n are closely related to shellings of its dual, the *cross-polytope* O_n (which we define next). Define $\pm U := \{\pm x_1, \ldots, \pm x_n\}$; so, $|\pm U| = 2n$, and we call $-x_i, +x_i$ *antipodal.* The n-dimensional *cross-polytope* is the pure simplicial complex of dimension n whose facets are all $\sigma \subseteq \pm U$ that contain exactly one element in each antipodal pair. Thus, O_n has 2^n facets and each facet σ of O_n corresponds to a vertex c of Q_n ($+x_i \in \sigma$ if and only if $x_i \in c$). Observe that $x_i \in c' \Delta c''$ if and only if $\{+x_i, -x_i\} \subseteq \sigma' \Delta \sigma''$ where σ' correspond to c' and σ'' corresponds to c''.

Let X be a pure simplicial complex (PSC) of dimension d, i.e., a simplicial complex in which all facets have size d. Two facets σ, σ' are adjacent if $|\sigma \Delta \sigma'| = 2$. A *shelling* of X is an ordering $\sigma_1, \ldots, \sigma_p$ of all of its facets such that $2^{\sigma_j} \bigcap (\bigcup_{i<j} 2^{\sigma_i})$ is a PSC of

dimension $d-1$ for every $j \leq p$ [**261**, Lecture 8]. A *partial shelling* is an ordering of some facets that satisfies the above condition. Note that $\sigma_1, \ldots, \sigma_m$ is a partial shelling if and only if for every $i < j$ there exists $k < j$ such that $\sigma_i \cap \sigma_j \subseteq \sigma_k \cap \sigma_j$, and $\sigma_k \cap \sigma_j$ is a facet of both σ_j and σ_k. X is *extendably shellable* if every partial shelling can be extended to a shelling. We next establish a relationship between partial shellings and isometric orderings.

PROPOSITION 4.11. *Every partial shelling of the cross-polytope O_n defines an isometric ordering of the corresponding vertices of the cube Q_n. Conversely, if C is an isometric class of Q_n, then any isometric ordering of C defines a partial shelling of O_n.*

Consequently, if all ample classes are dismantlable, then O_n is extendably shellable.

It was asked in [**261**] if any cross-polytope O_n is extendably shellable. In the PhD thesis of H. Tracy Hall from 2004, a nice counterexample to this question is given [**135**]. Hall's counterexample arises from the 299 regions of an arrangement of 12 pseudo-hyperplanes. These regions are encoded as facets of the cross-polytope O_{12} and it is shown in [**135**] that the subcomplex of O_{12} consisting of all other facets admits a shelling which cannot be extended. By Proposition 4.11, the ample concept class C_H defined by those 299 simplices does not have any corner (see Figure 4.2 for a picture of C_H).[3] A counting shows that C_H is a maximum class of VC-dimension 3. This completes the proof Theorem 4.1.

Implications on Previous Works. Theorem 4.1 proves that several previous results in machine learning are erroneous:

- Rubinstein and Rubinstein [**216**, Theorem 32] showed that any maximum class can be represented by a simple arrangement of piecewise-linear hyperplanes. In [**216**, Theorem 39], they claim that sweeping such an arrangement leads to a corner peeling of the corresponding maximum class. This is unfortunately false, as witnessed by Theorem 4.1.
- Kuzmin and Warmuth [**155**] constructed unlabeled sample compression schemes for maximum classes based on the presumed uniqueness of a certain matching (their Theorem 10). This theorem is wrong as it implies the existence of corners. However their conclusion is correct: in Theorem 4.2 we show that such unlabeled compression schemes exist.
- Theorem 3 by Samei, Yang, and Zilles [**220**] is built on a generalization of Theorem 10 from [**155**] to the multiclass case which is also incorrect.
- Theorem 26 by Doliwa et al. [**108**] uses the result by [**216**] to show that the Recursive Teaching Dimension (RTD) of maximum classes equals to their VC dimension. However the VC dimension 3 maximum class from Theorem 4.1 has RTD at least 4. It remains open whether the RTD of every maximum class C is bounded by $O(\text{VC-dim}(C))$.

Positive Results. On the positive side, some particular ample classes are dismantlable. For example, median classes are dismantlable. More generally, we show that conditional antimatroids are dismantlable.

PROPOSITION 4.12. *Conditional antimatroids are dismantlable.*

When considering 2-dimensional classes, it was proved in [**216**, Theorem 34] that 2-dimensional maximum classes are dismantlable. This was later generalized to 2-dimensional ample classes in [**171**]. In [**57**], we provide a different proof based on a local characterization of convex sets of general ample classes.

[3]For the interested reader, a file containing the 299 concepts of C_H represented as elements of $\{0,1\}^{12}$ is available at `https://arxiv.org/src/1812.02099/anc/CH.txt`

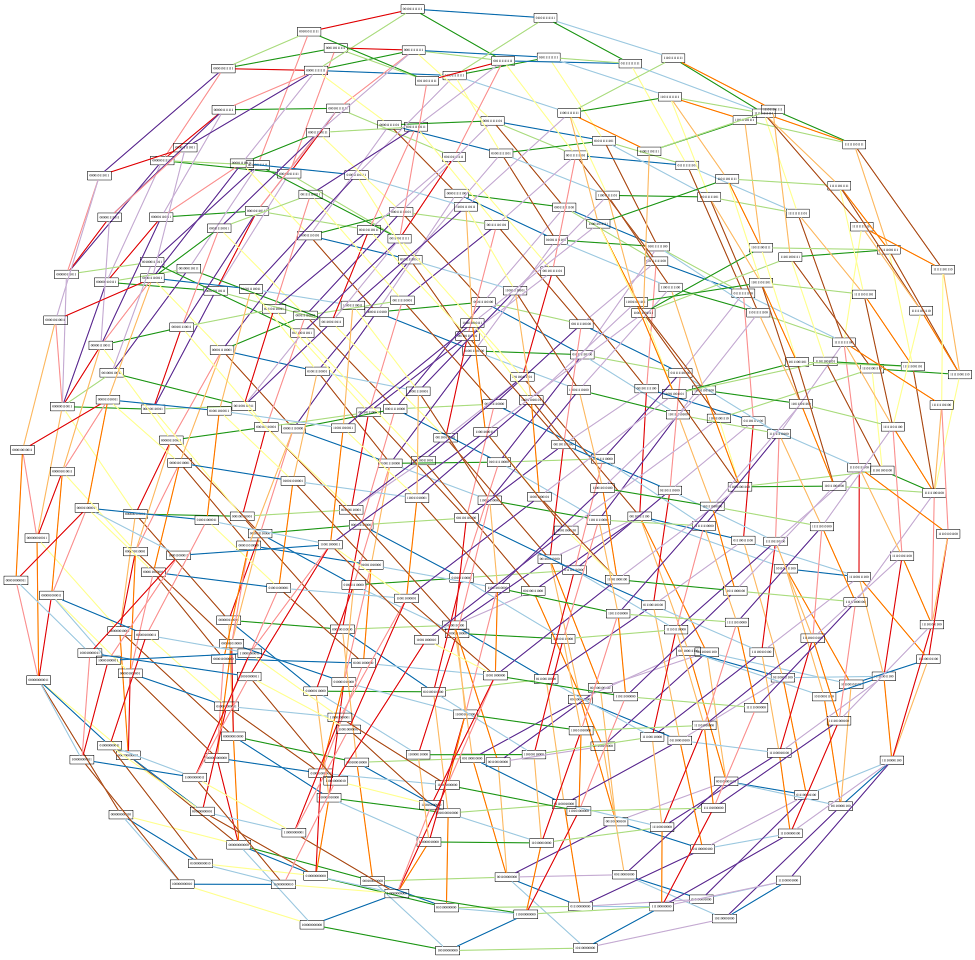


FIGURE 4.2. The maximum class $C_H \subset 2^{12}$ without corners of VC-dimension 3 with $\binom{12}{\leq 3} = 299$ concepts. A different edge color is used for each of the 12 dimensions. Best viewed in color.

A *free face* of a cube complex $Q(C)$ is a face Q of $Q(C)$ strictly contained in only one other face Q' of $Q(C)$. An *elementary collapse* is the deletion of a free face Q (thus also of Q') from $Q(C)$. A cube complex $Q(C)$ is *collapsible* if C can be reduced to a single vertex by a sequence of elementary collapses. Collapsibility is a stronger version of contractibility. The sequences of elementary collapses of a collapsible cube complex $Q(C)$ can be viewed as discrete Morse functions [**121**] without critical cells, i.e., acyclic perfect matchings of the face poset of $Q(C)$. From the definition it follows that if C has a corner peeling, then the cube complex $Q(C)$ is collapsible: the sequence of elementary collapses follows the corner peeling order (in general, detecting if a finite complex is collapsible is NP-complete [**237**]). Theorem 4.5(5) implies that the cube complexes of ample classes are contractible (see also [**20**] for a more general result). In fact, the cube complexes of ample classes are collapsible:

PROPOSITION 4.13. *If $C \subseteq 2^U$ is an ample class, then the cube complex $Q(C)$ is collapsible.*

4. Representation Maps for Maximum Classes

In this section, we prove Theorem 4.2, i.e., that maximum classes admit representation maps, and therefore, by a result of [**155**], they admit optimal unlabeled compression schemes.

The crux of the proof of Theorem 4.2 is the following proposition. Let C be a d-dimensional maximum class and let $D \subseteq C$ be a $(d-1)$-dimensional maximum subclass. A *missed simplex* for the pair (C, D) is a simplex $\sigma \in X(C) \setminus X(D)$. Note that any missed simplex has size d. An *incomplete cube* Q for (C, D) is a cube of C such that $\mathrm{supp}(Q)$ is a missed simplex. For any incomplete cube Q with $\sigma = \mathrm{supp}(Q)$, $C|\sigma$ and $D|\sigma$ are maximum classes of dimensions d and $d-1$, respectively. Since $|\sigma| = d$, we have $|C|\sigma| = \binom{d}{\leq d} = \binom{d}{\leq d-1} + 1 = |D|\sigma| + 1$. Since $Q|\sigma = C|\sigma$, there exists a unique concept $c \in Q$ such that $c|\sigma \notin D|\sigma$. We denote c by $s(Q)$, and call c the *source* of Q. In fact, the source map is a bijection between missed simplices for (C, D) and concepts of $C \setminus D$:

PROPOSITION 4.14. *Each $c \in C \setminus D$ is the source of a unique incomplete cube. Moreover, if $r' : D \to X(D)$ is a representation map for D and $r : C \to X(C)$ extends r' by setting $r(c) = \mathrm{supp}(s^{-1}(c))$ for each $c \in C \setminus D$, then r is a representation map for C.*

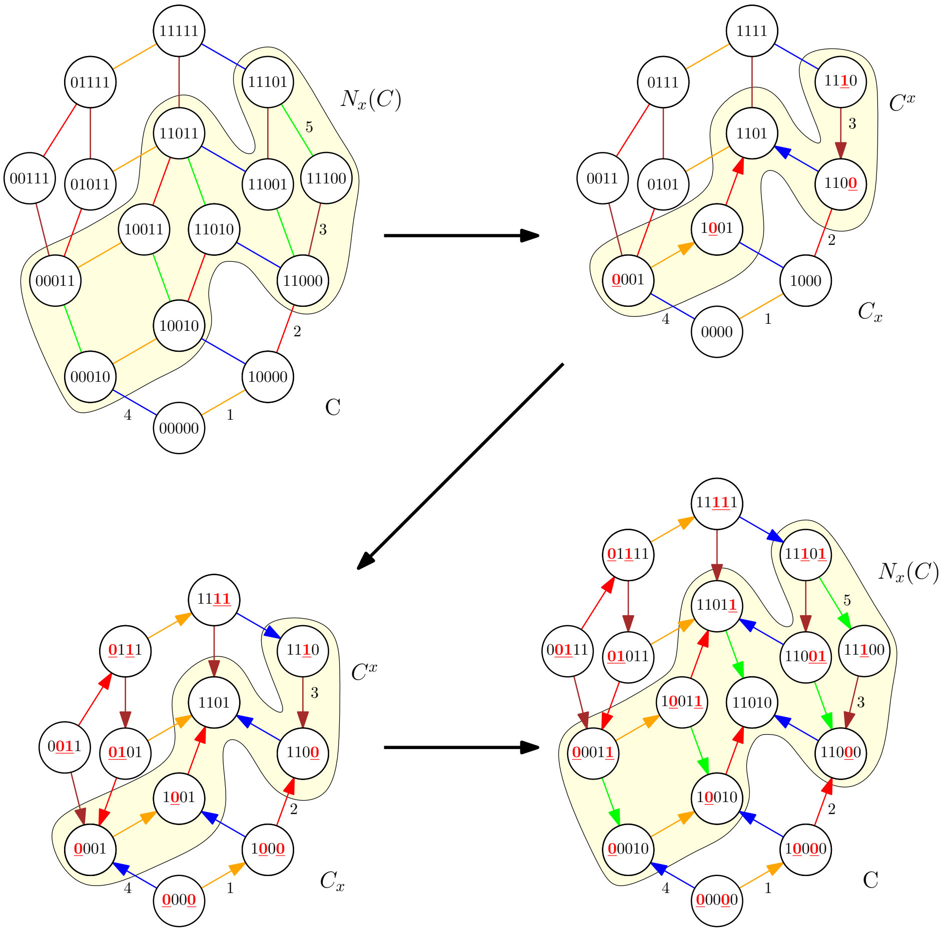


FIGURE 4.3. Illustrating the proof of Theorem 4.2 (when $x = 5$): to construct a representation map for C, we inductively construct a representation map r^x for C^x, extend it to a representation map r_x for C_x using Proposition 4.14 with $D = C^x$, and finally extend it to a representation map r for C. The representation maps r^x, r_x, and r are defined by the orientation as in Theorem 4.20 and by the coordinates of the underlined bits.

Proof of Theorem 4.2. Following the general idea of [**155**], we derive a representation map for C by induction on $|U|$. For the induction step (see Figure 4.3), pick $x \in U$ and consider the maximum classes C_x and $C^x \subset C_x$ with domain $U \setminus \{x\}$. By induction, C^x has a representation map r^x. Use Proposition 4.14 to extend r^x to a representation map r_x of C_x. Define a map r on C as follows:

- $r(c) = r_x(c_x)$ if $c_x \notin C^x$ or $x \notin c$,
- $r(c) = r_x(c_x) \cup \{x\}$ if $c_x \in C^x$ and $x \in c$.

It is easy to verify that r is non-clashing: indeed, if $c' \neq c'' \in C$ satisfy $c'_x \neq c''_x$ then $c'_x|r_x(c'_x) \cup r_x(c''_x) \neq c''_x|r_x(c'_x) \cup r_x(c''_x)$. Since $r_x(c'_x) \subseteq r(c'), r_x(c''_x) \subseteq r(c'')$, it follows that also c', c'' disagree on $r(c') \cup r(c'')$. Else, $c'_x = c''_x \in C^x$ and $c'(x) \neq c''(x)$. In this case, $x \in r(c') \cup r(c'')$ and therefore c', c'' disagree on $r(c') \cup r(c'')$.

It remains to show that r is a bijection between C and $X(C) = \binom{U}{\leq d}$. It is easy to verify that r is injective. So, it remains to show that $|r(c)| \leq d$, for every $c \in C$. This is clear when $c_x \notin C^x$ or $x \notin c$. If $c_x \in C^x$ and $x \in c$, then $r(c) = r^x(c_x) \cup \{x\}$ and $|r^x(c_x)| \leq d-1$ (since C^x is $(d-1)$-dimensional). Hence, $|r(c)| \leq d$ as required, concluding the proof. □

Proof of Proposition 4.14. Call a maximal cube of C a *chamber* and a facet of a chamber a *panel* (a σ'-*panel* if its support is σ'). Any σ'-panel in C satisfies $|\sigma'| = d-1$ and $\sigma' \in X(D)$. Recall that a gallery between two parallel cubes Q', Q'' (say, two σ'-cubes) is any simple path of σ'-cubes $(Q_0 := Q', Q_1, \ldots, Q_k := Q'')$, where $Q_i \cup Q_{i+1}$ is a d-cube. By Theorem 4.5(3), any two parallel cubes of C are connected by a gallery in C. Since D is a maximum class, any panel of C is parallel to a panel that is a maximal cube of D. Also for any maximal simplex $\sigma' \in X(D)$, the class $C^{\sigma'}$ is a maximum class of dimension 1 and $D^{\sigma'}$ is a maximum class of dimension 0 (single concept). Thus $C^{\sigma'}$ is a tree (e.g. [**123**, Lemma 7]) which contains the unique concept $c \in D^{\sigma'}$. We call c *the root* of $C^{\sigma'}$ and denote the σ'-panel P such that $P^{\sigma'} = c$ by $P(\sigma')$.

Lemma 4.15. *Let Q be an incomplete cube for (C, D) with source s and support σ, and let $x, y \in U$ such that $x \notin \sigma$ and $y \in \sigma$. Then, the following holds:*

(i) Q_x is an incomplete cube for (C_x, D_x) whose source is s_x.
(ii) Q^y is an incomplete cube for (C^y, D^y) whose source is s^y.

Next we prove that each concept of $C \setminus D$ is the source of a unique incomplete cube. Assume the contrary and let (C, D) be a counterexample minimizing the size of U. First, if a concept $c \in C \setminus D$ is the source of two incomplete cubes Q_1, Q_2, then $\mathrm{dom}(C) = \mathrm{supp}(Q_1) \cupdot \mathrm{supp}(Q_2)$. Indeed, let $\sigma_1 = \mathrm{supp}(Q_1)$ and $\sigma_2 = \mathrm{supp}(Q_2)$. By Lemma 4.15(i) and minimality of (C, D), $\mathrm{dom}(C) = \sigma_1 \cup \sigma_2$. By Lemma 4.15(ii) and minimality of (C, D), $\sigma_1 \cap \sigma_2 = \varnothing$. Indeed, if there exists x in $\sigma_1 \cap \sigma_2$, c^x is the source of the incomplete cubes Q_1^x and Q_2^x for (C^x, D^x), contrary to minimality of (C, D).

Next we assert that any $c \in C \setminus D$ is the source of at most 2 incomplete cubes. Indeed, let c be the source of incomplete cubes Q_1, Q_2, Q_3. Then $\mathrm{dom}(C) = \mathrm{supp}(Q_1) \cupdot \mathrm{supp}(Q_2)$, i.e., $\mathrm{supp}(Q_2) = \mathrm{dom}(C) \setminus \mathrm{supp}(Q_1)$. For similar reasons, $\mathrm{supp}(Q_3) = \mathrm{dom}(C) \setminus \mathrm{supp}(Q_1) = \mathrm{supp}(Q_2)$. Thus, by Corollary 4.6, $Q_2 = Q_3$.

Lemma 4.16. *Let $c', c'' \in C \setminus D$ be neighbors and let $c' \Delta c'' = \{x\}$. Then, c' is the source of 2 incomplete cubes if and only if c'' is the source of 0 incomplete cubes. Consequently, every connected component in $G(C \setminus D)$ either contains only concepts c with $|s^{-1}(c)| \in \{0, 2\}$, or only concepts c with $|s^{-1}(c)| = 1$.*

Pick $c \in C \setminus D$ that is the source of two incomplete cubes for (C, D) and an incomplete cube Q such that $c = s(Q)$. Let $\sigma = \mathrm{supp}(Q)$, $x \in \sigma$, and $\sigma' = \sigma \setminus \{x\}$. The concept c belongs to a unique σ'-panel P. Let $L = (P_0 = P(\sigma'), P_1, \ldots, P_{m-1}, P_m = P)$ be the unique gallery between the root $P(\sigma')$ of the tree C^σ and P. For $i = 1, \ldots, m$, denote

the chamber $P_{i-1} \cup P_i$ by Q_i. Since $P_i \cap D$ and $Q_i \cap D$ are ample for $i \geq 0$, and P_i is not contained in D for $i > 0$, it follows that the complements $P_i \setminus D$ and $Q_i \setminus D$ are nonempty ample classes. Hence $P_i \setminus D$ and $Q_i \setminus D$ induce nonempty connected subgraphs of $G(C \setminus D)$. Therefore, it follows that c and each concept $c' \in Q_i \setminus D$ are connected in $G(C \setminus D)$ by a path for $i > 0$, and by Lemma 4.16 it follows that
(4.1)
For every $i > 0$, each $c' \in Q_i \setminus D$ is the source of either 0 or 2 incomplete cubes.

Consider the chamber $Q_1 = P_0 \cup P_1$ and its source $s = s(Q_1)$. By the definition of the source, necessarily $s \in P_1$ and $s \notin D$. Therefore, Equation (4.1) implies that there must exist another cube Q' such that $s = s(Q')$. Let s' be the neighbor of s in $P_0 = P(\sigma')$; note that $s' \in D$. Since $\operatorname{supp}(Q_1) \cap \operatorname{supp}(Q') = \varnothing$, it follows that $s|\operatorname{supp}(Q') = s'|\operatorname{supp}(Q') \in D|\operatorname{supp}(Q')$, contradicting that $s = s(Q')$. This establishes the first assertion of Proposition 4.14.

We prove now that the map r defined in Proposition 4.14 is a representation map for C. It is easy to verify that it is a bijection between C and $X(C)$, so it remain to establish the non-clashing property: $c|(r(c) \cup r(c')) \neq c'|(r(c) \cup r(c'))$ for all distinct pairs $c, c' \in C$. This holds when $c, c' \in D$ because r' is a representation map. Next, if $c \in C \setminus D$ and $c' \in D$, this holds because $c|r(c) \notin D|r(c)$ by the properties of s. Thus, it remains to show that every distinct $c, c' \in C \setminus D$ satisfy $c|(\operatorname{supp}(Q) \cup \operatorname{supp}(Q')) \neq c'|(\operatorname{supp}(Q) \cup \operatorname{supp}(Q'))$, where $Q = s^{-1}(c), Q' = s^{-1}(c')$. Assume towards contradiction that this does not hold and consider a counterexample with minimal domain size $|U|$. By minimality, $\operatorname{supp}(Q') \cup \operatorname{supp}(Q) = U$ (or else (C_x, D_x), for some $x \notin \operatorname{supp}(Q') \cup \operatorname{supp}(Q)$ would be a smaller counterexample). Therefore, since c, c' are distinct, there must be $x \in U = \operatorname{supp}(Q') \cup \operatorname{supp}(Q)$ such that $c(x) \neq c'(x)$, which is a contradiction. This ends the proof of Proposition 4.14. ☐

5. Representation Maps for Ample Classes

In this section, we provide combinatorial and geometric characterizations of representation maps of ample classes (which lead to optimal unlabeled compression schemes).

THEOREM 4.17. *Let $C \subseteq 2^U$ be an ample class and let $r : C \to X(C)$ be a bijection. The following conditions are equivalent:*

(R1) $\cup$-non-clashing*: For all distinct concepts $c', c'' \in C$, $c'|r(c') \cup r(c'') \neq c''|r(c') \cup r(c'')$.*
(R2) Reconstruction*: For every realizable sample s of C, there is a unique $c \in C$ that is consistent with s and $r(c) \subseteq \operatorname{dom}(s)$.*
(R3) Cube injective*: For every cube B of 2^U, the mapping $c \mapsto r(c) \cap \operatorname{supp}(B)$ from $C \cap B$ to $X(C \cap B)$ is injective.*
(R4) Δ-non-clashing*: For all distinct concepts $c', c'' \in C$, $c'|r(c')\Delta r(c'') \neq c''|r(c')\Delta r(c'')$.*

Moreover, any Δ-non-clashing map $r : C \to X(C)$ is bijective and is therefore a representation map. Furthermore, if r is a representation map for C, then there exists an unlabeled sample compression scheme for C of size VC-dim(C).

PROOF. Fix $Y \subseteq U$ and partition C into equivalence classes where two concepts c, c' are equivalent if $c|Y = c'|Y$. Thus, each equivalence class corresponds to a sample of C with domain Y, i.e., a concept in $C|Y$. We first show that the number of such equivalence classes equals the number of concepts whose representation set is contained in Y:

$$\begin{aligned} |C|Y| &= |\overline{X}(C|Y)| && \text{(Since } C|Y \text{ is ample)} \\ &= |\overline{X}(C) \cap 2^Y| \\ &= |\{c : r(c) \subseteq Y\}| && \text{(Since } r : C \to \overline{X}(C) = \underline{X}(C) \text{ is a bijection)} \end{aligned}$$

Condition (R2) asserts that in each equivalence class there is exactly one concept c such that $r(c) \subseteq Y$.

(R1) $\Rightarrow$ (R2): Assume $\neg(R2)$ and consider a sample s for which the property does not hold. This implies that there exists an equivalence class with two distinct concepts $c', c'' \in C$ for which $r(c'), r(c'') \subseteq Y$. Therefore, there exist two equivalent concepts $c, c' \in C$ such that $r(c), r(c') \subseteq Y$. Since $c|Y = c'|Y$, we have $c|r(c) \cup r(c') = c'|r(c) \cup r(c')$, contradicting $(R1)$.

(R2) $\Rightarrow$ (R1): Assume $\neg(R1)$, i.e. for two distinct concepts $c', c'' \in C$, we have $c'|r(c') \cup r(c'') = c''|r(c') \cup r(c'')$. Now for the sample $s = c'|r(c') \cup r(c'')$, we have $\mathrm{dom}(s) = r(c') \cup r(c'')$. Furthermore, $c'|\,\mathrm{dom}(s) = c''|\,\mathrm{dom}(s)$ and $r(c'), r(c'') \subseteq \mathrm{dom}(s)$, contradicting $(R2)$.

(R1)&(R2) $\Rightarrow$ (R3): Since $C \cap B$ is ample, it suffices to show that for every $Y \in X(C \cap B)$ there is some $c \in C \cap B$ with $r(c) \cap \mathrm{supp}(B) = Y$. This is established by the following fundamental claim that is proved by induction on $|Y|$.

Claim 4.18. *Conditions (R1) and (R2) together imply that for any $Y \in X(C \cap B)$, there exists a unique concept $c_Y \in C \cap B$ such that $r(c_Y) \cap \mathrm{supp}(B) = Y$.*

(R3) $\Rightarrow$ (R4): For any distinct concepts $c', c'' \in C$, consider the minimal cube $B := B(c', c'')$ which contains both c', c''. This means that $c'(x) \neq c''(x)$ for every $x \in \mathrm{supp}(B)$, and that $c'(x) = c''(x)$ for every $x \notin \mathrm{supp}(B)$. Condition (R3) guarantees that the mapping $r(c) \mapsto r(c) \cap \mathrm{supp}(B)$ is an injection from $C \cap B$ to $X(C \cap B)$. Therefore $r(c') \cap \mathrm{supp}(B) \neq r(c'') \cap \mathrm{supp}(B)$. It follows that there must be some $x \in \mathrm{supp}(B)$ such that $x \in (r(c') \cap \mathrm{supp}(B)) \Delta (r(c'') \cap \mathrm{supp}(B))$. Since $x \in \mathrm{supp}(B)$, $c'(x) \neq c''(x)$ and therefore $c'|r(c') \Delta r(c'') \neq c''|r(c') \Delta r(c'')$ and condition (R4) holds for c' and c''.

(R4) $\Rightarrow$ (R1): This is immediate because if two concepts clash on their symmetric difference, then they also clash on their union.

Moreover, observe that for any map $r : C \to X(C)$, if $r(c) = r(c')$ for $c \neq c'$, then $r(c) \Delta r(c') = \varnothing$ and r is not Δ-non-clashing. Consequently, any Δ-non-clashing map $r : C \to X(C)$ is injective and thus bijective since $|C| = |X(C)|$.

We now show that if $r : C \to X(C)$ is a representation map for C then there exists an unlabeled sample compression scheme for C. Let $\mathrm{RS}(C)$ be the set of all samples realizable by C. Formally, an unlabeled sample compression scheme for C of size k is defined by a (compressor) function $\alpha : \mathrm{RS}(C) \to \binom{U}{\leq k}$ and a (reconstructor) function $\beta : \mathrm{Im}(\alpha) := \alpha(\mathrm{RS}(C)) \to C$ such that for any realizable sample s of C, the following conditions hold: $\alpha(s) \subseteq \mathrm{dom}(s)$ and $\beta(\alpha(s))|\,\mathrm{dom}(s) = s$.

By (R2), for each realizable sample $s \in \mathrm{RS}(C)$, let $\gamma(s)$ be the unique concept $c \in C$ such that $r(c) \subseteq \mathrm{dom}(s)$ and $c|\,\mathrm{dom}(s) = s$. Then consider the compressor $\alpha : \mathrm{RS}(C) \to X(C)$ such that for any $s \in \mathrm{RS}(C)$, $\alpha(s) = r(\gamma(s))$ and the reconstructor $\beta : X(C) \to C$ such that for any $Z \in X(C)$, $\beta(Z) = r^{-1}(Z)$. Observe that by the definition of $\gamma(s)$, $\alpha(s) \subseteq \mathrm{dom}(s)$ and $\beta(\alpha(s)) = \gamma(s)$ coincides with s on $\mathrm{dom}(s)$. Consequently, α and β defines an unlabeled sample compression scheme for C of size $\dim(X(C)) = \text{VC-dim}(C)$. This concludes the proof of the theorem. □

Theorem 4.17 implies that for any representation map $r : C \to X(C)$ and any x-edge cc', $r(c) \Delta r(c') = \{x\}$. Hence, r defines an orientation o_r of $G(C)$: an x-edge cc' is oriented from c to c' iff $x \in r(c) \setminus r(c')$. Moreover, o_r has the following properties (the proof relies on Claim 4.18):

Corollary 4.19. *If $r : C \to X(C)$ is a representation map for an ample class $C \subseteq 2^U$, then o_r satisfy the following two conditions:*

(C1) for any $c \in C$, all outgoing neighbors of c belong to a cube of C;

(C2) o_r is a USO on each cube of C.

An orientation o of the edges of $G(C)$ is a *unique sink orientation (USO)* if o satisfies (C1) and (C2). The *out-map* r_o of an orientation o associates to each $c \in C$ the coordinate set of the edges outgoing from c. We continue with a characterization of representation maps of ample classes as out-maps of USOs, extending a similar result of Szabó and Welzl [**236**] for cubes. This characterization is "local-to-global", since (C1) and (C2) are conditions on the *stars* $\mathrm{St}(c)$ of all concepts $c \in C$ ($\mathrm{St}(c)$ is the set of all faces of the cubes containing c).

THEOREM 4.20. *For an ample class C and a map $r : C \to 2^U$, (i)-(iii) are equivalent:*
(i) r is a representation map;
(ii) r is the out-map of a USO;
(iii) $r(c) \in X(C)$ for any $c \in C$ and o_r satisfies (C2).

PROOF. The implication (i)⇒(ii) is established in Corollary 4.19. Now, we prove (ii)⇒(i). Clearly, property (C1) implies that $r(c) \in X(C)$ for any $c \in C$, whence r is a map from C to $X(C)$. Let C be an ample class of smallest size admitting a non-representation map $r : C \to X(C)$ satisfying (C1) and (C2). Hence there exist $u_0, v_0 \in C$ such that $u_0|(r(u_0)\Delta r(v_0)) = v_0|(r(u_0)\Delta r(v_0))$, i.e., $(u_0\Delta v_0) \cap (r(u_0)\Delta r(v_0)) = \varnothing$; (u_0, v_0) is called a *clashing pair*. To establish the following claim, we use the minimality of C.

CLAIM 4.21. *If (u_0, v_0) is a clashing pair, then $C = C \cap B(u_0, v_0)$ and $r(u_0) = r(v_0) = \varnothing$.*

Using Claim 4.21, one can show that C has the following structure:

CLAIM 4.22. *C is a cube minus a vertex.*

By Claim 4.21, $r(u_0) = r(v_0) = \varnothing$. By condition (C1), $r(c) \neq U$ for any $c \in C$. Thus there exists a set $s \in X(C) = 2^U \setminus \{U, \varnothing\}$ such that $s \neq r(c)$ for any $c \in C$. Every s-cube B of C contains a source $p(B)$ for o_{r_B} (i.e., $s \subseteq r(p(B))$). For each s-cube B of C, let $t(B) = r(p(B)) \setminus s$. Notice that $\varnothing \subsetneq t(B) \subsetneq U \setminus s$ since $s \subsetneq r(p(B)) \subsetneq U$. Consequently, there are $2^{|U|-|s|} - 2$ choices for $t(B)$ and since C is a cube minus one vertex by Claim 4.22, there are $2^{|U|-|s|} - 1$ s-cubes in C. Consequently, there exist two s-cubes B, B' such that $t(B) = t(B')$. Thus $\varnothing \subsetneq s \subsetneq r(p(B)) = r(p(B'))$ and $(p(B), p(B'))$ is a clashing pair for C and r, contradicting Claim 4.21. The implication (ii)⇒(iii) is trivial. To prove (iii)⇒(ii), we show by induction on $|U|$ that a map $r : C \to X(C)$ satisfying (C2) also satisfies (C1). □

We continue with some remarks regarding Theorems 4.17 and 4.20. First, corner peelings correspond *exactly* to acyclic USOs.

PROPOSITION 4.23. *An ample class C admits a corner peeling if and only if there exists an acyclic orientation o of the edges of $G(C)$ that is a unique sink orientation.*

We also show that given a representation map for C one can derive representations maps for intersections of C with cubes, reductions C^Y, and restrictions C_Y.

Furthermore, there exist a bijection $r' : C \to X(C)$ satisfying (C1) and an injection $r'' : C \to 2^U$ satisfying (C2). Nevertheless, we were not able to find a map satisfying (C1) and (C2). It is surprising that, while each d-cube has at least $d^{\Omega(2^d)}$ USOs [**168**], it is so difficult to find a single USO for ample classes. One can try to find such maps by extending the approach for maximum classes: given ample classes C and D with $D \subset C$, a representation map r for C is called *D-entering* if all edges cd with $c \in C \setminus D$ and $d \in D$ are directed by o_r from c to d. The representation map defined in Proposition 4.14 is D-entering. Given $x \in \mathrm{dom}(C)$, suppose that r_x is a C^x-entering representation map for C_x. We can extend the orientation o_{r_x} to an orientation o of $G(C)$ as follows. Each

x-edge cc' of $G(C)$ is directed arbitrarily, while each other edge cc' is directed as the edge $c_x c'_x$ is directed by o_{r_x}. Since o_{r_x} satisfies (C1), (C2) and r_x is C^x-entering, o also satisfies (C1), (C2), thus the map r_o is a representation map for C. So, ample classes would admit representation maps, if *for any ample classes $D \subseteq C$, any representation map r' of D extends to a D-entering representation map r of C.*

6. Conclusion

Even if all maximum classes do not have a corner, we have shown that they always admit representation maps and thus unlabeled sample compression schemes of optimal size. The main open problem is now to extend this result to ample classes.

CONJECTURE 4.24. *Any ample class C admits a representation map.*

We believe that the local-to-global results established in Theorem 4.20 can be useful to establish such a result. Since ample classes of VC-dimension 2 always have corners, they also have representation maps. A first step would be to consider ample classes of VC-dimension 3. Note that the counterexample presented in Section 3 has VC-dimension 3, has no corner but admits a representation map.

One structural/combinatorial approach to the general sample compression conjecture is to cover any concept class C of VC-dimension d by one or a few maximum or ample classes of VC-dimension $O(d)$. Namely, Rubinstein et al. [**217**] asked if any concept class of VC-dimension d can be extended to a maximum class of VC-dimension $O(d)$. Moran and Warmuth [**174**] asked if any concept class of VC-dimension d can be covered by $O(2^d)$ ample sets of VC-dimension d. However both questions are already open for classes of VC-dimension 2.

For ample classes the VC-dimension coincides with the usual topological dimension. Other set systems for which the VC-dimension is a well-defined parameter are the set families defined by the topes of oriented matroids (OMs) [**30**], and more generally, the topes of Complexes of Oriented Matroids (COMs) [**20**] (that generalize ample classes). In this case, the VC-dimension equals the rank. Other structured set systems are bases of (non-oriented) matroids. It will be interesting to know if these structures admit representation maps or unique sink orientations, and to investigate the sample compression conjecture for these set systems.

Part 2

Using Coverings for Distributed Algorithms

CHAPTER 5

Graph Exploration with Binoculars

Since the seminal work of Angluin [**9**], covers of graphs have been used in distributed computing in order to express indistinguishability between processes in a network and establish impossibility results in distributed computing. When considering a network of processes communicating by exchanging messages, if the network is anonymous (i.e., processes do not have unique identifiers), one cannot always break the initial symmetry. For example, in a synchronous ring of identical processes, for any deterministic algorithm, all processes will always remain in the same state and leader election is impossible in this case. If the processes are not provided any initial information on the network, the processes will not even be able to compute the size of the network (or a bound of its size). Describing what distributed tasks are computable in anonymous networks in the message passing model has attracted the attention of many authors [**31**–**34**, **49**, **73**, **77**, **126**, **127**, **170**, **259**, **260**].

One important tool to study computability in anonymous networks is the notion of views [**34**, **259**]. The view of a vertex v in a graph G is a rooteed tree where each vertex corresponds to a path of G starting at v (see Section 3.1). As observed by Yamashita and Kameda [**259**] two nodes that have the same view will behave similarly during the synchronous execution of any deterministic message passing algorithm. This leads to two kinds of impossibility results. First, if two vertices in a network G have the same view, then it is impossible to deterministically distinguish them and thus there is no deterministic election algorithm for G. Second, if a vertex v in a network G and a vertex v' in a network G' have the same view, then the nodes cannot decide whether the underlying graph is G or G' if they are not given any initial information about the network they are in. The first result is in the same vein as the impossibility result of Angluin stating that one cannot elect in a graph G if G is a cover of a smaller graph H [**9**]. In fact, when we forget the root of the view of a vertex v in a graph G, we get exactly the universal cover of G. The second result can be used to establish impossibility results for family of graphs in the same vein as the following result of Angluin [**9**]: there is no universal election algorithm for any family of graphs that contain strictly the family of trees.

In this document, we focus on what can be computed by a unique agent evolving in a network (see Section 1 for more details). Mobile agents are computational units that can progress autonomously from place to place within an environment, interacting with the environment at each node that it is located on. Such software robots (sometimes called bots, or agents) are already prevalent in the Internet, and are used for performing a variety of tasks such as collecting information or negotiating a business deal. More generally, when the data is physically dispersed, it can be sometimes beneficial to move the computation to the data, instead of moving all the data to the entity performing the computation. The paradigm of mobile agent computing / distributed robotics is based on this idea. As underlined in [**98**], the use of mobile agents has been advocated for numerous reasons such as robustness against network disruptions, improving the latency and reducing network load, providing more autonomy and reducing the design complexity, and so on (see e.g. [**157**]). Autonomous mobile robots (or agents) are used for various tasks like cleaning, guarding, data retrieval, etc. in unknown environments.

Many tasks require coordination of the agents [**3**, **161**, **162**, **235**] and exploration of the environment [**99**, **148**, **197**, **234**].

For many distributed problems with mobile agents, exploring, that is visiting every location of the whole environment, is an important prerequisite. In its thorough exposition about Exploration by mobile agents [**98**], Das presents numerous variations of the problem. In particular, it can be noted that, given some global information about the environment (like its size or a bound on the diameter), it is always possible to explore, even in environments where there is no local information that enables to know, arriving on a node, whether it has already been visited (e.g. anonymous networks). If no global information is given to the agent, then the only way to perform a network traversal is to use a *unlimited* traversal (e.g. with a classical BFS or Universal Exploration Sequences [**6**, **152**, **210**] with increasing parameters). This infinite process is sometimes called *Perpetual Exploration* when the agent visits infinitely many times every node. Perpetual Exploration has application mainly to security and safety when the mobile agents are a way to regularly check that the environment is safe. But it is important to note that in the case where no global information is available, it is impossible to always detect when the Exploration has been completed. This is problematic when one would like to use the Exploration algorithm composed with another distributed algorithm.

In this chapter, we focus on Exploration with termination when the agent has no initial information about the network. We say that an algorithm $\mathcal{A}$ is an *Exploration algorithm* (see Section 2) if for any network and any starting position, either the agent visits every vertex or the agent never halts. A network G is *explorable* if there is an Exploration algorithm $\mathcal{A}$ that halts on G (after it has visited every vertex).

One can think that this model is not a distributed model since there is only one agent and we do not have to handle the cooperation between different computing entities. However, since the agent cannot a priori recognize whether it has already visited a node when it reaches it, there is some spatial uncertainty to handle. It turns out that the tools introduced to study anonymous message-passing systems are the right tools to study what can be computed in this model.

Main Results. We first recall that in the classical mobile agent model (see Section 1), the only networks that can be explored without any initial information are the trees. One can also express trees in terms of graphs that are "maximum" for covers of graphs (see Section 3 for the definitions).

THEOREM 5.1 (Folklore). *For a finite simple graph G, the following are equivalent:*
(1) G is explorable,
(2) G has no non-trivial cover, i.e., the universal cover $\widetilde{G}$ of G is isomorphic to G,
(3) G has no infinite cover,
(4) G is a tree.

We then introduce our model where the agent is given *binoculars* of range 1: when on a vertex v, it can see the graph induced by $B_G(v, 1)$. The counterpart of Theorem 5.1 is the following theorem.

THEOREM 5.2 ([**75**]). *A finite simple graph G is explorable with binoculars if and only if the clique complex $X(G)$ has no infinite cover.*

In order to prove that if $X(G)$ has an infinite cover (i.e., its universal cover is infinite) then G is not explorable, we use an adaptation of the classical Lifting lemma of Angluin [**9**]. However, the methods to show the other implication are very different from the classical ones. When dealing with trees, a simple Depth-First-Search strategy leads to an optimal algorithm. In our case, the class is much larger and the algorithm is much more involved. We present a universal Exploration algorithm that explores all explorable graphs (and never stop on any other graph), but the complexity of our algorithm is unbounded. In fact, we show that there is no universal Exploration algorithm such that one can bound its complexity by a computable function.

THEOREM 5.3 ([**75**]). *Consider any algorithm $\mathcal{A}$ using binoculars that explores every explorable graph G. For any computable function $\mathtt{t} : \mathbb{N} \to \mathbb{N}$, there exists an explorable graph G such that when executed on G, $\mathcal{A}$ executes strictly more than $\mathtt{t}(|V(G)|)$ steps.*

We then consider a subclass of explorable graphs, that we call the Weetman graphs (see Section 7). This class of graphs is rather large as it contains chordal graphs, (weakly-)bridged graphs, Helly graphs, Johnson graphs, and prime pre-median graphs. We present an Exploration algorithm that explores all Weetman graphs in a linear number of moves (even if Weetman graphs can contain a quadratic number of edges).

THEOREM 5.4 ([**74**,**76**]). *There is a universal Exploration algorithm using binoculars that explores any Weetman graph G using $O(|V(G)|)$ moves.*

The results of this chapter are based on the papers [**74**–**76**] and have appeared in the PhD thesis of Antoine Naudin [**181**].

1. Model

We consider a standard model of mobile agents. The environment is represented by a simple undirected connected graph $G = (V(G), E(G))$; each vertex $v \in V(G)$ may have a label $\lambda(v)$. The agent starts from a single node of the graph, called the *homebase*. The agent can traverse any edge of the graph incident to its current location. At each node $v \in V(G)$, the edges incident to v are distinguishable to any agent arriving at v. There is a bijective function $\delta_v : N(v) \to \{1, 2, \ldots \deg(v)\}$ which assigns unique labels (port-numbers) to the edges incident at node v (where $\deg(v)$ is the degree of v). We denote by (G, δ) the graph G endowed with a port numbering $\delta = \{\delta_v\}_{v \in V(G)}$. By abuse of notation, since the port numbering is usually fixed, we denote by G a graph (G, δ).

An agent at a node v can choose to go to any adjacent node u by specifying the port number $\delta_v(u)$. On reaching the node u, the agent knows the port number $\delta_u(v)$ of the edge through which it arrived, as well as the degree of u.

For any edge $uv \in E(G)$, we use $\delta(u,v)$ to denote the ordered pair of labels $(\delta_u(v),\delta_v(u))$. A path in G is a sequence of nodes $P = (u_0, u_1, \ldots, u_k)$ such that $u_j u_{j+1} \in E(G)$, $\forall j, 0 \leq j < k$ and the label sequence of path P is $\delta(P) = ((\delta(u_0, u_1), \delta(u_1, u_0)), \ldots, (\delta(u_{k-1}, u_k), \delta(u_k, u_{k-1})))$. Given a label sequence $s = ((p_1, q_1), \ldots, (p_k, q_k))$ and a vertex v_0, if there exists $v_1, \ldots, v_k$ such that $\delta_{v_i}(v_{i+1}) = p_i$ and $\delta_{v_{i+1}}(v_i) = q_i$ for every $0 \leq i < k$, then we denote v_k by $\mathrm{reach}(v_0, s)$. In other words, v_k is the vertex that is reached from v_0 by following the port numbers of s.

There is no global guarantee on the labels of the nodes of G, in particular vertices have no identity (anonymous/homonymous setting), i.e., labels are not guaranteed to be unique. In other words, nodes having the same degree and the same label look identical to the agents. The agent has computing and storage capabilities. When an agent moves from one node to another, it carries with its own local memory. When the agent is located at any node of the graph, it has access to a read-write memory which can be used for local computation (but not for storing information). When an agent arrives at a node, using the degree of the node, the incoming port number and its local memory, the agent computes its next move (i.e., the port number it wants to use next) and updates its local memory.

An execution ρ of an algorithm $\mathcal{A}$ for a mobile agent is composed by a (possibly infinite) sequence of moves by the agent. The length $|\rho|$ of an execution ρ is the total number of moves. The complexity measure we are interested in is the number of edge traversals (or moves) performed by the agent during the execution of the algorithm.

2. Graph Exploration without Information

In this chapter, we consider the *Exploration Problem without Information* for a mobile agent. In this setting, an algorithm $\mathcal{A}$ is an Exploration algorithm if for any graph $G = (V, E)$, for any port numbering δ_G, starting from any arbitrary vertex $v_0 \in V$,

- either the agent visits every vertex at least once and terminates;
- either the agent never halts.[1]

In other words, if the agent halts, then we know that every vertex has been visited. The intuition in this definition is to model the absence of global knowledge while maintaining safety of composition. Since we have no access to global information, we might not be able to visit every node on some networks, but, in this case, we do not allow the algorithm to appear as correct by terminating. This allows to safely compose an Exploration algorithm with another algorithm without additional global information.

We say that a graph G is *explorable* if there exists an Exploration algorithm that halts on G starting from any point. An algorithm $\mathcal{A}$ explores $\mathcal{F}$ if it is an Exploration algorithm such that for all $G \in \mathcal{F}$, $\mathcal{A}$ explores and halts. (Note that since $\mathcal{A}$ is an Exploration algorithm, for any $G \notin \mathcal{F}$, when executed on G from a starting point v, $\mathcal{A}$ either never halts, or $\mathcal{A}$ explores G.)

In the context of distributed computability, a very natural question is to characterize the maximal sets of explorable networks. Observe that because an Exploration algorithm cannot stop on any graph before it has visited all vertices, one can compose several Exploration algorithm: if were are given an Exploration algorithm $\mathcal{A}_1$ for a family $\mathcal{F}_1$ and an Exploration algorithm $\mathcal{A}_2$ for a family $\mathcal{F}_2$, then we can obtain an Exploration algorithm for $\mathcal{F}_1 \cup \mathcal{F}_2$ by interleaving the executions of $\mathcal{A}_1$ and $\mathcal{A}_2$. Consequently, there exist a maximum family of explorable graphs without information.

3. Covers of Graphs and Explorable Graphs

It is well-known that without information, the only explorable graphs are the trees. Moreover, trees can be explored using a Depth-First-Search strategy with a linear number of moves. In this section, we explain how one can use covers to obtain such a result.

3.1. Covers in the case of Graphs. If we consider a graph as a 1-dimensional simplicial complex, the notion of covers defined in Chapter 1 coincides with the standard notion of covers used in distributed computing since the seminal work of Angluin [**9**]. Given two simple graphs G, H, the graph G is a *cover* of a graph H via a covering map $\varphi : V(G) \to V(H)$ if φ is a *locally bijective* homomorphism, i.e., if for every $v \in V(G)$, φ induces a bijection between $N_G(v)$ and $N_H(\varphi(v))$. We say that G is a *non-trivial* cover of H if G is a cover of H, and G and H are not isomorphic. As usual, when considering labeled graphs (with labels on nodes and port-numbers), we only consider covering maps that preserve the labels.

Note that covers of graphs are much easier to understand to understand and manipulate than covers of simplicial complexes. For example, the result of Reidemeister [**209**] enables to construct all covers of size qn of a graph with n vertices for any integer q. Such a result does not exist for covers of simplicial complexes. One can easily show that for a graph H, there exists a non-trivial cover G of H if and only if H is a tree. On the other hand, one cannot algorithmically decide whether a simplicial complex X admits a non-trivial cover [**134**].

When dealing with covers, the universal cover of a graph G is always a tree: either G is a tree and its universal cover $\widetilde{G}$ is isomorphic to G, or G is not a tree and its universal

[1] a seemingly stronger definition could require that the agent performs perpetual exploration in this case. It is easy to see that this is actually equivalent for computability considerations since it is always possible to compose in parallel (see below) a perpetual BFS to any never halting algorithm.

cover $\widetilde{G}$ is an infinite tree. The notion of universal cover is very close to the notion of *view* used in distributed computing [**33**, **259**].

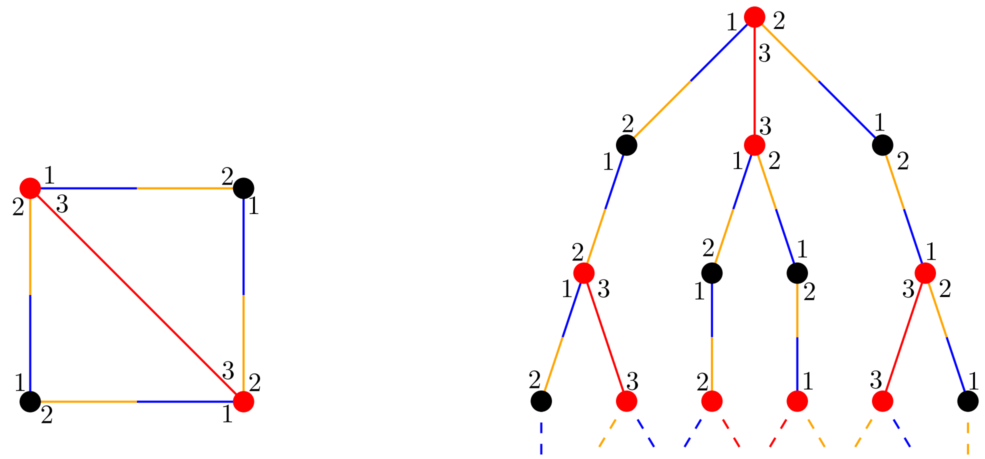


FIGURE 5.1. A graph G with a port-numbering δ and the first levels of the view $\mathcal{T}_G(v)$ of a vertex v of degree 3 in G. The view $\mathcal{T}_G(v)$ is an infinite tree that is isomorphic to the universal cover $\widetilde{G}$ of G.

The *view* $\mathcal{T}_G(v)$ of a vertex v in a graph G is an infinite tree where all vertices are the non-stuttering[2] paths starting in v_0 (See Figure 5.1 for an example). The label of this vertex $t = (v = v_0, v_1, \ldots, v_{p-1}, v_p)$ is $\lambda(t) = \lambda(v_p)$. For every such vertex $t = (v = v_0, v_1, \ldots, v_{p-1}, v_p)$ with $p > 0$, there is an edge from t to $t' = (v = v_0, v_1, \ldots, v_{p-1})$ in $\mathcal{T}_G(v)$ with $\delta_t(t, t') = \delta_{v_p}(v_p, v_{p-1})$ and $\delta_t(t', t) = \delta_{v_{p-1}}(v_{p-1}, v_p)$. It is straightforward to see that $\mathcal{T}_G(v)$ is a cover of G via a covering map that maps the root (v) of $\mathcal{T}_G(v)$ to v. Moreover, one can easily show that if K is a cover of G via a covering map φ, then $\mathcal{T}_K(v) = \mathcal{T}_G(\varphi(v))$ for any vertex $v \in V(K)$. Consequently, $\mathcal{T}_G(v)$ is the universal cover of G and is thus independent of the choice of v. However, it may be sometimes useful to remember the vertex v. In fact, the view $\mathcal{T}_G(v)$ of v in G can be seen as the universal $\widetilde{G}$ cover of G pointed at some vertex $\widetilde{v}$ that is mapped to v by the covering map. Since we are interested in algorithms that stop, an agent evolving in a graph G cannot a priori compute the view $\mathcal{T}_G(v_0)$ of its starting vertex v_0 since it can be infinite. The view $\mathcal{T}_G(v_0, k)$ of *depth* k of v_0 in G is the finite subtree of $\mathcal{T}_G(v_0)$ containing all vertices at distance at most k from the root of $\mathcal{T}_G(v_0)$. The nodes of $\mathcal{T}_G(v_0, k)$ correspond exactly to the non-stuttering paths starting in v_0 of length at most k. For an integer k, two graphs G, G' and two vertices $v \in V(G)$, $v' \in V(G')$, if $\mathcal{T}_G(v, k) = \mathcal{T}_{G'}(v', k)$, we say that v and v' are k-equivalent and we denote it by $v \equiv_k v'$.

REMARK 5.5. An agent starting on a graph G at a vertex v_0 can compute its view $\mathcal{T}_G(v_0, k)$ of depth k by performing a number of moves that is linear in the size of $\mathcal{T}_G(v_0, k)$. Note however that the number of vertices of $\mathcal{T}_G(v_0, k)$ can be exponential in k

When dealing with finite graphs (or complexes), the size of the preimages of vertices (or edges) are the same. The following proposition can be proved easily since we only consider connected graphs.

PROPOSITION 5.6 ([**209**]). *Given two finite simple graphs G, H such that G is a cover of H via φ, there exists $q \in \mathbb{N}$ such that for every $x \in V(H) \cup E(H)$, $\varphi^{-1}(v) = q$.*
This number q is called the number of sheets *of the cover.*

[2] A path $(v = v_0, v_1, \ldots, v_{p-1}, v_p)$ is non-stuttering if $v_{i-1} \neq v_{i+1}$ for every $0 < i < p$.

3.2. Explorable Graphs. Consider an algorithm $\mathcal{A}$ and an execution of $\mathcal{A}$ performed by a mobile agent starting on a vertex v in a network G. For any $i \in \mathbb{N}$, we denote respectively the position of the agent and its state (i.e., the content of its memory) at step i by $\mathtt{pos}_i(\mathcal{A}, G, v)$ and $\mathtt{mem}_i(\mathcal{A}, G, v)$. By standard techniques (see [**9**, **33**, **73**, **259**]), we have the following lemma.

Lemma 5.7 (Lifting Lemma). *Consider two graphs G and G' such that there exists a covering map $\varphi : G' \to G$. For any algorithm $\mathcal{A}$ and for any vertices $v \in V(G)$ and $v' \in V(G')$ such that $\varphi(v') = v$, for any step $i \in \mathbb{N}$, $\mathtt{mem}_i(\mathcal{A}, G', v') = \mathtt{mem}_i(\mathcal{A}, G, v)$ and $\varphi(\mathtt{pos}_i(\mathcal{A}, G', v')) = \mathtt{pos}_i(\mathcal{A}, G, v)$.*

Note that in particular, an agent evolving in a graph G behaves as if it was evolving in the view $\mathcal{T}_G(v_0)$ where v_0 is its starting position in v_0. Using this observation, we can establish the following theorem.

Theorem 5.1 (Folklore). *For a finite simple graph G, the following are equivalent:*
(1) G is explorable,
(2) G has no non-trivial cover, i.e., the universal cover $\widetilde{G}$ of G is isomorphic to G,
(3) G has no infinite cover,
(4) G is a tree.

Proof. If G is a tree, then its universal cover $\widetilde{G}$ is isomorphic to G and thus G has no infinite cover. Conversely, if G has no infinite cover, then $\mathcal{T}_G(v)$ is finite (for any choice of v), and consequently, the universal cover $\widetilde{G}$ of G is isomorphic to $\mathcal{T}_G(v)$ and is a tree. Consequently, $|V(\widetilde{G})| = |E(\widetilde{G})| + 1$. By Proposition 5.6, there exists $q \in \mathbb{N}$ such that $|V(\widetilde{G})| = q|V(G)|$ and $|E(\widetilde{G})| = q|E(G)|$. Since $|V(\widetilde{G})|$ and $|E(\widetilde{G})|$ are coprime, necessarily $q = 1$. Therefore $\widetilde{G}$ is isomorphic to G and G is a tree.

Using a simple depth first search strategy, one can design an algorithm that explores all the trees and stop. On any graph that is not a tree, this algorithm will never stop. This shows that trees are explorable. Conversely, consider a graph G that is not a tree and assume that there exists an Exploration algorithm $\mathcal{A}$ for G. Let v_0 be the starting position of the agent and assume that $\mathcal{A}$ stops after k steps. By Lemma 5.7, $\mathcal{A}$ stops after k steps when executed on $\mathcal{T}_G(v_0)$ starting from the root r. Consider the tree $T = B_{k+1}(\mathcal{T}_G(v_0), r)$. Since G is not a tree, $\mathcal{T}_G(v_0)$ is infinite and thus there exists a vertex v at distance $k + 1$ from r in T. Since $\mathcal{A}$ cannot distinguish T from $\mathcal{T}_G(v_0)$ (and thus from G) in k steps, the agent stops in T after k steps before visiting v. Consequently, $\mathcal{A}$ is not an Exploration algorithm, a contradiction. ☐

4. Mobile Agents with Binoculars

In the rest of this chapter, we empower the agent with *binoculars* of range 1: when on a vertex v, the agent can see the graph induced by $B_G(v, 1)$ (as well as the labels and the port numbers on this graph). In order to reuse standard techniques and algorithms, we actually assume that the nodes of the graph we are exploring are labeled by these induced balls. In the following, we assume that every vertex v of G has a label $\nu(v)$ corresponding to the *binoculars labeling* of v. This binoculars label $\nu(v)$ is a graph isomorphic to $B_G(v, 1)$ with its port numbering (See Figure 5.2 for an example). It is straightforward to see that in a graph with such a *binoculars labeling* of the nodes, an agent with binoculars has the same computational power as an agent without binoculars (the "binoculars" primitive gives only access to more information, it does not enable more moves). The following proposition enables to derive impossibility results for mobile agents with binoculars from impossibility results in the standard model.

Proposition 5.8. *For any graphs G, H, (G, ν_G) is a cover of (H, ν_H) if and only if the clique complex $X(G)$ of G is a cover of the clique complex $X(H)$ of H.*

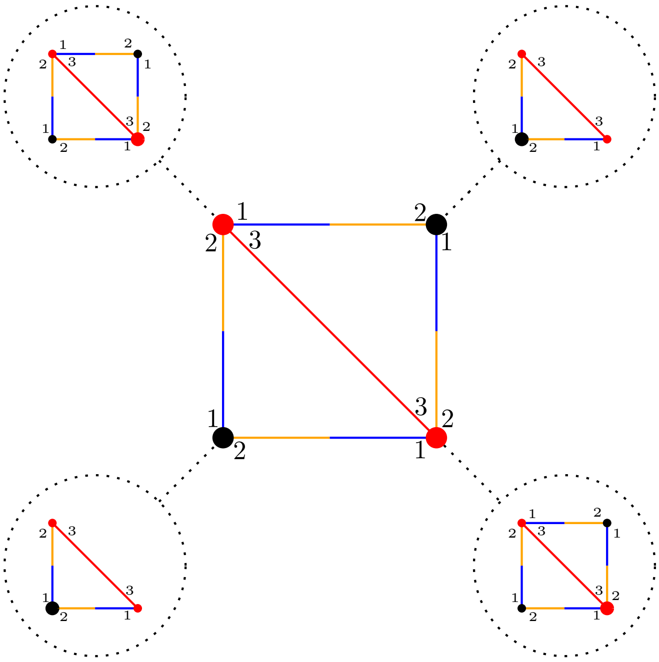


FIGURE 5.2. A graph G with the binocular labeling attached to each vertex. In the label of a vertex v, the bold vertex is the center of $B_G(v, 1)$ and corresponds to v.

Combining Lemma 5.7 and Proposition 5.8, we obtain a Lifting Lemma for covers of clique complexes that is the counterpart of Lemma 5.7 for covers of graphs. This lemma shows that every execution on a graph G can be lifted up to every graph G' such that $X(G')$ is a cover of $X(G)$, and in particular, to the 1-skeleton of the universal cover $\widetilde{X}(G)$ of $X(G)$.

LEMMA 5.9 (Lifting Lemma). *Consider two graphs G and G' such that there exists a covering map $\varphi : X(G') \to X(G)$. For any algorithm $\mathcal{A}$ (using binoculars) and for any vertices $v \in V(G)$ and $v' \in V(G')$ such that $\varphi(v') = v$, for any step $i \in \mathbb{N}$, $\mathit{mem}_i(\mathcal{A}, G', v') = \mathit{mem}_i(\mathcal{A}, G, v)$ and $\varphi(\mathit{pos}_i(\mathcal{A}, G', v')) = \mathit{pos}_i(\mathcal{A}, G, v)$.*

In the following, we will need to distinguish the graphs depending on whether their clique complexes have a finite or infinite universal cover. We define the three following classes:

- $\mathcal{FC} = \{G :$ the universal cover of $X(G)$ is finite$\}$,
- $\mathcal{IC} = \{G : G$ is finite and the universal cover of $X(G)$ is infinite$\}$,
- $\mathcal{SC} = \{G : G$ is finite and $X(G)$ is simply connected$\}$.

Note that $\mathcal{FC} = \{G : \exists K \in \mathcal{SC}, X(K)$ is a cover of $X(G)\} \subsetneq \mathcal{SC}$.

Using Lemma 5.9 above, we are now able to prove a first result about explorable graphs and the move complexity of their exploration.

PROPOSITION 5.10. *Any graph G that is explorable with binoculars belongs to $\mathcal{FC}$, and any Exploration algorithm exploring G performs at least $|V(G')| - 1$ moves, where $G' = \widetilde{X}^1(G)$ is the 1-skeleton of the universal cover $\widetilde{X}(G)$ of the clique complex $X(G)$.*

PROOF. Suppose it is not the case and assume there exists an Exploration algorithm $\mathcal{A}$ that explores a graph $G \in \mathcal{IC}$ when it starts from a vertex $v_0 \in V(G)$. Let r be the number of steps performed by $\mathcal{A}$ on G when it starts on v_0.

Let $G' = \widetilde{X}^1(G)$ be the 1-skeleton of the universal cover of $X(G)$. Consider a covering map $\varphi : \widetilde{X}(G) \to X(G)$ and consider a vertex $v'_0 \in V(G')$ such that $\varphi(v'_0) =$

Algorithm 5.1: $\mathcal{FC}$-Exploration algorithm

$k := 0$;
repeat
  Increment k ;
  Compute $\mathcal{T}_{(G,\nu)}(v_0, 2k)$;
  Find a complex H (if it exists) such that:
  - $|V(H)| < k$, and
  - $\exists \widetilde{v}_0 \in V(H)$ such that $\widetilde{v}_0 \equiv_{2k} v_0$, and
  - every simple cycle of $X(H)$ is k-contractible;

until *H is defined*;
Stop the exploration;

v_0. By Lemma 5.9, when executed on G', $\mathcal{A}$ stops after k steps. Consider the graph $H = B_{G'}(v'_0, k+1)$. Since $G \in \mathcal{IC}$, G' is infinite and $|V(H)| > k+1$. When executed on H starting in v'_0, $\mathcal{A}$ behaves as in G' during at least k steps since the k first moves can only depend of $B_H(v'_0, k) = B_{G'}(v'_0, k)$. Consequently $\mathcal{A}$ stops after k steps when executed on H starting in v'_0. Since $|V(H)| > k+1$, $\mathcal{A}$ stops before it has visited all nodes of H and thus $\mathcal{A}$ is not an Exploration algorithm, a contradiction.

The move complexity bound is obtained from the Lifting Lemma applied to any covering map $\varphi : \widetilde{X}(G) \to X(G)$. Assume we have an Exploration algorithm $\mathcal{A}$ halting on G at some step q. If $|V(G')| > q+1$ then $\mathcal{A}$ halts on G' and has not visited all vertices of G', a contradiction. □

Note that this is the same lifting technique that shows that, without binoculars, tree networks are the only explorable networks without global knowledge.

5. Exploration of $\mathcal{FC}$

We propose in this section an Exploration algorithm for the family $\mathcal{FC}$ in order to prove that this family is the maximum set of explorable networks.

The goal of Algorithm 5.1 is to visit, in a BFS fashion, a ball centered on the homebase of the agent until the radius of the ball is sufficiently large to ensure that G is explored. Once such a radius is reached, the agent stops. To detect when the radius is sufficiently large, we use the view of the homebase (more details below) to search for a simply connected graph which locally looks like the explored ball.

Note that in the following, we will consider the case where each node v is labeled by $\nu(v)$, the graph that is obtained using binoculars from v.

5.1. Presentation of the Algorithm. Consider a graph G and let $v_0 \in V(G)$ be the homebase of the agent in G. Let k be an integer initialized to 1. Algorithm 5.1 is divided in phases. At the beginning of a phase, the agent follows all paths of length at most $2k$ originating from v_0 in order to compute the labeled view $\mathcal{T}(v_0, 2k)$ of v_0.

At the end of the phase, the agent backtracks to its homebase, and enumerates all graphs of size at most k until it finds a graph H such that every simple cycle c of $X(H)$ is k-contractible (i.e., $\text{Area}_\triangle(c) \leq k$) and such that there exists a vertex $\widetilde{v}_0 \in V(H)$ that has the same view at distance $2k$ as v_0, i.e., $\mathcal{T}_H(\widetilde{v}_0, 2k) = \mathcal{T}_G(v_0, 2k)$. If such an H exists then the algorithm stops. Otherwise, k is incremented and the agent starts another phase. Deciding if $\text{Area}_\triangle(c) \leq k$ for a given cycle c is computable (by considering all disk diagrams of area at most k). Since the total number of simple cycles of a graph is finite, Algorithm 5.1 can be implemented on a Turing machine.

5.2. Correction of the algorithm. In order to prove the correction of this algorithm, we prove that when the first graph H satisfying every condition of Algorithm 5.1

is found, then $X(H)$ is actually the universal cover of $X(G)$ (Corollary 5.12). Intuitively, this is because it is not possible to find a simply connected complex that looks locally the same as a *strict subpart* of another complex.

Remember that given a path p in a complex G, $\delta(p)$ denotes the sequence of port numbers followed by p in G. We denote by $\textsc{dest}_G(v_0, \delta(p))$, the vertex in G reached by the path starting in v_0 and labeled by $\delta(p)$. We show (Proposition 5.11) that if we fix a vertex $\widetilde{v}_0 \in V(H)$ such that $\widetilde{v}_0 \equiv_{2k} v_0$, we can define unambiguously a map φ from $V(H)$ to $V(G)$ as follows: for any $\widetilde{u} \in V(H)$, let p be any path from $\widetilde{v}_0$ to $\widetilde{u}$ in H and let $u = \varphi(\widetilde{u})$ be the vertex reached from v_0 in G by the path labeled by $\delta(p)$. The technical part of the proof is the following proposition.

Proposition 5.11. *Consider a graph G such that Algorithm 5.1 stops on G when it starts in v_0. Let $k \in \mathbb{N}$ and let H be the graph computed by the algorithm before it stops. Consider any vertex $\widetilde{v}_0 \in V(H)$ such that $v_0 \equiv_{2k} \widetilde{v}_0$.*

For any vertex $\widetilde{u} \in V(H)$, for any two paths $\widetilde{q}, \widetilde{q}'$ from $\widetilde{v}_0$ to $\widetilde{u}$ in H, $\textsc{dest}_G(v_0, \delta(\widetilde{q})) = \textsc{dest}_G(v_0, \delta(\widetilde{q}'))$.

Showing that φ is a covering, we get the following corollary.

Corollary 5.12. *Consider a graph G such that Algorithm 5.1 stops on G when it starts in $v_0 \in V(G)$ and let H be the graph computed by the algorithm before it stops. The clique complex $X(H)$ is the universal cover of $X(G)$.*

Proof. By the definition of Algorithm 5.1, the complex $X(H)$ is simply connected. Consequently, we just have to show that $X(H)$ is a cover of $X(G)$.

Consider any vertex $\widetilde{v}_0 \in V(H)$ such that $v_0 \equiv_{2k} \widetilde{v}_0$. For any vertex $\widetilde{u} \in V(H)$, consider any path $\widetilde{p_{\widetilde{u}}}$ from $\widetilde{v}_0$ to $\widetilde{u}$ and let $\varphi(\widetilde{u}) = \textsc{dest}_G(v_0, \delta(\widetilde{p_{\widetilde{u}}}))$. From Proposition 5.11, $\varphi(\widetilde{u})$ is independent from our choice of $\widetilde{p_{\widetilde{u}}}$. Since $v_0 \equiv_{2k} \widetilde{v}_0$ and since $|V(H)| \leq k$, for any $\widetilde{u} \in V(H)$, $\nu(\varphi(\widetilde{u})) = \nu(\widetilde{u})$. Consequently, for any $\widetilde{u} \in V(H)$ and for any neighbour $\widetilde{w} \in N_H(\widetilde{u})$, there exists a unique $w \in N_G(\varphi(\widetilde{u}))$ such that $\delta(\widetilde{u}, \widetilde{w}) = \delta(\varphi(\widetilde{u}), w)$. Conversely, for any $w \in N_G(\varphi(\widetilde{u}))$, there exists a unique $\widetilde{w} \in N_H(\widetilde{u})$ such that $\delta(\widetilde{u}, \widetilde{w}) = \delta(\varphi(\widetilde{u}), w)$. In both cases, let $\widetilde{p_{\widetilde{w}}} = \widetilde{p_{\widetilde{u}}} \cdot (\widetilde{u}, \widetilde{w})$; this is a path from $\widetilde{v}_0$ to $\widetilde{w}$. From Proposition 5.11, $\varphi(\widetilde{w}) = \textsc{dest}_G(v_0, \delta(\widetilde{p_{\widetilde{w}}})) = \textsc{dest}_G(u, \delta(\widetilde{u}, \widetilde{w})) = w$. Consequently, φ is a covering map from H to G, and by definition of H, φ also preserves the binoculars labeling. Therefore, the complex $X(H)$ is a cover of the complex $X(G)$. ☐

To finish to prove that Algorithm 5.1 is an Exploration algorithm for $\mathcal{FC}$, we remark that, when considering connected complexes (or graphs), coverings are always surjective. Consequently, G has been explored when the algorithm stops.

Theorem 5.13. *Algorithm 5.1 is an Exploration algorithm for $\mathcal{FC}$.*

Proof. From Corollary 5.12, we know that if Algorithm 5.1 stops, then the clique complex $X(H)$ of the graph H computed by the algorithm is a cover of $X(G)$. Moreover, since $|V(G)| \leq |V(H)| \leq k$ and since the agent has constructed $\mathcal{T}_G(v, 2k)$, it has visited all vertices of G.

We just have to prove that Algorithm 5.1 always halts on any graph $G \in \mathcal{FC}$. Consider any graph $G \in \mathcal{FC}$ and let $G' = \widetilde{X}^1(G)$ be the 1-skeleton of the universal cover $\widetilde{X}(G)$ of $X(G)$. Since $G \in \mathcal{FC}$, G' is finite and there exists $k' \in \mathbb{N}$ such that every simple cycle of G' is k'-contractible. Let $k = \max(|V(G')|, k')$. At phase k, since $X(G') = \widetilde{X}(G)$ is the universal cover of $X(G)$, there exists $v'_0 \in V(G')$ such that $\mathcal{T}_G(v_0) = \mathcal{T}_{G'}(v'_0)$. Consequently, $\mathcal{T}_G(v_0, 2k) = \mathcal{T}_{G'}(v'_0, 2k)$, $|V(G')| \leq k$, and every simple cycle of $X(G')$ is k-contractible. Therefore, at iteration k, the halting condition of Algorithm 5.1 is satisfied. ☐

From Proposition 5.10 and Theorem 5.13 above, we get the following corollary.

Corollary 5.14. *The family $\mathcal{FC}$ is the maximum set of Explorable networks.*

Observe that the family of graphs explorable with binoculars is much larger than the family of trees: it contains planar triangulations, triangulations of the projective plane, as well as many classes studied in metric graph theory such as chordal graphs, (weakly)-bridged graphs, Helly graphs, Johnson graphs, or prime pre-median graphs. Note that all these classes of graphs are contained in $\mathcal{SC}$ except the class of triangulations of the projective plane that are in $\mathcal{FC} \setminus \mathcal{SC}$.

6. Complexity of the Exploration Problem for $\mathcal{SC}$

In the previous section, we did not provide any bound on the number of moves performed by an agent executing our universal exploration algorithm. In this section, we study the complexity of the problem and we show that there does not exist any exploration algorithm for all graphs in $\mathcal{FC}$ such that one can bound the number of moves performed by the agent by a computable function.

The first reason that such a bound cannot exist is rather simple: if the 1-skeleton $G' = \widetilde{X}^1(G)$ of the universal cover of the clique complex of G is finite, then by Lemma 5.9, when executed on G, any exploration algorithm has to perform at least $|V(G')|-1$ steps before it halts. In other words, one can only hope to bound the number of moves performed by an exploration algorithm on a graph G by a function of the size of $\widetilde{X}^1(G)$.

However, in the following theorem, we show that even if we consider only graphs with simply connected clique complexes (i.e., graphs that are isomorphic to their universal covers), there is no Exploration algorithm for this class of graph such that one can bound its complexity by a computable function. Our proof relies on a result of Haken [**134**] that show that it is undecidable to detect whether a finite simplicial complex is simply connected or not.

Theorem 5.15. *Consider any algorithm $\mathcal{A}$ that explores every finite graph $G \in \mathcal{SC}$. For any computable function $\mathtt{t} : \mathbb{N} \to \mathbb{N}$, there exists a graph $G \in \mathcal{SC}$ such that when executed on G, $\mathcal{A}$ executes strictly more than $\mathtt{t}(|V(G)|)$ steps.*

Proof. Suppose this is not true and consider an algorithm $\mathcal{A}$ and a computable function $\mathtt{t} : \mathbb{N} \to \mathbb{N}$ such that for any graph $G \in \mathcal{SC}$, $\mathcal{A}$ visits all the vertices of G and stops in at most $\mathtt{t}(|V(G)|)$ steps. We show that in this case, it is possible to algorithmically decide whether the clique complex of any given graph G is simply connected or not. However, this problem is undecidable [**134**] and thus we get a contradiction[3].

Algorithm 5.2 is an algorithm that takes as an input a graph G and then simulates $\mathcal{A}$ on G for $\mathtt{t}(|V(G)|)$ steps. If $\mathcal{A}$ does not stop within these $\mathtt{t}(|V(G)|)$ steps, then by our assumption on $\mathcal{A}$, we know that $G \notin \mathcal{SC}$ and the algorithm returns no. If $\mathcal{A}$ stops within these $\mathtt{t}(|V(G)|)$ steps, then we check whether there exists a graph H such that $|V(G)| < |V(H)| \leq \mathtt{t}(|V(G)|)$ and such that the clique complex $X(H)$ is a cover of $X(G)$. If such an H exists, then $G \notin \mathcal{SC}$ and the algorithms returns no. If we do not find such an H, the algorithm returns yes.

In order to show Algorithm 5.2 decides simple connectivity, it is sufficient to show that when the algorithm returns yes on a graph G, the clique complex $X(G)$ is simply connected. Suppose it is not the case and let $G' = \widetilde{X}^1(G)$ be the 1-skeleton of the universal cover $\widetilde{X}(G)$ of the clique complex $X(G)$. Consider a covering map φ from $\widetilde{X}(G)$ to $X(G)$ and let $v_0' \in V(G')$ be any vertex such that $\varphi(v_0') = v_0$. By Lemma 5.9, when executed on G' starting in v_0', $\mathcal{A}$ stops after at most $\mathtt{t}(|V(G)|)$ steps.

If G' is finite, then $G' \in \mathcal{SC}$ and by our assumption on $\mathcal{A}$, when executed on G', $\mathcal{A}$ must explore all vertices of G' before it halts. Consequently, $X(G') = \widetilde{X}(G)$ is a cover of

[3] Note that the original result of Haken [**134**] does not assume that the simplicial complexes are clique complexes. However, for any simplicial complex K, the barycentric subdivision K' of K is a clique complex that is simply connected if and only if K is simply connected (see [**137**]).

Algorithm 5.2: An algorithm to check simple connectivity

Input: a graph G

Simulate $\mathcal{A}$ starting from an arbitrary starting vertex v_0 during $\mathtt{t}(|V(G)|)$ steps;
if *$\mathcal{A}$ halts within $\mathtt{t}(|V(G)|)$ steps* **then**
  if *there exists a graph H such that $|V(G)| < |V(H)| \leq \mathtt{t}(|V(G)|)$ and such that the clique complex $X(H)$ is a cover of the clique complex $X(G)$* **then**
    return NO; // $X(G)$ is not simply connected
  else
    return YES; // $X(G)$ is simply connected
else return NO; // $X(G)$ is not simply connected;

$X(G)$ with at most $\mathtt{t}(|V(G)|)$ vertices. Since $X(G)$ is not simply connected, necessarily $|V(G)| < |V(G')|$ and in this case, the algorithm returns NO and we are done.

Assume now that $G' = \widetilde{X}^1(G)$ is infinite. Let $k = \mathtt{t}(|V(G)|)$ and let $B = B_{G'}(v'_0, k)$. Note that when $\mathcal{A}$ is executed on G' starting in v'_0, any node visited by $\mathcal{A}$ belongs to B. Given two vertices, $u', v' \in V(G')$, we say that $u' \equiv_B v'$ if there exists a path from u' to v' in $G' \setminus B$. Observe that $\equiv_B$ is an equivalence relation, and that every vertex of B is the only vertex in its equivalence class. For a vertex $u' \in V(\widetilde{G})$, we denote its equivalence class by $[u']$. Let H be the graph defined by $V(H) = \{[u'] : u' \in V(G')\}$ and $E(H) = \{[u'][v'] : \exists u'' \in [u'], v'' \in [v'], u''v'' \in E(G')\}$.

One can show that the graph H is finite and that its clique complex $X(H)$ is simply connected, i.e., that $H \in \mathcal{SC}$. Since for every $u' \in B$, $[u'] = \{u'\}$, the ball $B_H([v'_0], k)$ is isomorphic to B. Consequently, when $\mathcal{A}$ is executed on H starting in $[v'_0]$, $\mathcal{A}$ stops after at most k steps, and thus before it has visited all vertices of H, contradicting our assumption on $\mathcal{A}$. $\square$

7. An Efficient Exploration Algorithm for Weetman Graphs

Even if there is no hope to find an Exploration algorithm that is efficient for all graphs that are explorable with binoculars (see Section 6), it is possible to design Exploration algorithms that are efficient for subclasses of $\mathcal{FC}$.

In this section, we present an Exploration algorithm for a large class of graphs that we call Weetman graphs.

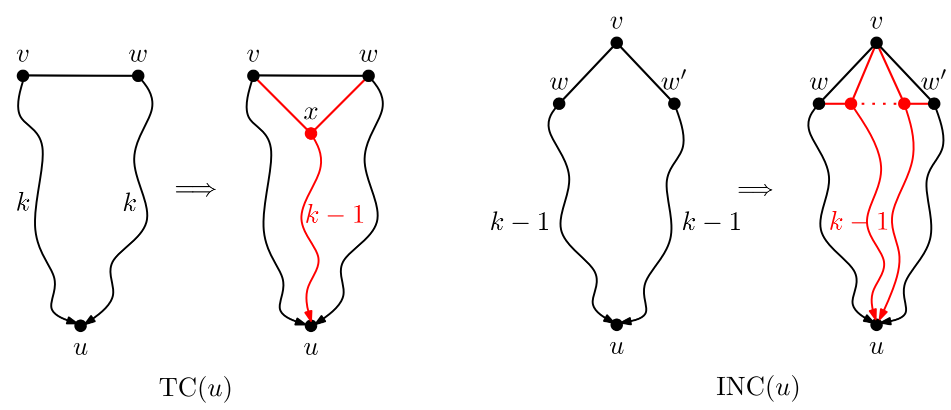


FIGURE 5.3. Triangle and interval-neighborhood conditions

DEFINITION 5.16 (Weetman graphs [**229**, **248**]). A graph G is a *Weetman graph with respect to a vertex u* if its distance function d satisfies the following triangle and interval-neighborhood conditions (see Figure 5.3):

- *Triangle condition* TC(u): for any two vertices v, w with $1 = d(v, w) < d(u, v) = d(u, w)$ there exists a common neighbor x of v and w such that $d(u, x) = d(u, v) - 1$.
- *Interval-Neighborhood condition* INC(u): for any vertex v with $d(u, v) = k \geq 1$, the set of vertices $\{w : w \in N_G(v) \text{ and } d(u, w) = k - 1\}$ induces a connected subgraph of G.

A graph G is called *Weetman* if G is Weetman with respect to any vertex u.

Observe that chordal graphs, (weakly-)bridged graphs, Helly graphs, Johnson graphs and prime pre-median graphs are example of Weetman graphs.

Using the Triangle and the Interval-Neighborhood Conditions, one can show that for any Weetman graph G, its clique complex $X(G)$ is simply connected, i.e., $G \in \mathcal{SC} \subseteq \mathcal{FC}$, and thus Weetman graphs are explorable with binoculars.

Given a vertex v_0 in a graph G, recall that the sphere $S(v_0, k)$ of radius k centered at v_0 is $S(v_0, k) = \{x \in V : d_G(v_0, x) = k\}$, that a connected component Υ of the subgraph of G induced by $V \setminus S(v_0, k-1)$ is called an end of G, and that the vertices of $\Upsilon \cap S(v_0, k)$ form a cluster of G with respect to v_0.

Observe that in a Weetman graph, two vertices u, v at distance k from v_0 are in the same cluster if and only if there exists a path from u to v in $S(v_0, k)$. We define the *cluster-tree* of G with root v_0 as the tree obtained by contracting every cluster of G with respect to v_0. In order to obtain an efficient Exploration algorithm for Weetman graphs, we will use this cluster-tree to guide the exploration of the agent.

7.1. The Exploration Algorithm. The pseudo-code of the algorithm is given in Algorithm 5.3. The general idea of the algorithm is to visit the clusters of G in the order given by a DFS on the cluster tree. While the agent explores a cluster C that is at distance k from its starting point v_0, the agent can discover (using its binoculars) the nodes that are incident to a vertex of V and at distance $k+1$ from v_0. Note that during this exploration of C, the agent visits only the vertices of C. Using the local observations made at each vertex of C and using ideas that are similar to what we did in the universal cover construction in Chapter 1, the agent will construct a map of every cluster C' at distance $k+1$ from v_0 such that C is the parent of C' in the cluster-tree. This map of C' will then be used in order to navigate efficiently in C' when the agent will explore the vertices of C.

However, there is no guarantee that the input graph G is a Weetman graph, or even that the clique complex $X(G)$ is simply connected. During the execution, the agent builds a graph $\widetilde{G}$ in such a way that if the execution terminates, at the end of the execution $X(\widetilde{G})$ is a cover of $X(G)$. Since the clique complexes of Weetman graphs are simply connected, if G is a Weetman graph, then $\widetilde{G}$ will be isomorphic to G at the end of the execution.

In order to obtain local information on the graph G, we assume that a function `getBino` is available to the agent. When called on a vertex $u \in V(G)$, the function `getBino` returns a graph B endowed with a port-labeling δ pointed at a vertex w_0 that is isomorphic to the ball $B_1(u, G)$ pointed at u.

During the execution, the agent builds a graph $\widetilde{G}$ as well as the collection of clusters of $\widetilde{G}$. At each step, the "unexplored" vertices of $\widetilde{G}$ are the vertices of the clusters corresponding to the leaves of the cluster tree of $\widetilde{G}$. Moreover, there exists a map $f : \widetilde{G} \to G$ such that for every $\widetilde{u} \in \widetilde{G}$ that is not in a leaf cluster, f induces an isomorphism between $B_1(\widetilde{u}, \widetilde{G})$ and $B_1(f(\widetilde{u}), G)$. During the algorithm, when the agent calls `getBino` while processing a vertex $\widetilde{u} \in V(\widetilde{G})$, it is located on $f(\widetilde{u})$ and it gets a map of $B_1(f(\widetilde{u}), G)$.

During one step, the agent consider such an unexplored cluster $\widetilde{C}$ and builds a shortest path $\pi_{\widetilde{C}}$ in $\widetilde{C}$ that visit all vertices of $\widetilde{C}$. Then, it follows the port numbers appearing on $\pi_{\widetilde{C}}$ in G and for each node $\widetilde{u} \in \pi_{\widetilde{C}}$, it calls `getBino` to get a local map B of $B_1(f(\widetilde{u}), G)$. Using this information, it detects what are the missing neighbors of $\widetilde{u}$ in $\widetilde{G}$ (with the associated port-numbers) and stores all missing neighbors in a set Z. Using the local maps, he also detects if the missing neighbors of two adjacent nodes of $\widetilde{C}$ coincide (using the set $\equiv_0$), and remembers what edges should be added between two missing neighbors of a given node (in a set H). Then, computing the reflexive and transitive closure $\equiv$ of $\equiv_0$, the agent computes the set of vertices and edges that should be added to $\widetilde{G}$ in such a way that f induces an isomorphism between $B_1(\widetilde{u}, \widetilde{G})$ and $B_1(f(\widetilde{u}), G)$ for every $\widetilde{u}$ in $\widetilde{C}$. The clusters containing the new vertices of $\widetilde{G}$ are then computed and added to the list of clusters to be explored.

While exploring a cluster $\widetilde{C}$, the agent may discover an error in the construction of the graph $\widetilde{G}$. The following errors can occur:

- The reconstructed graph $\widetilde{G}$ is not a simple graph, or the labeling $\widetilde{\delta}$ is not a port-numbering;
- There is a vertex $\widetilde{u}_i$ in $\pi_{\widetilde{C}}$ such that the ball B obtained via `getBino` in the execution of Algorithm 5.3 when considering $\widetilde{u}_i$ is not isomorphic to $B_1(\widetilde{u}_i, \widetilde{G})$.

If such an error is detected during the algorithm, the agent stops executing the algorithm and starts executing Algorithm 5.1.

7.2. Correction of the Algorithm. We first show that Algorithm 5.3 is an exploration algorithm. Observe that if an error is detected during the execution of the algorithm on a graph G, then Algorithm 5.1 is executed and if the algorithm stops, then G is explored. Suppose now that no error is detected during the execution.

Let $\widetilde{G}$ be the graph constructed during the execution of the algorithm on G. Let $\widetilde{C}_0 = \{\widetilde{v}_0\}, \widetilde{C}_1, \widetilde{C}_2, \ldots, \widetilde{C}_i, \ldots$ be the list of clusters of $\widetilde{G}$ constructed during the execution of the algorithm. Observe that this list of clusters is obtained by a depth-first-search traversal of the cluster-tree of $\widetilde{G}$. We denote by $\widetilde{G}_i$ the graph constructed after the first i clusters $\widetilde{C}_0 = \{\widetilde{v}_0\}, \widetilde{C}_1, \widetilde{C}_2, \ldots, \widetilde{C}_i$ have been explored. Let $\partial\widetilde{G}_i = \widetilde{G}_i \setminus \cup_{i=1}^{i} \widetilde{C}_i$, i.e., the vertices of $\widetilde{G}_i$ that have not been explored yet, i.e., the vertices of $\widetilde{G}_i$ that belong to some cluster of ToVisit once $\widetilde{C}_i$ has been explored.

We first observe that the graph $\widetilde{G}_i$ is a Weetman graph with respect to $\widetilde{v}_0$. This property holds by construction of $\widetilde{G}_i$: the Triangle condition follows from the way the edges in E_{hor} are defined, and the Interval-Neighborhood condition follows from the way the equivalence relation $\equiv$ is defined.

LEMMA 5.17. *The graph $\widetilde{G}_i$ satisfies TC($\widetilde{v}_0$) and INC($\widetilde{v}_0$)*

We now define a map $f_i : V(\widetilde{G}_i) \to V(G)$. For any vertex $\widetilde{u}$ in $\widetilde{G}$, let $\pi_{\widetilde{u}}$ be a shortest path from $\widetilde{v}_0$ to $\widetilde{u}$ in $\widetilde{G}$, let $f_i(\widetilde{u}) = \text{reach}_G(v_0, \widetilde{\delta}(\pi_{\widetilde{u}})$ (if it exists). One can show that this map is well defined and that $f(\widetilde{u})$ is independent of the choice of $\pi_{\widetilde{u}}$:

LEMMA 5.18. *For every $u \in V(\widetilde{G})$, for every path $\pi'_{\widetilde{u}}$ from $\widetilde{v}_0$ to $\widetilde{u}$ in $\widetilde{G}$, $f(\widetilde{u}) = \text{reach}_G(v_0, \widetilde{\delta}(\pi'_{\widetilde{u}}))$.*

REMARK 5.19. By Lemma 5.18, if the agent executes `getBino` while considering a vertex $\widetilde{u}$ of $\widetilde{G}$, then the agent is located on $u = f(\widetilde{u})$ and gets `getBino` returns the graph $B_1(u, G)$ pointed at u.

We prove by induction on i that f_i is a local isomorphism for vertices in $\widetilde{G}_i \setminus \partial\widetilde{G}_i$ and is locally injective for vertices in $\partial\widetilde{G}_i$.

PROPOSITION 5.20. *At any step i, the following holds:*

(1) for any $\widetilde{u} \in \widetilde{G}_i \setminus \partial\widetilde{G}_i$, $B_1(\widetilde{u}, \widetilde{G}_i)$ is isomorphic to $B_1(f_i(\widetilde{u}), G)$,

Algorithm 5.3: Exploration Algorithm for Weetman Graphs

ToVisit $\leftarrow$ emptyStack(); $\widetilde{G} \leftarrow (\{\widetilde{v}_0\}, \emptyset)$; push$((\{\widetilde{v}_0\}, \emptyset)$, ToVisit); $\mathsf{pos}_{\widetilde{G}} \leftarrow \widetilde{v}_0$;
repeat
- $\widetilde{C} \leftarrow$ pop(ToVisit);
- Compute in $\widetilde{G}$ a shortest path $\pi'_{\widetilde{C}}$ from $\mathsf{pos}_{\widetilde{G}}$ to a vertex $\widetilde{u}_0 \in \widetilde{C}$;
- Compute a shortest path $\pi_{\widetilde{C}} = (\widetilde{u}_0, \widetilde{u}_1, \ldots, \widetilde{u}_k)$ in $\widetilde{G}$ visiting all vertices of $\widetilde{C}$;
- Follow the port-numbers of the path $\pi'_{\widetilde{C}}$ in G;
- $\mathsf{pos}_{\widetilde{G}} \leftarrow \widetilde{u}_0$; $Z \leftarrow \emptyset$; $\equiv_0 \leftarrow \emptyset$; $H \leftarrow \emptyset$;
- **for each** $\widetilde{u}_i \in \pi_{\widetilde{C}}$ **do**
  - $B \leftarrow$ getBino();
  - **for each** $w \in N_B(w_0)$ **do**
    - $p \leftarrow \delta_{w_0}(w)$; $q \leftarrow \delta_w(w_0)$;
    - **if** $\nexists \widetilde{u} \in N_{\widetilde{G}}(\widetilde{u}_i)$ such that $\widetilde{\delta}_{\widetilde{u}_i}(\widetilde{u}) = p$ **then**
      - $Z \leftarrow Z \cup \{(\widetilde{u}_i, p, q)\}$;
      - **for each** $w' \in N_B(w_0) \cap N_B(w)$ **do**
        - $p' \leftarrow \delta_{w_0}(w')$; $q \leftarrow \delta_{w'}(w_0)$; $r \leftarrow \delta_w(w')$; $r' \leftarrow \delta_{w'}(w)$;
        - **if** *there exists* $\widetilde{u}' \in N_G(\widetilde{u}_i)$ *with* $\delta_{\widetilde{u}_i}(\widetilde{u}') = p'$ **then**
          - $\equiv_0 \leftarrow \equiv_0 \cup \{((\widetilde{u}_i, p, q), (\widetilde{u}', p', q'))\}$;
        - **else**
          - $H \leftarrow H \cup \{((\widetilde{u}_i, p, q), (\widetilde{u}_i, p', q'), (r, r'))\}$;
  - **if** $i < k$ **then**
    - move through port $\widetilde{\delta}_{\widetilde{u}_i}(\widetilde{u}_{i+1})$ in G;
    - $\mathsf{pos}_{\widetilde{G}} \leftarrow \widetilde{u}_{i+1}$;
- Compute the reflexive and transitive closure $\equiv$ of $\equiv_0$ (the equivalence class of $(\widetilde{u}, p, q) \in Z$ is denoted by $[\widetilde{u}, p, q]$);
- $V_{\text{new}} \leftarrow Z/\equiv$; $E_{\text{vert}} \leftarrow \emptyset$; $E_{\text{hor}} \leftarrow \emptyset$;
- **for each** $(\widetilde{u}, p, q) \in Z$ **do**
  - Add the edge $\widetilde{u}[\widetilde{u}, p, q]$ to E_{vert} with $\widetilde{\delta}_{\widetilde{u}}([\widetilde{u}, p, q]) = p$ and $\widetilde{\delta}_{[\widetilde{u},p,q]}(\widetilde{u}) = q$;
- **for each** $((\widetilde{u}, p, q), (\widetilde{u}, p', q'), (r, r')) \in H$ **do**
  - Add the edge $[\widetilde{u}, p, q][\widetilde{u}, p', q']$ to E_{hor} with $\widetilde{\delta}_{[\widetilde{u},p,q]}([\widetilde{u}, p', q']) = r$ and $\widetilde{\delta}_{[\widetilde{u},p',q']}([\widetilde{u}, p, q]) = r'$;
- $V(\widetilde{G}) \leftarrow V(\widetilde{G}) \cup V_{\text{new}}$; $E(\widetilde{G}) \leftarrow E(\widetilde{G}) \cup E_{\text{vert}} \cup E_{\text{hor}}$;
- Compute the connected components $\widetilde{C}'_1, \widetilde{C}'_2, \ldots, \widetilde{C}'_\ell$ of $(V_{\text{new}}, E_{\text{hor}})$;
- **for each** $\widetilde{C}'_j$ **do** index$(\widetilde{C}'_j) \leftarrow \min\{i \mid \exists \widetilde{u}' \in \widetilde{C}'_j \cap N_{\widetilde{G}}(\widetilde{u}_i)\}$;
- **for each** $\widetilde{C}'_j$ *(sorted by increasing index)* **do** push$(\widetilde{C}'_j$, ToVisit) ;

until ToVisit is empty **or** an error is detected in $\widetilde{G}$;
if an error is detected in $\widetilde{G}$ **then**
- Execute Algorithm 5.1

(2) for any $\widetilde{u} \in \partial\widetilde{G}_i$, f_i is an injective map from $N_{\widetilde{G}}(\widetilde{u})$ to $N_G(f_i(\widetilde{u}))$.

Consequently, if the algorithm terminates without detecting an error, the clique complex $X(\widetilde{G})$ of the reconstructed graph $\widetilde{G}$ is a cover of the clique complex $X(G)$ of G. Since $\widetilde{G}$ is a Weetman graph, $X(\widetilde{G})$ is simply connected and is the universal cover of

$X(G)$. Consequently, since G is connected, the cover $f : \widetilde{G} \to G$ is surjective and every vertex u of G has been explored.

When Algorithm 5.3 is executed on a Weetman graph G, then the reconstructed graph $\widetilde{G}$ is isomorphic to G.

PROPOSITION 5.21. *If G satisfies TC(v_0) and INC(v_0), then for each step i, $\widetilde{G}_i$ is isomorphic to $f_i(\widetilde{G}_i)$.*

Consequently, no error is detected during the execution of Algorithm 5.3 on a Weetman graph G and the execution stops after all vertices of $\widetilde{G}$ have been explored.

7.3. Complexity of the Algorithm. We are interested in evaluating the number of moves performed by an agent that executes Algorithm 5.3 on a Weetman graph G. The number of moves the agent executes on G is related to the number of moves the agent "executes" on $\widetilde{G}$: each move on G corresponds to a move on $\widetilde{G}$. We denote by $\mathcal{C}$ the set of clusters of $\widetilde{G}$.

When it explores a cluster $\widetilde{C} \in \mathcal{C}$, the agent compute a shortest path $\pi_{\widetilde{C}}$ visiting all vertices of $\widetilde{C}$. This path has length at most $2(|V(\widetilde{C})| - 1)$.

When going from one cluster to another, the agent computes a shortest path $\pi'_{\widetilde{C}}$ to reach $\widetilde{C}$. In order to evaluate the sum of the length of these paths, we observe that the clusters are explored in an order that is a depth-first-search order on the cluster-tree of $\widetilde{G}$. Observe also that when considering all the children $\widetilde{C}'_1, \ldots, \widetilde{C}'_\ell$ of a cluster in the cluster-tree, these clusters are sorted according to the order in which their first neighbor appears in the path $\pi_{\widetilde{C}}$. Consequently, $\sum_{\widetilde{C} \in \mathcal{C}} |\pi'_{\widetilde{C}}| \leq 2|\mathcal{C}| + 3\sum_{\widetilde{C} \in \mathcal{C}}(|V(\widetilde{C})| - 1) \leq 3\sum_{\widetilde{C} \in \mathcal{C}} |V(\widetilde{C})|$.

Since $\sum_{\widetilde{C} \in \mathcal{C}} |V(\widetilde{C})| = |V(\widetilde{G})| = |V(G)|$, the number of moves performed by the agent is linear in the number of vertices of G.

THEOREM 5.22. *Algorithm 5.3 is an Exploration algorithm that explores any Weetman graph G in $O(|V(G)|)$ moves.*

REMARK 5.23. We can wonder what happens when Algorithm 5.3 is executed on a graph G that is not a Weetman graph. First, observe that if G satisfies TC(v_0) and INC(v_0) for the starting position v_0 of the agent, then Algorithm 5.3 is still an Exploration algorithm for G that uses a linear number of moves.

If $X(G)$ is not simply connected but if the universal cover $\widetilde{X}(G)$ is the clique complex $X(\widetilde{G})$ of a Weetman graph $\widetilde{G}$ (or satisfies TC($\widetilde{v}_0$) and INC($\widetilde{v}_0$) where $\widetilde{v}_0$ is the preimage of the starting position v_0 of the agent in G), then Algorithm 5.3 reconstructs $\widetilde{G}$ and explores G in $O(|V(\widetilde{G})|)$ moves. Observe that by Proposition 5.10, $\Omega(|V(\widetilde{G})|)$ moves are necessary for an Exploration algorithm before it halts in G.

8. Conclusion

Enhancing a mobile agent with binoculars, we have shown that, even without any global information it is possible to explore and halt in the class of graphs whose clique complexes have a finite universal cover. This class is maximal and is the counterpart of tree networks in the classical case without binoculars. Note that, contrary to the classical case, where the detection of unvisited nodes is somehow trivial (any node that is visited while not backtracking is new, and the end of the discovery of new nodes is immediate at leaves), it is more challenging in the new model to detect when it is no more possible to encounter "new" nodes.

The class where we are able to explore is fairly large and has been proved maximal when using binoculars of range 1. When considering binoculars of range k, clique complexes are no longer the right tool to use, but we believe we can obtain a similar characterization of explorable graphs by considering other cell complexes associated with the graph. Note that for triangle-free networks, enhancing the agent with binoculars of range 1 does not change the class of explorable networks. More generally, it can also be shown that providing only local information (e.g. using binoculars of range k) cannot be enough to explore all graphs (e.g. graphs with large girth).

Note that our universal Exploration algorithm can actually compute the universal cover of the graph, and therefore yields a Map Construction algorithm if we know that the underlying graph has a simply connected clique complex. However, note that there is no algorithm that can construct the map for all graphs of $\mathcal{FC}$. Indeed, there exist graphs in $\mathcal{FC}$ that are not simply connected (e.g. triangulations of the projective plane) and by the Lifting Lemma, they are indistinguishable from their universal cover. Note that without binoculars, the class of trees is not only the class of graphs that are explorable without information, but also the class of graphs where we can reconstruct the map without information. Here, adding binoculars, not only enables to explore more networks but also give a model with a richer computability structure : some problems (like Exploration and Map Construction) are no longer equivalent.

While providing binoculars is a natural enhancement, it appears here that explorability increases at the cost of a huge increase in complexity: the number of moves, as a function of the size of the graph, increase faster that any computable function. This cannot be expected to be reduced for all explorable graphs for fundamental Turing computability reasons. However, our results on Weetman graphs show that with binoculars, there is a linear exploration algorithm for a class that is much larger than the class of trees. So the fact that the full class of explorable networks is not explorable efficiently should not hide the fact that the improvement is real for large classes of graphs.

One of the interesting open problem is to identify classes of networks that can be explored efficiently (with linear or polynomial algorithms) with binoculars. Natural classes to consider are classes of graphs arising from metric and geometric graph theory. For example, if we consider binoculars of radius 2, it should be quite easy to adapt our algorithm for Weetman graphs to show that all weakly modular graphs (as well as basis graphs of matroids) can be explored in linear time. A next step would be to consider δ-hyperbolic graphs. For these graphs, we know from Theorem 3.9 that the cell complex of a hyperbolic graph G where all cycles of length at most 16δ are 2-dimensional cells is simply connected. Consequently, with binoculars of large enough radius, δ-hyperbolic graphs should be explorable. Moreover, we know that these cell complexes satisfy a linear isoperimetric inequality and it would be interesting to know if we can use this property in order to derive an efficient Exploration algorithm for δ-hyperbolic graphs.

CHAPTER 6

Minimum Base Construction in Anonymous Networks

We saw in the previous chapter that in the standard mobile agent model, an agent can explore a graph without information if and only if the graph is a tree. We now assume that the agent is given initially some information about the network. Observe that if the agent knows a bound $\hat{n}$ on the size n of the network G, then the agent can use this information to explore G: it can explore all paths of length at most $\hat{n}$ starting at its initial position. Observe that if we are given enough information to explore the graph with termination, i.e., if we are given an algorithm that explores a graph G and stops, then one can easily compute an upper bound on the number of nodes of the graph just by counting the number of steps performed during the execution. In other words, from a computability point of view, exploration is as hard as computing an upper bound on the size of the network.

In this chaper, we assume that the agent initially knows an upper bound $\hat{n}$ on the size of the network. Observe that the exploration algorithm sketched above is very inefficient since the number of moves performed by such an algorithm is the size of the view $\mathcal{T}_G(v_0, \hat{n})$ that can be exponential in $\hat{n}$. Several methods have been proposed in the litterature (see Section 3) to solve the exploration problem much more efficiently when an upper bound on the size of the network is initially given to the agent. The number of moves performed by these algorithms is polynomial in the given bound.

As observed by Yamashita and Kameda [**259**], the view of a node is the maximum amount of information the node can gather about the network it belongs to. Similarly, a mobile agent evolving in an unknown graph cannot learn more information during the execution of any algorithm than the information contained in the view of its homebase. Even if views are a priori infinite, Norris showed in [**186**] that it is enough to compute the view of a node up to depth $2n$ to obtain all the computable information about a network of size n. Hendrickx later showed that it is enough to compute the view of depth $O(D \log(n/D)$ where D is the diameter of the network (See Section 1). In general, the view of depth k of a node has a size that is exponential in k, but Tani showed that in anonymous message passing systems, the nodes can compute efficiently compressed representations of their views that have a size polynomial in the number of nodes of the network.

Even if views contain all the information that can be gathered, they are not always easy to handle since they are infinite and since their compressed representations can still be quite large. As observed by Boldi and Vigna [**34**] as well as Yamashita and Kameda [**259**], when dealing with networks endowed with a port-numbering, one can use the notion of minimum bases (that is called quotient graph by Yamashita and Kameda) that also contains all the information one can gather about the network. The minimum base $B(G)$ of G is a (di)graph such that for any (di)graph H, if (the directed version of) G is a cover of H, then H is a cover of $B(G)$ (See Section 2). In some sense, computing the view of a node of G is looking for the largest cover of G (its universal cover), while constructing the minimum base is looking for the smallest (di)graph that is covered by $B(G)$. In anonymous message passing systems, one can compute the minimum base of the network with polynomial algorithms (See Section 2).

In this chapter, we explain how a mobile agent evolving in a network can also compute the minimum base of the network in polynomial time. More precisely, we show that if we are given an exploration algorithm for a given network (or a family of networks), we can use it as a black-box to obtain an algorithm that computes the minimum base of the network. The overhead in the complexity is polynomial in the size of the minimum base. The techniques we use are similar to the techniques used to minimize finite deterministic automata (See Section 4).

The results of this chapter are based on the paper [**68**].

1. View Construction in Anonymous Message-passing Systems

As observed by Yamashita and Kameda [**259**], in a network of processes, the view $\mathcal{T}_G(v)$ of a vertex v contains all the information v can learn about the network. Consequently, to solve any distributed task, each vertex v can first compute its view $\mathcal{T}_G(v)$ and then perform some local computation based on $\mathcal{T}_G(v)$. Note however that if the graph G is not a tree, then the view of a vertex is an infinite tree. However, Norris [**186**] showed that for any graph G with n vertices, if two vertices $u, v \in V(G)$ have the same view of depth $n-1$, i.e., $\mathcal{T}_G(u, n-1) = \mathcal{T}_G(v, n-1)$, then they have the same view, i.e., $\mathcal{T}_G(u) = \mathcal{T}_G(v)$. Using a pseudo-synchronous algorithm, each node v can easily compute its view $\mathcal{T}_G(v, k)$ of depth k for any given k. Consequently, if the processes know the number n of vertices in the network as well as the diameter D of the network, each vertex v can compute its view $\mathcal{T}_G(v, n + D)$ of depth $n + D$. In this way, each node knows all the views of depth n that appear in the network and can compute its view up to arbitrary depth. It can then use this to locally solve the distributed task. If instead of knowing the size n and the diameter D, the vertices only know a bound $\hat{n}$ on the size of the network, the nodes can compute their view up to depth $2\hat{n}$ and then solve the problem locally. This shows that any distributed task can be solved (if it is solvable) in $O(n)$ (resp. $O(\hat{n})$) rounds of communication in an anonymous network provided the size n (resp. a bound $\hat{n}$ on the size) of the network is initially known by the processes. Note however that again the views that the vertices exchange can be of exponential size. In order to reduce the size of the messages exchanged while the vertices construct their views, Tani [**238**] observed that for any k, there are at most n different views of depth k in a network G of size n. Using this, he was able to design an algorithm allowing each node to compute its view of depth k in G in $2k$ rounds while exchanging messages of size polynomial in n for any given k.

Hendrickx [**139**] showed that since we are considering networks that are endowed with a port-numbering, if two vertices u, v have the same view up to depth k (i.e., $\mathcal{T}_G(u, k) = \mathcal{T}_G(v, k)$) for some $k = O(D + D\log(n/D))$, then they have the same view. Dereniowski, Kosowski, and Pajak [**102**] showed that one cannot find a better upper bound for k up to a constant factor. This shows in particular that we cannot remove the dependency on the size of G. One of the consequence of the result of Hendrickx is that if the processes initially know only a bound $\hat{D}$ on the diameter of the network, then by constructing their view up to depth $\hat{D}$, the vertices can discover the maximum degree Δ of G. Using $\hat{n} = \Delta^{\hat{D}}$ as an upper bound on the size of G, this means that it is enough for each vertex v to compute its view $\mathcal{T}_G(v, k)$ for $k = O(\hat{D}^2 \log(\Delta))$ in order to be able to reconstruct its view locally up to any arbitrary depth.

2. Minimum Bases

Views enable to express the symmetries between nodes but they are not always easy to handle since we generally have to handle infinite or very large trees. The notion of minimum bases [**34**] (or equivalently quotient graphs [**259**]) enables to express the symmetries of the network using only finite graphs. Moreover, as it is the case for the views, the minimum base of a network contains all the information a node can learn about the network. The view of a node in a network G is the universal cover of G and is thus the “largest” cover of G. On the opposite, the minimum base B of G is the smallest graph that is covered by G: for any network H such that G covers H, H covers B. Note however that contrary to universal covers, for an unlabeled graph G, the minimum base of G does not always exists (but it always exists if we consider (op)fibrations instead of covering maps, see Section 5), but as shown by Boldi and Vigna [**34**], if the graph G is endowed with a port-numbering, then the minimum base of G always exists. Note also that in order to be able to talk about minimum bases, one has to consider directed graphs with multiple arcs and self-loops.

Given a directed graph $D = (V(D), A(D))$ (with possibly self-loops and multiple arcs), for each arc $a \in A(D)$, we denote by $s_D(a)$ its source and by $t_D(a)$ its target (when D will be clear from the context, the subscripts will be omitted). A directed graph D is symmetric if there exists an involution $\sigma_D : A(D) \to A(D)$ such that for each arc $a \in A(D)$, $s_D(\sigma(a)) = t_D(a)$ and $t_D(\sigma(a)) = s_D(a)$. Observe that if a is a self-loop (i.e., $s(a) = t(a)$), it is possible to have $a = \sigma(a)$; in this case, a is called a *symmetric* self-loop, and it is called an *asymmetric* self-loop otherwise.

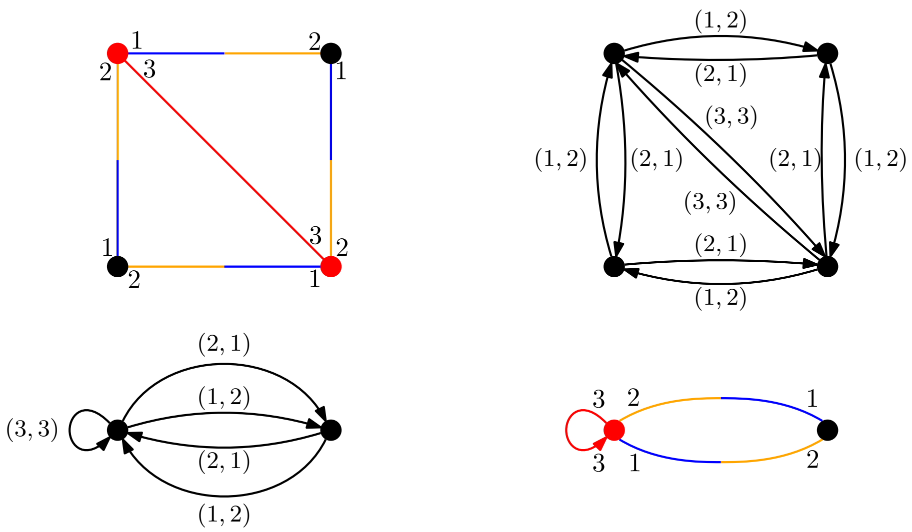


FIGURE 6.1. A graph G with a port-numbering δ, the digraph $Dir(G)$, the minimum base $B(G)$, and a simpler graphical representation of $B(G)$ where each undirected edge corresponds to two symmetric directed arcs.

Given a network G with a port-numbering δ and a vertex-labeling λ, the symmetric directed graph $Dir(G)$ is defined as follows. The vertices of $Dir(G)$ are the vertices of G (with the same labels). For each edge $uv \in E(G)$ there are two arcs a_{uv} and a_{vu} with $s(a_{uv}) = t(a_{vu}) = u$, $t(a_{uv}) = s(a_{vu}) = u$, $\sigma(a_{uv}) = a_{vu}$ and each arc a_{uv} is labeled by $(\delta_u(v), \delta_v(u))$ (See Figure 6.1 (top) for an example).

A homomorphism φ from a directed graph D to a directed graph D' is given by a map from $V(D)$ to $V(D')$ and a map from $A(D)$ to $A(D')$ (that are both denoted by φ) such that for each $a \in A(D)$, $s_{D'}(\varphi(a)) = \varphi(s_D(a))$ and $t_{D'}(\varphi(a)) = \varphi(t_D(a))$. A directed graph D is a *cover* of a directed graph D' via a homomorphism $\varphi : D \to D'$ if for any vertex $v \in V(D)$, φ induces a bijection between $I_D^-(v) = \{a \in A(D) \mid s_D(a) = v\}$ and $I_D^-(\varphi(v)) = \{a' \in A(D') \mid s_{D'}(a') = \varphi(v)\}$ as well as a bijection between $I_D^+(v) = \{a \in A(D) \mid t_D(a) = v\}$ and $I_D^+(\varphi(v)) = \{a' \in A(D') \mid t_{D'}(a') = \varphi(v)\}$. A symmetric directed graph D is a *symmetric cover* of a symmetric directed graph D' via φ if D is a cover of D' such that for any arc $a \in A(D)$, $\sigma_{D'}(\varphi(a)) = \varphi(\sigma_D(a))$. Observe that if a simple undirected graph G is a cover of a simple undirected graph H, then $Dir(G)$ is a symmetric cover of $Dir(H)$. As always, when dealing with labeled digraphs, we only consider homomorphisms and (symmetric) covers that preserve the labels.

When dealing with (symmetric) covers of directed graphs, we have a result similar to Proposition 5.6.

PROPOSITION 6.1 ([**34**]). *If a digraph D is a cover of a (connected) digraph D' via a covering map φ, there exists q such that for all $x \in V(D') \cup A(D')$, $|\varphi^{-1}(x)| = q$. This number q is called the number of* sheets *of the cover.*

Given a network G, the *minimum base* $B(G)$ of G is a labeled digraph such that for any digraph D such that $Dir(G)$ is a covering of D, then D is a covering of $B(G)$ [**34**] (See Figure 6.1 for an example). Observe that $B(G)$ and $Dir(G)$ have the same universal cover $Dir(\widetilde{G})$. In fact, one can reconstruct the minimum base $B(G)$ of G from the collection of the views of the vertices of G:

- $V(B(G)) = \{\mathcal{T}_G(v) \mid v \in V(G)\}$, i.e., there is a vertex in $B(G)$ for each distinct view of G.
- there is an arc $a_{T_1,T_2,p,q}$ from $T_1 = \mathcal{T}_G(v_1)$ to T_2 if there exists a neighbor v_2 of v_1 in G such that $\mathcal{T}_G(v_2) = T_2$, $\delta_{v_1}(v_2) = p$, and $\delta_{v_2}(v_1) = q$. The label of $a_{T_1,T_2,p,q}$ is (p, q) and $\sigma_{B(G)}(a_{T_1,T_2,p,q}) = a_{T_2,T_1,q,p}$.

The symmetric covering map φ from $Dir(G)$ to $B(G)$ maps a vertex v to $\mathcal{T}_G(v)$ and thus, two nodes of G have the same view if and only if they have the same image in $B(G)$ by φ.

Observe that using the results mentionned above, if we know a bound on the size of the graph, then we can compute the minimum base of a network G in a number of rounds that is linear in the given bound by exchanging messages of size polynomial in the size of G. Other polynomial algorithms have been proposed to compute the minimum base of a network. In [**73**, **77**], reusing ideas from Mazurkiewicz [**170**], the idea is to compute the minimum base using a procedure that is similar to degree refinment [**34**, **160**] (or to the method used to minimize a deterministic finite automata [**142**, **172**]). One can also obtain another algorithm by adapting the algorithm of [**99**] to message passing systems; the construction proposed in [**99**] is inspired from the characterization of coverings of a given graph by Reidemeister [**209**]

When an agent evolving in a network is given a bound $\hat{n}$ on the size of the network, it is also possible to compute the minimum base by computing the view of depth $2\hat{n}$ by traversing all paths of length $2\hat{n}$ starting from the initial position of the agent. Note however that this leads to a very inefficient algorithm since the number of such paths can be exponential in $\hat{n}$.

3. Universal Exploration Sequences

In order to explore more efficiently an unknown graph, one can use *universal exploration sequence* (UXS) [**152**] if we initially know a bound on the number of vertices of the network.

Given a a node u and an integer $p \in [1, \deg(u)]$, we denote by $\mathrm{succ}(u, p)$ the unique node v such that $\delta_u(v) = p$. Given a sequence of integers $(a_1, a_2, \ldots, a_k)$, an *application* of this sequence to a graph G at node u is the sequence of nodes $(u_0, \ldots, u_{k+1})$ obtained as follows: $u_0 = u$, $u_1 = \mathrm{succ}(u_0, 0)$; for any $1 \leq i \leq k$, $u_{i+1} = \mathrm{succ}(u_i, (p + a_i) \bmod d(u_i))$, where $p = \delta_{u_i}(u_{i-1})$. A sequence $(a_1, a_2, \ldots, a_k)$ whose application to a graph G at any node u contains all nodes of this graph is called a UXS for this graph. A UXS for a class $\mathcal{G}$ of graphs is a UXS for all graphs in this class.

For all integers $\hat{n}$ and Δ, let $U(\hat{n}, \Delta)$ be a UXS for the class $\mathcal{G}_{\hat{n},\Delta}$ of all graphs with at most $\hat{n}$ nodes and maximum degree at most Δ. The following important result, based on a reduction from Koucký [**152**], is due to Reingold [**210**].

THEOREM 6.2 ([**210**]). *For any positive integer $\hat{n}$, there exists a UXS $Y(\hat{n}) = (a_1, a_2, \ldots, a_M)$ for the class $\mathcal{G}_{\hat{n},\hat{n}}$ of all graphs with at most $\hat{n}$ nodes, such that*

- *M is polynomial in $\hat{n}$,*
- *for any $i \leq M$, the integer a_i can be constructed using $O(\log \hat{n})$ bits of memory.*

The above result implies that a (usually non-simple) path $(u_0, \ldots, u_{M+1})$ traversing all nodes can be computed (node by node) in memory $O(\log n)$, for any graph with at most n nodes. Moreover, logarithmic memory suffices to walk back and forth on this path: to walk forward at node u_i, port $(p + a_i) \bmod d(u_i)$ should be computed when coming by port p, to walk backward, port $(p - a_i) \bmod d(u_i)$ should be computed.

The degree of the polynomial in Theorem 6.2 bounding the length of the UXS is very large. Aleliunas et al. [**6**] proved the existence of universal exploration sequences of length of small polynomial size.

THEOREM 6.3 ([**6**]). *For any positive integers $\hat{n}, \Delta$, $\Delta < \hat{n}$, there exists a universal exploration sequence of length $O(\hat{n}^3 \Delta^2 \log \hat{n})$ for the family of all graphs with at most $\hat{n}$ nodes and maximum degree at most Δ.*

Note that the exploration sequences in the proposition above are not constructible in logarithmic memory, while the log-space constructible sequences from Proposition 6.2 are much longer (though still polynomial in n). The resutl of Aleliunas et al. [**6**] is obtained by derandomizing a (classical) random walk where at each step the next outgoing port is chosen uniformly at random. Derandomizing the Metropolis walk (that can be seen as a biased random walk), Kosowski [**150, 151**] shows that one can gain a factor Δ in the complexity (at the cost of a $\log \hat{n}$ factor).

THEOREM 6.4 ([**150**]). *For any positive integers $\hat{n}, \Delta$, $\Delta < \hat{n}$, there exists an algorithm for a mobile agent that explores any anonymous graph with at most $\hat{n}$ nodes and maximum degree at most Δ in $O(\hat{n}^3 \Delta \log^2 \hat{n})$ steps.*

These results show that even very little information about the network (a bound on the size of the network), an agent can explore a network quite efficiently (in a time that is polynomial in the given bound).

4. Minimum Base Construction by a Mobile Agent

As explained above, if we are given an upper bound $\hat{n}$ on the size of a network G, the agent can compute the view $\mathcal{T}_G(v_0, \hat{n})$ of the homebase v_0 of the agent in G and from this view, it can compute the minimum base of G.

We now present a more efficient algorithm to compute the minimum base of a network G assuming we are given an exploration algorithm $\mathcal{A}$ for G. The overhead of this algorithm is polynomial in the size of the minimum base $B(G)$ of G. Our algorithm uses ideas that are usually used to minimize a deterministic automaton.

Given a graph G, a node u of G and a sequence of edge-labels $Y = ((p_1, q_1), (p_2, q_2), \ldots, (p_j, q_j))$, we say that Y is *accepted* from u if there exists a path $P = (u = u_0, u_1, \ldots, u_j)$ in G such that $\delta(P) = Y$, i.e., for each i, $1 \leq i \leq j$, $(p_i, q_i) = \delta(u_{i-1}, u_i)$. Recall that for any $k > 0$, two vertices u, v that have the same view up to depth k are said to be k-equivalent; we denote it by $u \equiv_k v$. The k-class of u is the set of all vertices that are k-equivalent to u. Given any two distinct k-classes C, C', a (C, C')-*distinguishing path* is a sequence of edge-labels $Y_{C,C'} = ((p_1, q_1), (p_2, q_2), \ldots, (p_j, q_j))$ of length at most k such that $Y_{C,C'}$ is accepted from each node $u \in C$ and it is not accepted from any node $v \in C'$. For any two distinct k-classes, there always exists either a (C, C')-distinguishing path or a (C', C)-distinguishing path.

We present an algorithm (See Algorithm 6.1) that iterates over k, and for each k, explores the graph and identifies the k-classes of the visited nodes and their neighborhoods. We use the exploration algorithm $\mathcal{A}$ to visit all vertices of G.

For $k = 1$, it is easy to determine the k-class of any node v by traversing each edge incident to v and noting the labels. From this information, one can find the distinguishing paths for any pair of 1-classes. For $k \geq 2$, it is possible to identify the k-classes and the corresponding distinguishing paths (from the knowledge of the $k-1$ classes) using the properties below.

PROPOSITION 6.5. *For $k \geq 2$, two nodes u and v belong to the same $(k+1)$-class, if and only if*

(i) u and v belong to the same k-class, and
(ii) for each i, $0 \leq i \leq deg_G(u) = deg_G(v)$, $u_i = \mathrm{succ}(u, i)$ and $v_in = \mathrm{succ}(v, i)$ belong to the same k-class and $\delta(u, u_i) = \delta(v, v_i) = (i, j)$, for some $j \geq 0$.

For a vertex $v \in V(G)$, we denote its k-class by $C_k(v)$. Note that if $C_k(u) = C_k(v)$ for two vertices $u, v \in V(G)$, then $C_{k-1}(u) = C_{k-1}(v)$. Consequently, for each k-class C, there is a unique $(k-1)$-class C' containing the vertices of C. We can arrange the classes in a tree: there is only one class at depth 0, and for each k, the children of a k-class C are the $(k+1)$-classes $C_1, \ldots, C_p$ such that $C = \bigcup_{i=1}^{p} C_i$. The number of children of a class C is called the degree of C and is denoted by $\deg(C)$.

Algorithm 6.1: Class-Refinement(n)

```
Let v1, v2, ... vt be the sequence of nodes visited by A, possibly containing
  duplicate nodes ;
Follow A and for each node vi do
    Store the labels of each edge incident to vi;
Compute the number of 1-classes and store a distinguishing path for each pair
  of distinct classes ;
Return to v1;
k := 0;
repeat
    Increment k;
    Execute A and for each visited node vi do
        for each edge (vi, w) incident to vi do
            Compute the k-class of w (using the distinguishing paths between
              the children of the (k − 1) class of w);
            Store the label of (vi, w) and the index of the k-class of w ;
    Compute the number of (k + 1)-classes and store a distinguishing path for
      each pair of distinct (k + 1)-classes ;
    Return to v1;
until the number of (k + 1)-classes is equal to the number of k-classes;
Compute the minimum base ;
```

THEOREM 6.6. *Algorithm 6.1 builds the minimum base of any graph G in $O(N_G \cdot \Delta n_B D_B \log(n_B/D_B))$ moves where N_G is the number of moves performed by $\mathcal{A}$ on G and where Δ, n_B, and D_B are respectively the maximum degree, the size, and the diameter of the minimum base B of G.*

PROOF. During the $(k+1)$th iteration, on each node v reached by the execution of $\mathcal{A}$, for each neighbor w of v, the agent computes the k-class of w. Since it knows the k-class $C_k(v)$ of v, it also knows the $(k-1)$ class $C_{k-1}(w)$ of w. Let $C_1, C_2, \ldots, C_{\deg(C_{k-1}(w))}$ be the k-classes of $\mathbf{G}$ that are the children of $C_{k-1}(w)$. In order to compute the k-class $C_k(w)$ of w, the agent needs to check at most $\deg(C_{k-1}(w)) - 1$ different distinguishing paths of length at most k.

Assume that the algorithm ends after $p+1$ iterations of the loop, i.e., the set of p-classes is the same as the set of $(p+1)$-classes. For each vertex v, and for each neighbor w of v, overall the agent traverses at most $\Sigma_{k=0}^{p}(\deg(C_k(w) - 1)$ paths of length at most p. Note that there are at least $\Sigma_{k=0}^{p}(\deg(C_k(w) - 1)$ p-classes, i.e., $\Sigma_{k=0}^{p}(\deg(C_k(w)-1) \leq n_B$. Consequently, for each neighbor w of each vertex v traversed by the exploration algorithm $\mathcal{A}$, the agent traverses at most $(p+1) \cdot n_B$ edges.

Since we know that $p \leq D_B \log(n_B/D_B)$ [**139**], overall the agent performs $O(N_G \cdot \Delta n_B D_B \log(n_B/D_B))$ moves. □

If we use the algorithms of Aleliunas et al. [**6**] and Kosowski [**150**] described in Section 3 as an exploration algorithm, we obtain the following corollary.

Corollary 6.7. *Given an upper bound $\hat{n}$ on the size of a network G, one can compute its minimum base in time $O(\hat{n}^3\Delta^3 \log \hat{n} \cdot n_B D_B \log(n_B/D_B)) = O(\hat{n}^5\Delta^3 \log \hat{n})$ or $O(\hat{n}^3\Delta^2 \log^2 \hat{n} \cdot n_B D_B \log(n_B/D_B)) = O(\hat{n}^5\Delta^2 \log^2 \hat{n})$.*

Remark 6.8. The algorithm presented in this chapter is the same as the one in [**68**] but the analysis is different, leading to a better complexity (improved by a factor of $\hat{n}$).

5. Computing the Minimum Base without Incoming Port-numbers

All the algorithms presented in the previous sections use intensively the ability for the agent to backtrack. This is possible since when the agent reaches a node, it knows its incoming port-number.

When the agent is not able to detect the incoming port-number of a node, one has to use opfibrations of graphs instead of covers of graphs. A directed graph D is an *opfibration* of a directed graph D' via a homomorphism $\varphi : D \to D'$ if for any vertex $v \in V(D)$, φ induces a bijection $I_D^+(v) = \{a \in A(D) \mid t_D(a) = v\}$ and $I_D^+(\varphi(v)) = \{a' \in A(D') \mid t_{D'}(a') = \varphi(v)\}$.

Given a graph G with port-numbering δ, the digraph $Dir(G)$ is a directed graph (that is no longer symmetric) whose vertices are the vertices of G and where each edge $uv \in E(G)$ is replaced by an arc a_{uv} labeled $\delta_u(v)$ and an arc a_{vu} labeled $\delta_u(v)$ such that $s(a_{uv}) = t(a_{vu}) = u$ and $t(a_{uv}) = s(a_{vu}) = u$. In this case, the minimum base of G is the smallest digraph $B(G)$ such that G is an opfibration of $B(G)$.

Even if the previous algorithms cannot be used to compute the minimum base of G, we showed in [**64**,**67**] that one can still compute the minimum base of a network G when we are given an upper bound $\hat{n}$ on its size. The idea is to enumerate all digraphs H of size at most $\hat{n}$ that are not opfibred over any other digraph and to consider all possible vertices $v \in V(H)$. For two given pairs (H, v), (H', v'), there is always a distinguishing path that enables to distinguish (H, v) and (H', v'). By enumerating all such paths, it is possible to find the unique pair (H, v) such that H is the minimum base of G and v is the image of the homebase v_0 of the agent in v by the opfibration map.

CHAPTER 7

Mapping Polygons

As explained in the previous chapter, the minimum base contains all the information an agent can gather about the network (provided a bound on its size is given). A natural problem to consider is the mapping problem where the goal is to output a map of the underlying network. Unfortunately, it is easy to build different networks that have the same minimum base, showing that the mapping problem cannot always be solved. For example, if we are in the family of cycles, the agent has no way to be able to compute the size of the cycle. In fact, using Reidemeister theorem [**209**], one can easily construct families of networks that have the same size and the same minimum base, showing that there is no hope to infer the map of the network from its size. However, one can ask if for some classes of graphs, the problem is solvable (using the fact that we know that the network is in this class). For example, one can consider planar graphs or chordal graphs. The idea is to use the structural properties of these classes.

In this chapter, we focus on the mapping problem for visibility graphs of simple polygons. There are potentially many geometric information that we can provide to an agent moving in a polygon. The difficulty of the mapping problem depends on the characteristics of the environment itself and on the sophistication of the agents, i.e., on their sensory and locomotive capabilities. A natural question is how much sophistication an agent needs to be able to solve the problem. The ultimate goal is to characterize the difficulty of the mapping problem by finding minimal agent models that allow an agent to create a map.

We consider agents operating in environments in the shape of simple polygons. For many tasks, instead of inferring a detailed map of the geometry of the environment, it is enough to obtain the visibility graph. The visibility graph has a node for each vertex of the polygon and an edge connecting two nodes if the corresponding vertices see each other, i.e. if the straight-line segment between them is contained in the polygon. The goal in this context becomes to find minimal agent models that allow an agent inside a polygonal environment to reconstruct the visibility graph of the environment. The information the agent can gather must be sufficient to uniquely infer the visibility graph. A variety of minimalistic agent models have been studied, focusing on different types of environments and objectives [**8, 45, 92, 148**]. The model considered here originates from [**234**]. Roughly speaking, our agent is allowed to move along the edges of the visibility graph. While at a vertex, the agent sees the vertices visible from its current location in counter-clockwise (ccw) order starting with its ccw neighbor along the boundary. Apart from this ordering the vertices are indistinguishable to the agent. In each move the agent may select one of them and move to it.

Table 1 summarizes known results that are based on the agent model considered here, as well as open problems. Besides employing different sensors, the results differ in the agent's initial knowledge about the size n of the polygon, its movement capabilities, and, in case of positive results, the running times of the reconstruction algorithms. The first part of the table concerns agents that are restricted to moving along the boundary only. It was shown that even with this severe movement restriction an agent can still reconstruct the visibility graph, as long as it can measure the exact angle between any pair of visible vertices [**105, 106**]. On the other hand, only measuring the angle between the two neighboring vertices along the boundary is not sufficient, even if the agent can

Visibility Graph Reconstruction

sensors	initial info	movement	results solvable	time	source
angles, directions	–	boundary	yes	poly	[**29**]
angles	n	boundary	yes	poly	[**105**]
angles	–	boundary	yes	poly	[**106**]
cvv, boundary angles	n	boundary	no		[**29**]
angle-types		boundary	*open*		
distances		boundary	*open*		
pebble	–	free	yes	poly	[**234**]
cvv, look-back	–	free	no		[**45**]
angle-types, look-back	–	free	yes	poly	[**29**]
angle-types, directions	–	free	yes	poly	[**29**]
directions	$\hat{n}$	free	yes	exp	[**104**]
look-back	$\hat{n}$	free	yes	poly	[**63**, **65**]
angle-types	$\hat{n}$	free	yes	exp	[**64**, **67**]
distances		free	*open*		
no sensors	n	free	*open*		

TABLE 1. Summary of the cases in which an agent is known to be able/not able to solve the visibility graph reconstruction problem. Note that a polynomial running time in a setting where only an upper bound $\hat{n} \geq n$ is known a priori means polynomial in $\hat{n}$ rather than n.

distinguish whether any two visible vertices are neighbors along the boundary ("cvv" in the table) [**29**]

For agents that move across the polygon (as opposed to along the boundary), it is sufficient to be able to mark a single vertex (e.g., with a pebble) in order reconstruct the visibility graph [**234**]. Without this powerful ability, it is difficult for the agent to relate the information it collected so far to subsequent observations. One way to overcome this difficulty is to endowed the agent with a *look-back* sensor that allows to identify the vertex the agent came from in its last move. But even with a look-back sensor some knowledge of an upper bound on the size n of the polygon is required to solve the reconstruction problem [**45**]. A *direction* sensor that measures the angle between the boundary and a global reference direction makes it possible to reconstruct the visibility graph, even in the presence of holes [**104**]

An *angle-type* sensor allows the agent to distinguish convex ($\leq \pi$) from reflex ($\geq \pi$) angles. It was shown before that an agent with a look-back sensor and an angle-type sensor is powerful enough to allow visibility graph reconstruction [**29**]. In this chapter, we show that provided the agent is given an upper bound on the size of the polygon, any of these two sensors alone is enough to be able to reconstruct the visibility graph of the polygon.

THEOREM 7.1 ([**63**, **65**]). *Given an upper bound $\hat{n}$ on the size of a simple polygon $\mathcal{P}$, a look-back agent can reconstruct the visibility graph $G(\mathcal{P})$ of $\mathcal{P}$.*

THEOREM 7.2 ([**64**, **67**]). *Given an upper bound $\hat{n}$ on the size of a simple polygon $\mathcal{P}$, an angle-type agent can reconstruct the visibility graph $G(\mathcal{P})$ of $\mathcal{P}$.*

It remains an open problem whether the angle-type sensor is sufficient even when the agent is restricted to moving along the boundary. Other interesting open problems are whether knowledge of n (or of an upper bound on n) on its own is already enough to reconstruct the visibility graph, and how a distance sensor may be used for reconstruction.

In the agent model we use, the agent moves along the edges of the visibility graph and can locally access some information about the edges. We can model this in the context of general graph exploration of labeled graphs considered in the previous chapters: the information given by the different sensors we consider can be encoded in the labels of the vertices of the visibility graph as well as in the port-numbering associated to this visibility graph.

The results of this chapter are based on the papers [**63**, **65**], [**64**, **67**], and on the survey [**66**].

1. The Visibility Graph Reconstruction Problem – Model and Notations

1.1. Simple Polygons and Visibility Graphs. We consider simple polygons only and we assume polygons to be in general position, i.e. no three vertices lie on a common line. Let in the following $\mathcal{P}$ be such a simple polygon with n vertices. We denote the set of vertices of $\mathcal{P}$ by $V(\mathcal{P})$, where we drop the argument $\mathcal{P}$ whenever the polygon is evident from the context. The boundary of $\mathcal{P}$ together with an (arbitrary) choice of a starting vertex v_0 induces an order among the vertices and we write $v_0, \ldots, v_{n-1}$ to denote the vertices along the boundary in counter-clockwise order. We define chain $(v_i, v_j) := (v_i, v_{i+1}, \ldots, v_j)$. Note that all indices of vertices are modulo n, unless otherwise specified.

We say two vertices $u, w \in V$ *see* each other or u *sees* w if and only if the line segment $\overline{uw}$ lies entirely in $\mathcal{P}$ – in particular v_i sees v_{i+1} for all i. If v_{i-1} sees v_{i+1}, we say v_i forms an *ear*.

The visibility graph $G(\mathcal{P})$ of a polygon has a node for every vertex of the polygon and an edge for every pair of vertices that see each other. We use m to denote the number of edges in the visibility graph of a given polygon. We write $d_i, 0 \leq i < n$, to denote the degree of v_i, i.e., the number of edges incident to v_i in $G(\mathcal{P})$. For convenience, all operations on indices are understood modulo n.

With $\text{vis}(v_i) = (u_1, \ldots, u_{d_i})$ we denote the sequence of vertices that a vertex $v_i \in V$ of degree d_i sees, enumerated in counter-clockwise order along the boundary starting at $u_1 = v_{i+1}$ and ending at $u_d = v_{i-1}$. We write $\text{vis}_j(v_i)$, $1 \leq j \leq d$, to denote u_j, $\text{vis}_{-j}(v_i)$ to denote u_{d+1-j} and $\text{vis}_0(v_i)$ to denote v_i itself. For a given sequence $\text{chain}(v_i, v_j)$ we denote by $\text{chain}_v(v_i, v_j)$ the subsequence of $\text{chain}(v_i, v_j)$ containing only the vertices visible to v.

Let $C = (u_0, \ldots, u_{l-1})$ be a cycle of length l in the visibility graph of $\mathcal{P}$. We say C is an *ordered cycle*, if and only if $u_0, \ldots, u_{l-1}$ appear on the boundary of $\mathcal{P}$ in that order (counter-clockwise). As u_i sees u_{i+1} for $0 \leq i \leq l-1$, an ordered cycle C induces a subpolygon of $\mathcal{P}$ with C being the boundary of the subpolygon. Note that C being an ordered cycle implies that the boundary of the induced subpolygon does not self-intersect.

LEMMA 7.3. *Let $\mathcal{P}$ be a simple polygon of size $n \geq 4$. For all $0 \leq i < n$ we have that either the degree of v_i or the degree of v_{i+1} is greater than two.*

1.2. Look-back Agents and Angle-type Agents. In the following, we consider a mobile agent exploring a simple polygon $\mathcal{P}$ with n vertices (cf. Figure 7.1). The goal of the agent is to reconstruct the visibility graph $G(\mathcal{P})$ of $\mathcal{P}$.

While located at a vertex v_i, the agent perceives the edges of $G(\mathcal{P})$ incident to v_i in counter-clockwise order, starting with the boundary edge (v_i, v_{i+1}) (cf. Figure 7.1). In other words, the agent is moving in the graph $G(\mathcal{P})$ endowed with a port-numbering where the port-numbers are given by this counter-clockwise order.

We choose v_0 to be the agent's initial location, therefore the agent can keep track of its global position as long as it moves along the boundary only. However, it can neither perceive the global index i of its location v_i directly, nor the global indices of the vertices to which the edges from v_i lead. This means that once it moves along an

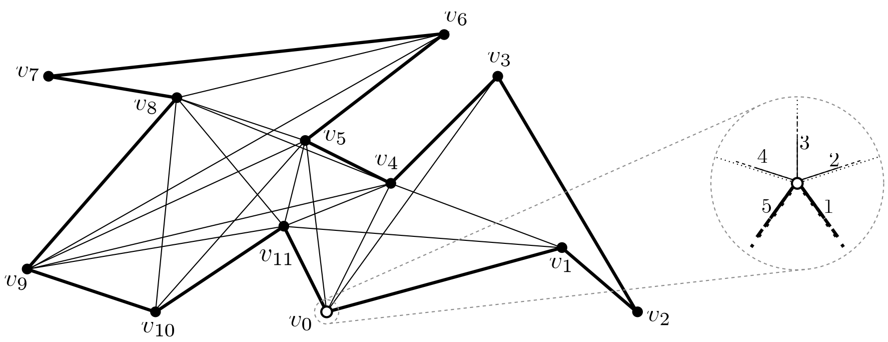


FIGURE 7.1. Left: A simple polygon with embedded visibility graph. Right: The polygon as perceived locally by an agent located at v_0. All that the agent can observe is an order of the edges incident to its location. Except for the two boundary-edges, the agent does not know to which vertex, or in which direction, each edge leads.

edge through the inside of $\mathcal{P}$, in a way, the agent looses sense of its global position. The counter-clockwise ordering of the edges at a vertex is the only means of orientation that the agent has when deciding a move.

The move itself is assumed to be instantaneous, i.e., the agent cannot make any observations while moving. Every movement decision of the agent and the conclusions it draws from local observations are based on all the information it has collected so far – a history of movement decisions and observed vertex degrees. Because our focus is to study the effect of movement and sensing capabilities, we do not restrict the agent computationally, and we assume that the agent has enough memory to store all the history of movements and observations. The question is whether the information collected this way is sufficient for the agent to infer $G(\mathcal{P})$.

In order to reconstruct the visibility graph, it is sufficient to decide for every vertex where the edges incident to this vertex lead in terms of global identities (i.e., global indices). This task becomes trivial if there is a vertex $v^\star$ with the property that the agent can distinguish at any time whether or not it is currently located at $v^\star$. In that case, the agent can decide where an edge leads simply by moving along the edge and then counting the number of moves along the boundary that it takes to get back to $v^\star$. Hence, the visibility graph reconstruction problem is non-trivial only if no individual vertex of $\mathcal{P}$ can be recognized by the agent. In some sense, the problem is difficult only if $\mathcal{P}$ is symmetric with respect to the data which the agent is able to perceive.

Observe that in the algorithm proposed above when the agent can distinguish a particular node, when an agent enters a node, it does not need to be able to backtrack, and thus it does not need to know the label of the edge leading back to its previous location. It is an open problem to determine whether one can reconstruct $G(\mathcal{P})$ from an upper bound on the size of $\mathcal{P}$ in this model when the agent is not given any additional sensor.

A natural way for an agent to be able to backtrack is to to equip the agent with an additional sensor which perceives the label of the arc that leads back to the agent's previous location (cf. Figure 7.2), i.e., in the labeled graph $G(\mathcal{P})$ to allow the agent to detect the incoming port-number when it enters a node (cf. Figure 7.2). We refer to an agent with this capability as a *look-back agent.*

The *standard angle* sensor [**105**, **106**] measures all counter-clockwise angles between pairs of edges of $G(\mathcal{P})$ which are incident to the agent's current location (see. Figure 7.3, left). The *angle-type* sensor is the same as the standard angle sensor, except that angles are not measured exactly: for each angle, the angle-type sensor only returns whether

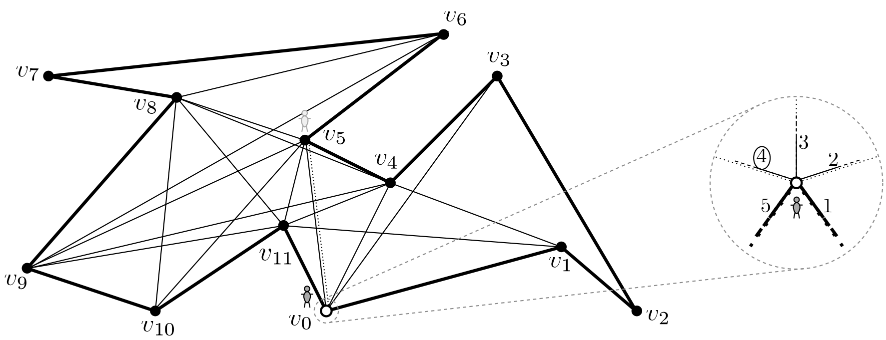


FIGURE 7.2. The perception of the agent with look-back capability after its move from v_5 to v_0. The agent can distinguish the arc leading back to v_5, i.e., it knows its index "4" in the local ordering.

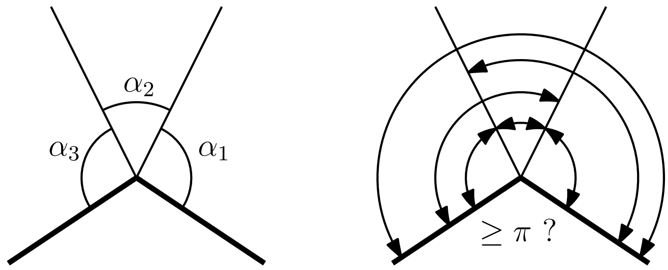


FIGURE 7.3. Local perception with a standard angle sensor (left) and an angle type sensor(right).

this angle is convex ($\leq \pi$) or reflex ($> \pi$) (cf. Figure 7.3, right). We refer to an agent endowed with an angle-type sensor as a *angle-type agent.*

2. Reconstructing a Polygon with a Look-Back Agent

2.1. Computing the Minimum Base. The first step of our algorithm is for the agent to reconstruct the minimum base of $G(\mathcal{P})$. Observe that the agent is able to explore all vertices of $G(\mathcal{P})$ by following the boundary of the polygon, i.e., by following port-numbers 1. We assume that the agent knows initially an upper bound $\hat{n}$ on the number n of vertices of $\mathcal{P}$. Consequently, by following $\hat{n}$ times the port number 1, the agent knows it has explored all vertices of the polygon. Since we consider a look-back agent, one can apply Theorem 6.6 with $N_G = \hat{n}$. In fact, since the agent follows a hamiltonian path during its exploration algorithm, we can replace $N_G\Delta = \hat{n}\Delta$ by $\hat{n}\frac{m}{n} = \hat{n}\frac{m_B}{n_B}$ and we get the following proposition.

PROPOSITION 7.4. *Given an upper bound $\hat{n}$ on the size of a polygon $\mathcal{P}$, a look-back agent can reconstruct the minimum base $B(\mathcal{P})$ of $G(\mathcal{P})$ in $O(\hat{n} \cdot m_B D_B \log(n_B/D_B)$ where m_B, n_B, and D_B are respectively the number of arcs, the number of vertices and the diameter of $B(\mathcal{P})$.*

2.2. Identifying a Clique in $G(\mathcal{P})$. Once the agent has computed $B(\mathcal{P})$, it has gathered all the information it can collect by exploring $G(\mathcal{P})$ (or $\mathcal{P}$). We now explain how one can extract a map of $G(\mathcal{P})$ from $B(\mathcal{P})$. In the following, we say that two vertices are equivalent if they have the same image in $B(\mathcal{P})$, and we denote the equivalence class of a vertex v by $[v]_{\mathcal{P}}$, or $[v]$ when $\mathcal{P}$ is clear from the context. The equivalence classes $[v]$ can be identify with the vertices of $B(\mathcal{P})$. First observe that by Proposition 6.1, all equivalence classes have the same size q. In our particular case, the sequence of classes to which the vertices along the boundary belong is periodical with period $\frac{n}{q}$.

Lemma 7.5. *Let v_i be a vertex of a simple polygon $\mathcal{P}$ of size n. For all vertices $u \in V(\mathcal{P})$ we have $q := |[v_i]| = |[u]|$ and $p := \frac{n}{q} = |V(B(\mathcal{P}))|$ is an integer equal to the number of different classes of $\mathcal{P}$.*

For all integers k we have $[v_i] = [v_{i+kp}]$.

The following simple lemma is the key lemma in the reconstruction of $G(\mathcal{P})$. It shows that one can detect if a vertex v_i is an ear by looking at the labels of path of length at most 2 starting at v_i.

Lemma 7.6. *Let v_i be a vertex of a simple polygon $\mathcal{P}$ of size $n \geq 3$. We have that v_i is an ear if and only if v_{i-1} has a neighbor u such that $\delta_{v_{i-1}}(u) = 2$ and $\delta_u v_{i-1} = -2$.*

Since two vertices are equivalent if and only if they have the same view, we get the following proposition.

Proposition 7.7. *If a vertex v_i of $\mathcal{P}$ is a polygon, then every vertex of $\mathcal{P}$ that is equivalent to v_i is also an ear.*

The following lemma allows an agent to 'cut off' ears of the polygon. With cutting off an ear v_i of a polygon $\mathcal{P}$ we mean the operation that removes a vertex v_i yielding the subpolygon induced by the ordered cycle $v_0, \ldots, v_{i-1}, v_{i+1}, \ldots, v_{n-1}$ in $\mathcal{P}$'s visibility graph. Cutting off a single ear is problematic for an agent as it has no obvious way of deciding which edges of the visibility graph it has to ignore afterwards in order to restrict itself to the remaining subpolygon. An edge might lead to a vertex of the same class as the one the agent cut off, in which case it has no way of distinguishing whether the vertex is still there or not. Cutting off all vertices of one class however is possible as the agent can then simply ignore all edges leading to vertices of the corresponding class altogether.

Lemma 7.8. *Let v be a vertex of a simple polygon $\mathcal{P}$ of size n with $|[v]_{\mathcal{P}}| < n$ (i.e., $\mathcal{P}$ has more than one class). If v is an ear of $\mathcal{P}$, the subpolygon $\mathcal{P}'$ of $\mathcal{P}$ obtained by cutting off the vertices $[v]$ is well-defined and for all vertices u of $\mathcal{P}'$ we have $[u]_{\mathcal{P}} \subseteq [u]_{\mathcal{P}'}$.*

As long as there are at least two classes, the agent can identify a class $[v]$ of ears, and update its map of $B(\mathcal{P})$ by removing $[v]$ and the arcs incident to $[v]$ and by updating the port-numbers of the remaining vertices. When we are left with only one equivalence class, we know that all vertices of $\mathcal{P}$ are ears, i.e., we know that $\mathcal{P}$ is convex and thus $G(\mathcal{P})$ is a clique.

Theorem 7.9. *For any simple polygon $\mathcal{P}$ there is an equivalence class $[v^*]_{\mathcal{P}}$ that forms a clique in the visibility graph of $\mathcal{P}$.*

Moreover $[v^*]_{\mathcal{P}}$ forms a clique in $G(\mathcal{P})$ if and only if there are $|[v^*]_{\mathcal{P}}| - 1$ self-loops at $[v^*]_{\mathcal{P}}$ in $B(\mathcal{P})$.

Observe that even if the agent does not initially know the number n of vertices of $\mathcal{P}$, but only the minimum base $B(\mathcal{P})$ of $G(\mathcal{P})$ (or enough information allowing the agent to compute $B(\mathcal{P})$), then it is able to compute the value of n. Indeed, once it has computed $B(\mathcal{P})$, it looks for the node $[v] \in B(\mathcal{P})$ with the maximum number k of self-loops at $[v]$. Then, the agent knows that $\mathcal{P}$ has $(k+1)|V(B(\mathcal{P}))|$ vertices.

Corollary 7.10. *Given the minimum base $B(\mathcal{P})$ of the visibility graph $G(\mathcal{P})$ of a polygon $\mathcal{P}$ (or an upper bound $\hat{n}$ on the size of $\mathcal{P}$), a look-back agent can compute the size of $\mathcal{P}$*

2.3. Reconstructing $G(\mathcal{P})$. Since the classes of $G(\mathcal{P})$ appear periodically on the boundary of $\mathcal{P}$, and since the agent knows $B(\mathcal{P})$ and the number n of vertices of $\mathcal{P}$, it can reconstruct a cycle of size n and it knows the class $[v]_{\mathcal{P}}$ of each vertex of this cycle. The agent they identify a class $[v^*]$ of vertices that form a clique. Observe that since the port-numbers are given in the ccw order and since $[v^*]$ is a clique, the agent can reconstruct all edges vu with $v \in [v^*]$.

In order to identify the other edges of $G(\mathcal{P})$, the agent proceeds by induction. Assuming that the agent has reconstructed all the edges v_iv_j with $|j - i| < k$, one has to decide whether v_i and v_{i+k} are adjacent. The following lemma shows that one can decide it from the label of the arcs of the minimum base $B(\mathcal{P})$.

LEMMA 7.11. *If v_i is adjacent to p vertices in $\{v_{i+1}, \dots, v_k\}$ and v_{i+k} is adjacent to q vertices in $\{v_{i+1}, \dots, v_k\}$, then v_i is adjacent to v_{i+k} if and only if v_i has a neighbor u such that $\delta_{v_i}(u) = p + 1$ and $\delta_u(v_i) = q + 1$.*

Observe that one direction of the lemma is trivial. The other direction is more involved and relies on the use of the vertices from $[v^*]$. This gives us the following theorem.

THEOREM 7.1 ([**63**, **65**]). *Given an upper bound $\hat{n}$ on the size of a simple polygon $\mathcal{P}$, a look-back agent can reconstruct the visibility graph $G(\mathcal{P})$ of $\mathcal{P}$.*

Observe that if we are given the minimum base $B(\mathcal{P})$ of $G(\mathcal{P})$ instead of an upper bound on the size, one can also reconstruct $\mathcal{P}$ and thus we have the following corollary.

COROLLARY 7.12. *For any digraph D, there is at most one polygon $\mathcal{P}$ such that D is the minimum base $B(\mathcal{P})$ of $G(\mathcal{P})$.*

Observe that once the agent has computed the minimum base $B(\mathcal{P})$ of the visibility graph $G(\mathcal{P})$ of the polygon $\mathcal{P}$, the agent does not need to move anymore in order to compute $G(\mathcal{P})$. Consequently, the agent can reconstruct the visibility graph $G(\mathcal{P})$ of $\mathcal{P}$ in time $O(\hat{n} \cdot m_B D_B \log(n_B/D_B))$ where m_B, n_B, and D_B are respectively the number of arcs, the number of vertices and the diameter of $B(\mathcal{P})$.

3. Reconstructing a Polygon with an Angle-type Agent

The general idea of the algorithm is similar to the one presented in the previous section. Using the method described in Section 5 of Chapter 6, the agent can reconstruct the minimum base $B(\mathcal{P})$ of the visibility graph $G(\mathcal{P})$ of the polygon $\mathcal{P}$ where each node v of $G(\mathcal{P})$ is labeled by the observations that the agent can make at v. In our case, the observations an agent can make at a node v are the angle types of all angles incident to v, but the following proposition holds for any kind of observation the agent can make.

PROPOSITION 7.13. *An angle-type agent can reconstruct the minimum base $B(\mathcal{P})$ of the visibility graph $G(\mathcal{P})$ of the polygon $\mathcal{P}$.*

As in the previous section, we denote by $[v]_\mathcal{P}$ (or simply $[v]$) the equivalence class of a vertex v of $G(\mathcal{P})$ and as before, one can identify these equivalence classes with the vertices of $B(\mathcal{P})$. Note that contrary to the previous section, $Dir(G(\mathcal{P}))$ is an opfibration of $B(\mathcal{P})$ and not a cover. However, the different classes all appear periodically on the boundary and consequently, all classes have the same size.

Even if the result is similar to Proposition 7.7, the proof is completely different and more involved. It relies on the fact that if two nodes are equivalent, the agent makes the same angle-type observations at these two nodes.

PROPOSITION 7.14. *If a vertex v of $G(\mathcal{P})$ is an ear, then all vertices in $[v]_\mathcal{P}$ are ears.*

Cutting of classes of ears by considering the minimum base $B(\mathcal{P})$, we get the following theorem as in the previous section.

THEOREM 7.15. *For any simple polygon $\mathcal{P}$ there is an equivalence class $[v^*]_\mathcal{P}$ that forms a clique in the visibility graph of $\mathcal{P}$.*

Moreover $[v^*]_\mathcal{P}$ forms a clique in $G(\mathcal{P})$ if and only if there are $|[v^*]_\mathcal{P}| - 1$ self-loops at $[v^*]_\mathcal{P}$ in $B(\mathcal{P})$.

As before, observe that this theorem allows an agent to compute the size of the polygon $\mathcal{P}$ from the minimum base $B(\mathcal{P})$ (or an upper bound $\hat{n}$ on its size).

Finally, we reconstruct the graph $G(\mathcal{P})$. The method is different from the one presented in the previous section. Here we add back the classes of vertices that have been removed one after the other in the same order as the one in which they have been cut off to find the class $[v^*]$. Let $[v^*] = [v_0], [v_1], \ldots, [v_{n_B}]$ be the classes of vertices and assume that for each $1 \leq i \leq n_B$, $[v_i]$ is a class of ears in the polygon $\mathcal{P}_i$ induced by the vertices in $\bigcup_{j \leq i} [v_i j$. We show that we can reconstruct $G(\mathcal{P}_{i+1})$ from $G(\mathcal{P}_i)$ and we get the following theorem.

Theorem 7.2 ([**64, 67**]). *Given an upper bound $\hat{n}$ on the size of a simple polygon $\mathcal{P}$, an angle-type agent can reconstruct the visibility graph $G(\mathcal{P})$ of $\mathcal{P}$.*

Contrary to the algorithm presented in the previous section, the number of moves performed by an agent executing our algorithm is exponential in the upper bound $\hat{n}$ on the size of $\mathcal{P}$. Indeed, the minimum base construction algorithm presented in Section 5 of Chapter 6 requires an exponential number of moves.

4. Conclusion

Our solutions for the mapping problem with look-back agent and with an angle-type agent follow the same approach:

(1) Construct the minimum base $B(\mathcal{P})$ of the visibility graph $G(\mathcal{P})$ of the polygon $\mathcal{P}$ (where the labels of $G(\mathcal{P})$ contain all the information available to the agent).
(2) Establishing that if v is an ear of $\mathcal{P}$, then all vertices in its equivalence class $[v]_{\mathcal{P}}$ are ears. This implies that there exists an equivalence class $[v^*]_{\mathcal{P}}$ that forms a clique in $G(\mathcal{P})$.
(3) Showing that starting from the clique $[v^*]_{\mathcal{P}}$, one can reconstruct $G(\mathcal{P})$ from $B(\mathcal{P})$.

Note also that in both cases, the agent is able to compute the size n of the polygon from an upper bound $\hat{n}$. This is impossible when we consider general families of graphs (consider the family of cycles for example).

A natural question is to identify other types of sensors we can give to the agent so that the previous approach can be applied. In particular, one can ask whether any information is really necessary besides the knowledge of an upper bound on the size of the polygon.

Conjecture 7.16. *The knowledge of an upper bound $\hat{n}$ on the size of $\mathcal{P}$ allow an agent to reconstruct the visibility graph $G(\mathcal{P})$.*

Observe that in this case, we can reconstruct the minimum base $B(\mathcal{P})$ of $G(\mathcal{P})$ as explained in Section 3 and that $Dir(G(\mathcal{P}))$ is an opfibration of $B(\mathcal{P})$ via some homomorphism φ. We can also show that all equivalence classes have the same size in this case. In order to solve Question 7.16, it should be enough to establish that φ is a covering map. Indeed, in this case, we would be able to reuse the techniques of Section 3 to establish Property (2) and (3). In [**103**], Disser established that if the minimum base $B(\mathcal{P})$ has only one vertex (i.e., if $G(\mathcal{P})$ is regular), then $G(\mathcal{P})$ is a complete graph and $\mathcal{P}$ is a convex polygon. This implies that in this very particular case, φ is a covering map.

When we consider look-back agents, we know that without any information, the agent cannot detect when it has visited all the vertices of the polygon [**45**], and thus the knowledge of an upper bound $\hat{n}$ on the size of the polygon (or some initial knowledge allowing the agent to compute such a bound) is necessary. However, for angle-type agent, we do not have such an impossibility result and thus one can ask the following question.

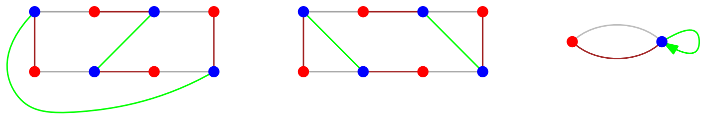


FIGURE 7.4. Two planar graphs (left and center) that have the same minimum base (right).

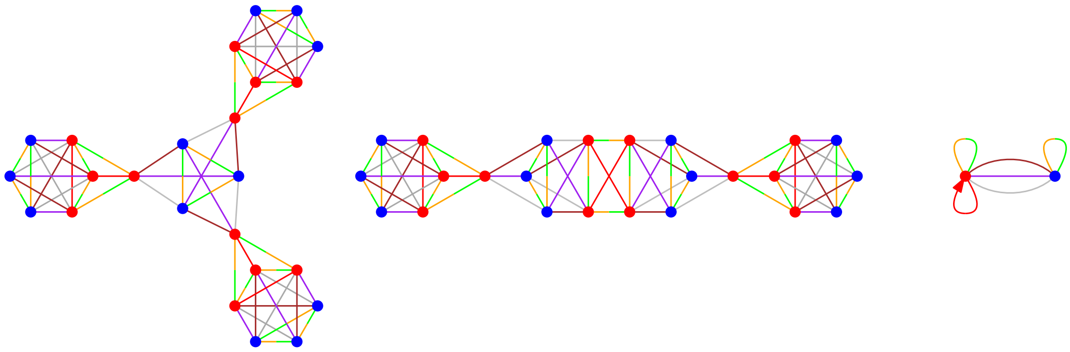


FIGURE 7.5. Two chordal graphs (left and center) that have the same minimum base (right).

QUESTION 7.17. *Can an angle-type agent reconstruct the visibility graph $G(\mathcal{P})$ of the polygon $\mathcal{P}$ without initial information?*

Observe that from what we have shown above, this is equivalent to ask whether an angle-type agent can compute an upper bound on the size of the polygon.

When we consider agents that can only move on the boundary, we do not know whether angle-types are sufficient to reconstruct the visibility graph. Since angle-types sensors seem quite weak, we suspect the answer to be negative, but we do not have any counter-example.

Another interesting open problem when the agent is restricted to move on the boundary of the polygon is when the agent can measure the distances to the other vertices of the polygon. We assume that when an agent is at a vertex v_i with $\text{vis}(v_i) = (u_1, \ldots, u_{d_i})$, it can access the ordered list of distances $(d(v_i, u_1), \ldots, d(v_i, u_{d_i}))$.

QUESTION 7.18. *Can an agent measuring distances reconstruct the polygon $\mathcal{P}$?*

Note that if the agent additionally knows the size of the polygon, then this problem is an offline problem that is also widely open. In fact, even if the agent can move freely in the polygon, we do not know whether being able to measure distances is enough to reconstruct the polygon.

Another direction of research would be to identify other natural classes of graphs where it is possible to reconstruct the graph from the minimum base. In other words, we would like to identify some class $\mathcal{F}$ of graphs such that for any digraph D, there is at most one graph $G \in \mathcal{F}$ such that D is the minimum base of G.

The graphs on Figures 7.4 and 7.5 show that the families of chordal graphs and planar graphs do not satisfy this property. Observe that even if we fix the size of the graph, the graphs on Figures 7.4 and 7.5 are still counterexamples.

Bibliography

1. M. Abu-Ata and F. F. Dragan, *Metric tree-like structures in real-world networks: an empirical study*, Networks **67** (2016), no. 1, 49–68.
2. A. B. Adcock, B. D. Sullivan, and M. W. Mahoney, *Tree-like structure in large social and information networks*, ICDM 2013, IEEE Computer Society, 2013, pp. 1–10.
3. N. Agmon and D. Peleg, *Fault-tolerant gathering algorithms for autonomous mobile robots*, SIAM J. Comput. **36** (2006), no. 1, 56–82.
4. I. Agol, *The virtual Haken conjecture*, Doc. Math. **18** (2013), 1045–1087, With an appendix by Agol, Daniel Groves, and Jason Manning.
5. ———, *Virtual properties of 3-manifolds*, Proceedings of the International Congress of Mathematicians 2014 (Seoul), vol. I, 2014, pp. 141–170.
6. R. Aleliunas, R. M. Karp, Lipton R. J., L. Lovász, and C. Rackoff, *Random walks, universal traversal sequences, and the complexity of maze problems*, FOCS 1979, IEEE Computer Society, 1979, pp. 218–223.
7. J. M. Alonso, T. Brady, D. Cooper, V. Ferlini, M. Lustig, M. Mihalik, M. Shapiro, and H. Short, *Notes on word hyperbolic groups*, Group Theory from a Geometrical Viewpoint (E. Ghys, A. Haefliger, and A. Verjovsky, eds.), World Scientific, 1991, ICTP Trieste 1990, pp. 3–63.
8. H. Ando, Y. Oasa, I. Suzuki, and M. Yamashita, *Distributed memoryless point convergence algorithm for mobile robots with limited visibility*, IEEE Trans. Robotics and Automation **15** (1999), no. 5, 818–828.
9. D. Angluin, *Local and global properties in networks of processors (extended abstract)*, STOC 1980, ACM, 1980, pp. 82–93.
10. R. P. Anstee, L. Rónyai, and A. Sali, *Shattering news*, Graphs Combin. **18** (2002), no. 1, 59–73.
11. F. Ardila, M. Owen, and S. Sullivant, *Geodesics in CAT(0) cubical complexes*, Adv. in Appl. Math. **48** (2012), no. 1, 142–163.
12. M. R. Assous, V. Bouchitté, C. Charretton, and B. Rozoy, *Finite labelling problem in event structures*, Theoret. Comput. Sci. **123** (1994), no. 1, 9–19.
13. S. P. Avann, *Metric ternary distributive semi-lattices*, Proc. Amer. Math. Soc. **12** (1961), no. 3, 407–414.
14. E. Badouel, Ph. Darondeau, and J.-C. Raoult, *Context-free event domains are recognizable*, Inform. and Comput. **149** (1999), no. 2, 134–172.
15. H.-J. Bandelt, *Retracts of hypercubes*, J. Graph Theory **8** (1984), no. 4, 501–510.
16. H.-J. Bandelt and V. Chepoi, *A Helly theorem in weakly modular space*, Discrete Math. **160** (1996), no. 1-3, 25–39.
17. ———, *Decomposition and l_1-embedding of weakly median graphs*, European J. Combin. **21** (2000), no. 6, 701–714, Discrete metric spaces (Marseille, 1998).
18. ———, *Metric graph theory and geometry: a survey*, Surveys on Discrete and Computational Geometry: Twenty Years Later (J. E. Goodman, J. Pach, and R. Pollack, eds.), Contemp. Math., vol. 453, Amer. Math. Soc., Providence, RI, 2008, pp. 49–86.
19. H.-J. Bandelt, V. Chepoi, A. W. M. Dress, and J. H. Koolen, *Combinatorics of lopsided sets*, European J. Combin. **27** (2006), no. 5, 669–689.
20. H.-J. Bandelt, V. Chepoi, and K. Knauer, *COMs: Complexes of oriented matroids*, J. Combin. Theory Ser. A **156** (2018), 195–237.
21. H.-J. Bandelt and H. M. Mulder, *Distance-hereditary graphs*, J. Combin. Theory Ser. B **41** (1986), no. 2, 182–208.
22. H.-J. Bandelt, H. M. Mulder, and E. Wilkeit, *Quasi-median graphs and algebras*, J. Graph Theory **18** (1994), no. 7, 681–703.
23. H.-J. Bandelt and E. Pesch, *Dismantling absolute retracts of reflexive graphs*, European J. Combin. **10** (1989), no. 3, 211–220.
24. H.-J. Bandelt and E. Prisner, *Clique graphs and Helly graphs*, J. Combin. Theory Ser. B **51** (1991), no. 1, 34–45.
25. J.-P. Barthélemy and J. Constantin, *Median graphs, parallelism and posets*, Discrete Math. **111** (1993), no. 1-3, 49–63, Graph Theory and Combinatorics (Marseille-Luminy, 1990).

26. M. A. Bednarczyk, *Categories of asynchronous systems*, Ph.D. thesis, University of Sussex, 1987.
27. R. Berger, *The undecidability of the domino problem*, Mem. Amer. Math. Soc. **66** (1966).
28. M. Bestvina, M. Sageev, and K. Vogtmann (eds.), *Geometric group theory*, IAS/Park City Math. Ser., vol. 21, Amer. Math. Soc., Inst. Adv. Study, 2012.
29. D. Bilò, Y. Disser, M. Mihalák, S. Suri, E. Vicari, and P. Widmayer, *Reconstructing visibility graphs with simple robots*, Theoret. Comput. Sci. **444** (2012), 52–59.
30. A. Björner, M. Las Vergnas, B. Sturmfels, N. White, and G. M. Ziegler, *Oriented matroids*, Encyclopedia Math. Appl., vol. 46, Cambridge Univ. Press, Cambridge, 1993.
31. P. Boldi, S. Shammah, S. Vigna, B. Codenotti, P. Gemmell, and J. Simon, *Symmetry breaking in anonymous networks: Characterizations*, ISTCS 1996, IEEE Computer Society, 1996, pp. 16–26.
32. P. Boldi and S. Vigna, *Computing anonymously with arbitrary knowledge*, PODC 1999, ACM, 1999, pp. 181–188.
33. ———, *An effective characterization of computability in anonymous networks*, DISC 2001, Lecture Notes in Comput. Sci., vol. 2180, Springer, 2001, pp. 33–47.
34. ———, *Fibrations of graphs*, Discrete Math. **243** (2002), no. 1-3, 21–66.
35. B. Bollobás and A. J. Radcliffe, *Defect Sauer results*, J. Combin. Theory Ser. A **72** (1995), no. 2, 189–208.
36. M. Borassi, D. Coudert, P. Crescenzi, and A. Marino, *On computing the hyperbolicity of real-world graphs*, ESA 2015, Lecture Notes in Comput. Sci., vol. 9294, Springer, 2015, pp. 215–226.
37. M. Borassi, P. Crescenzi, and M. Habib, *Into the square: On the complexity of some quadratic-time solvable problems*, Electron. Notes Theor. Comput. Sci. **322** (2016), 51–67, ICTCS 2015.
38. B. H. Bowditch, *Notes on Gromov's hyperbolicity criterion for path-metric spaces*, Group Theory from a Geometrical Viewpoint (E. Ghys, A. Haefliger, and A. Verjovsky, eds.), World Scientific, 1991, ICTP Trieste 1990, pp. 64–167.
39. B. Brešar, J. Chalopin, V. Chepoi, T. Gologranc, and D. Osajda, *Bucolic complexes*, Adv. Math. **243** (2013), 127–167.
40. B. Brešar, J. Chalopin, V. Chepoi, M. Kovše, A. Labourel, and Y. Vaxès, *Retracts of products of chordal graphs*, J. Graph Theory **73** (2013), no. 2, 161–180.
41. M. R. Bridson, *Cube complexes, subgroups of mapping class groups, and nilpotent genus*, Geometric Group Theory (M. Bestvina, M. Sageev, and K. Vogtmann, eds.), IAS/Park City Math. Ser., vol. 21, Amer. Math. Soc., Inst. Adv. Study, 2012, pp. 381–399.
42. M. R. Bridson and A. Haefliger, *Metric spaces of non-positive curvature*, Grundlehren Math. Wiss., vol. 319, Springer-Verlag, Berlin, 1999.
43. M. R. Bridson and H. Wilton, *On the recognition problem for virtually special cube complexes*, Hyperbolic geometry and geometric group theory, Adv. Stud. Pure Math., vol. 73, Math. Soc. Japan, Tokyo, 2017, pp. 37–46. MR 3728492
44. A. E. Brouwer and A. M. Cohen, *Local recognition of Tits geometries of classical type*, Geom. Dedicata **20** (1986), no. 2, 181–199.
45. J. Brunner, M. Mihalák, S. Suri, E. Vicari, and P. Widmayer, *Simple robots in polygonal environments: A hierarchy*, ALGOSENSORS 2008, Lecture Notes in Comput. Sci., vol. 5389, Springer, 2008, pp. 111–124.
46. J. P. Burling, *On coloring problems of prototypes*, Ph.D. thesis, University of Colorado, Boulder, 1965.
47. P. J. Cameron, *Dual polar spaces*, Geom. Dedicata **12** (1982), no. 1, 75–85.
48. J. W. Cannon, *The combinatorial structure of cocompact discrete hyperbolic groups*, Geom. Dedicata **16** (1984), no. 2, 123–148.
49. J. Chalopin, *Algorithmique distribuée, calculs locaux et homomorphismes de graphes*, Ph.D. thesis, Université Bordeaux 1, 2006.
50. J. Chalopin and V. Chepoi, *A counterexample to Thiagarajan's conjecture on regular event structures*, ICALP 2017, LIPIcs, vol. 80, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2017, pp. 101:1–101:14.
51. ———, *A counterexample to Thiagarajan's conjecture on regular event structures*, arXiv preprint **1605.08288v3** (2018).
52. ———, *1-safe Petri nets and special cube complexes: Equivalence and applications*, ACM Trans. Comput. Log. **20** (2019), no. 3, 17:1–17:49.
53. J. Chalopin, V. Chepoi, F. F. Dragan, G. Ducoffe, A. Mohammed, and Y. Vaxès, *Fast approximation and exact computation of negative curvature parameters of graphs*, SoCG 2018, LIPIcs, vol. 99, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2018, pp. 22:1–22:15.
54. ———, *Fast approximation and exact computation of negative curvature parameters of graphs*, Discrete Comput. Geom. (2019), to appear.
55. J. Chalopin, V. Chepoi, A. Genevois, H. Hirai, and D. Osajda, *Helly groups*, arXiv preprint **1107.5789v7** (2020).

56. J. Chalopin, V. Chepoi, H. Hirai, and D. Osajda, *Weakly modular graphs and nonpositive curvature*, Mem. Amer. Math. Soc. (2017), to appear.
57. J. Chalopin, V. Chepoi, S. Moran, and M. K. Warmuth, *Unlabeled sample compression schemes and corner peelings for ample and maximum classes*, arXiv preprint **1812.02099** (2018).
58. ______, *Unlabeled sample compression schemes and corner peelings for ample and maximum classes*, ICALP 2019, LIPIcs, vol. 132, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2019, pp. 29:1–29:15.
59. J. Chalopin, V. Chepoi, N. Nisse, and Y. Vaxès, *Cop and robber games when the robber can hide and ride*, SIAM J. Discrete Math. **25** (2011), no. 1, 333–359.
60. J. Chalopin, V. Chepoi, and D. Osajda, *Proof of two Maurer's conjectures on basis graphs of matroids*, J. Combin. Theory Ser. B **114** (2015), 1–32.
61. J Chalopin, V. Chepoi, P. Papasoglu, and T. Pecatte, *Cop and robber game and hyperbolicity*, SIAM J. Discrete Math. **28** (2014), no. 4, 1987–2007.
62. J. Chalopin and S. Das, *Rendezvous of mobile agents without agreement on local orientation*, ICALP 2010, Lecture Notes in Comput. Sci., vol. 6199, Springer, 2010, pp. 515–526.
63. J. Chalopin, S. Das, Y. Disser, M. Mihalák, and P. Widmayer, *How simple robots benefit from looking back*, CIAC 2010, Lecture Notes in Comput. Sci., vol. 6078, Springer, 2010, pp. 229–239.
64. ______, *Telling convex from reflex allows to map a polygon*, STACS 2011, LIPIcs, vol. 9, Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2011, pp. 153–164.
65. ______, *Mapping simple polygons: How robots benefit from looking back*, Algorithmica **65** (2013), no. 1, 43–59.
66. ______, *Simple agents learn to find their way: An introduction on mapping polygons*, Discrete Appl. Math. **161** (2013), no. 10–11, 1287–1307.
67. ______, *Mapping simple polygons: The power of telling convex from reflex*, ACM Trans. Algorithms **11** (2015), no. 4, 33, 15 pages.
68. J. Chalopin, S. Das, and A. Kosowski, *Constructing a map of an anonymous graph: Applications of universal sequences*, OPODIS 2010, Lecture Notes in Comput. Sci., vol. 6490, Springer, 2010, pp. 119–134.
69. J. Chalopin, S. Das, and N. Santoro, *Rendezvous of mobile agents in unknown graphs with faulty links*, DISC 2007, Lecture Notes in Comput. Sci., vol. 4731, Springer, 2007, pp. 108–122.
70. J. Chalopin, L. Esperet, Z. Li, and P. Ossona de Mendez, *Restricted frame graphs and a conjecture of Scott*, Electron. J. Combin. **23** (2016), no. 1, P1.30.
71. J. Chalopin, E. Godard, and Y. Métivier, *Local terminations and distributed computability in anonymous networks*, DISC 2008, Lecture Notes in Comput. Sci., vol. 5218, Springer, 2008, pp. 47–62.
72. ______, *Election in partially anonymous networks with arbitrary knowledge in message passing systems*, Distrib. Comput. **25** (2012), no. 4, 297–311.
73. J. Chalopin, E. Godard, and Y. Métivier, *Election in partially anonymous networks with arbitrary knowledge in message passing systems*, Distrib. Comput. **25** (2012), no. 4, 297–311.
74. J. Chalopin, E. Godard, and A. Naudin, *Using binoculars for fast exploration and map construction in chordal graphs and extensions*, in preparation.
75. ______, *Anonymous graph exploration with binoculars*, DISC 2015, Lecture Notes in Comput. Sci., vol. 9363, Springer, 2015, pp. 107–122.
76. ______, *Utiliser des jumelles pour explorer rapidement les graphes triangulés*, ALGOTEL 2016, 2016.
77. J. Chalopin and Y. Métivier, *An efficient message passing election algorithm based on Mazurkiewicz's algorithm*, Fund. Inform. **80** (2007), no. 1-3, 221–246.
78. M. Chastand, *Fiber-complemented graphs I: Structure and invariant subgraphs*, Discrete Math. **226** (2001), no. 1-3, 107–141.
79. ______, *Fiber-complemented graphs II: Retractions and endomorphisms*, Discrete Math. **268** (2003), no. 1-3, 81–101.
80. V. Chepoi, *Classification of graphs by means of metric triangles*, Metody Diskret. Analiz. (1989), no. 49, 75–93, 96 (Russian).
81. ______, *Bridged graphs are cop-win graphs: an algorithmic proof*, J. Combin. Theory Ser. B **69** (1997), no. 1, 97–100.
82. ______, *Graphs of some CAT(0) complexes*, Adv. in Appl. Math. **24** (2000), no. 2, 125–179.
83. ______, *Nice labeling problem for event structures: a counterexample*, SIAM J. Comput. **41** (2012), no. 4, 715–727.
84. V. Chepoi, F. F. Dragan, B. Estellon, M. Habib, and Y Vaxès, *Diameters, centers, and approximating trees of δ-hyperbolic geodesic spaces and graphs*, SoCG 2008, ACM, 2008, pp. 59–68.

85. V. Chepoi, F. F. Dragan, B. Estellon, M. Habib, Y. Vaxès, and Y. Xiang, *Additive spanners and distance and routing labeling schemes for hyperbolic graphs*, Algorithmica **62** (2012), no. 3-4, 713–732.
86. V. Chepoi, F. F. Dragan, and Y. Vaxès, *Core congestion is inherent in hyperbolic networks*, SODA 2017, SIAM, 2017, pp. 2264–2279.
87. V. Chepoi and B. Estellon, *Packing and covering δ-hyperbolic spaces by balls*, APPROX-RANDOM 2007, Lecture Notes in Comput. Sci., vol. 4627, Springer, 2007, pp. 59–73.
88. V. Chepoi and M. F. Hagen, *On embeddings of CAT(0) cube complexes into products of trees via colouring their hyperplanes*, J. Combin. Theory Ser. B **103** (2013), no. 4, 428–467.
89. V. Chepoi and D. Osajda, *Dismantlability of weakly systolic complexes and applications*, Trans. Amer. Math. Soc. **367** (2015), no. 2, 1247–1272.
90. M. Clay and D. Margalit (eds.), *Office hours with a geometric group theorist*, Princeton Univ. Press, 2017.
91. N. Cohen, D. Coudert, and A. Lancin, *On computing the Gromov hyperbolicity*, ACM J. of Exp. Algorithmics **20** (2015), 1.6:1–1.6:18.
92. R. Cohen and D. Peleg, *Convergence of autonomous mobile robots with inaccurate sensors and movements*, SIAM J. Comput. **38** (2008), no. 1, 276–302.
93. D. Coudert and G. Ducoffe, *Recognition of C_4-free and 1/2-hyperbolic graphs*, SIAM J. Discrete Math. **28** (2014), no. 3, 1601–1617.
94. D. Coudert, G. Ducoffe, and A. Popa, *Fully polynomial FPT algorithms for some classes of bounded clique-width graphs*, SODA 2018, SIAM, 2018, pp. 2765–2784.
95. B. Courcelle, *The monadic second-order logic of graphs II: Infinite graphs of bounded width*, Math. Systems Theory **21** (1989), no. 4, 187–221.
96. ———, *The monadic second-order logic of graphs VI: On several representations of graphs by relational structures*, Discrete Appl. Math. **54** (1994), no. 2, 117–149.
97. B. Courcelle and J. Engelfriet, *Graph structure and monadic second-order logic*, Encyclopedia Math. Appl., vol. 138, Cambridge Univ. Press, Cambridge, 2012.
98. S. Das, *Mobile agents in distributed computing: Network exploration*, Bulletin of the EATCS **109** (2013), 54–69.
99. S. Das, P. Flocchini, S. Kutten, A. Nayak, and N. Santoro, *Map construction of unknown graphs by multiple agents*, Theoret. Comput. Sci. **385** (2007), no. 1-3, 34–48.
100. B. DasGupta, M. Karpinski, N. Mobasheri, and F. Yahyanejad, *Effect of Gromov-hyperbolicity parameter on cuts and expansions in graphs and some algorithmic implications*, Algorithmica **80** (2018), no. 2, 772–800.
101. T. Delzant and M. Gromov, *Courbure mésoscopique et théorie de la toute petite simplification*, J. Topol. **1** (2008), no. 4, 804–836.
102. D. Dereniowski, A. Kosowski, and D. Pajak, *Distinguishing views in symmetric networks: A tight lower bound*, Theoret. Comput. Sci. **582** (2015), 27–34.
103. Y. Disser, *Mapping polygons*, Ph.D. thesis, ETH Zürich, 2011.
104. Y. Disser, S. K. Ghosh, M. Mihalák, and P. Widmayer, *Mapping a polygon with holes using a compass*, Theoret. Comput. Sci. **553** (2014), 106–113.
105. Y. Disser, M. Mihalák, and P. Widmayer, *A polygon is determined by its angles*, Comput. Geom. **44** (2011), no. 8, 418–426.
106. ———, *Mapping polygons with agents that measure angles*, WAFR 2012, Springer Tracts Adv. Robot., vol. 86, Springer, 2012, pp. 415–425.
107. D. Ž. Djoković, *Distance-preserving subgraphs of hypercubes*, J. Combin. Theory Ser. B **14** (1973), no. 3, 263–267.
108. T. Doliwa, G. Fan, H. U. Simon, and S. Zilles, *Recursive teaching dimension, VC-dimension and sample compression*, J. Mach. Learn. Res. **15** (2014), no. 1, 3107–3131.
109. J. D. Donald, C. A. Holzmann, and M. D. Tobey, *A characterization of complete matroid base graphs*, J. Combin. Theory Ser. B **22** (1977), no. 2, 139–158.
110. A. W. M. Dress, *Towards a theory of holistic clustering*, Mathematical Hierarchies and Biology, DIMACS Ser. Discrete Math. Theoret. Comput. Sci., vol. 37, DIMACS, Amer. Math. Soc., 1996, pp. 271–290.
111. A. W. M. Dress and R. Scharlau, *Gated sets in metric spaces*, Aequationes Math. **34** (1987), no. 1, 112–120.
112. C. Druţu and M. Kapovich, *Geometric group theory*, Amer. Math. Soc. Colloq. Publ., vol. 63, Amer. Math. Soc., 2018.
113. R. Duan, *Approximation algorithms for the Gromov hyperbolicity of discrete metric spaces*, LATIN 2014, Lecture Notes in Comput. Sci., vol. 8392, Springer, 2014, pp. 285–293.
114. P. H. Edelman and R. E. Jamison, *The theory of convex geometries*, Geom. Dedicata **19** (1985), no. 3, 247–270.

115. K. Edwards, W. S. Kennedy, and I. Saniee, *Fast approximation algorithms for p-centers in large δ-hyperbolic graphs*, WAW 2016, Lecture Notes in Comput. Sci., vol. 10088, 2016, pp. 60–73.
116. D. B. A. Epstein, J. W. Cannon, D. F. Holt, S. V. F. Levy, M. S. Paterson, and W. P. Thurston, *Word processing in groups*, Jones and Bartlett, Boston, MA, 1992.
117. M. Farber and R. E. Jamison, *On local convexity in graphs*, Discrete Math. **66** (1987), no. 3, 231–247.
118. S. Floyd, *On space bounded learning and the Vapnik-Chervonenkis dimension*, Ph.D. thesis, International Computer Science Institut, Berkeley, CA, 1989.
119. S. Floyd and M. K. Warmuth, *Sample compression, learnability, and the Vapnik-Chervonenkis dimension*, Mach. Learn. **21** (1995), no. 3, 269–304.
120. T. Fluschnik, C. Komusiewicz, G. B. Mertzios, A. Nichterlein, R. Niedermeier, and N. Talmon, *When can graph hyperbolicity be computed in linear time?*, WADS 2017, Lecture Notes in Comput. Sci., vol. 10389, Springer, 2017, pp. 397–408.
121. R. Forman, *Morse theory for cell complexes*, Adv. Math. **134** (1998), no. 1, 90–145.
122. H. Fournier, A. Ismail, and A. Vigneron, *Computing the Gromov hyperbolicity of a discrete metric space*, Inform. Process. Lett. **115** (2015), no. 6-8, 576–579.
123. B. Gärtner and E. Welzl, *Vapnik-Chervonenkis dimension and (pseudo-)hyperplane arrangements*, Discrete Comput. Geom. **12** (1994), no. 4, 399–432.
124. I. M. Gel′fand, R. M. Goresky, R. D. MacPherson, and V. V. Serganova, *Combinatorial geometries, convex polyhedra, and Schubert cells*, Adv. Math. **63** (1987), no. 3, 301–316.
125. E Ghys and P. de la Harpe (eds.), *Les groupes hyperboliques d'après M. Gromov*, Progr. Math., vol. 83, Birkhäuser, 1990.
126. E. Godard, *Réécritures de graphes et algorithmique distribuée*, Ph.D. thesis, Université Bordeaux 1, 2002.
127. E. Godard and Y. Métivier, *A characterization of families of graphs in which election is possible*, FOSSACS 2002, Lecture Notes in Comput. Sci., vol. 2303, Springer, 2002, pp. 159–172.
128. M. Gromov, *Hyperbolic groups*, Essays in Group Theory (S. M. Gersten, ed.), Math. Sci. Res. Inst. Publ., vol. 8, Springer, New York, 1987, pp. 75–263.
129. M. F. Hagen, *Weak hyperbolicity of cube complexes and quasi-arboreal groups*, J. Topol. **7** (2014), no. 2, 385–418.
130. F. Haglund, *Complexes simpliciaux hyperboliques de grande dimension*, Tech. report, Université d'Orsay, 2003.
131. ———, *Aspects combinatoires de la théorie géométrique des groupes*, HDR, Université Paris Sud, 2008.
132. F. Haglund and D. T. Wise, *Special cube complexes*, Geom. Funct. Anal. **17** (2008), no. 5, 1551–1620.
133. ———, *A combination theorem for special cube complexes*, Ann. of Math. **176** (2012), no. 3, 1427–1482.
134. W. Haken, *Connections between topological and group theoretical decision problems*, Word Problems: Decision Problems and the Burnside Problem in Group Theory (W. W. Boone, F. B. Cannonito, and R. C. Lyndon, eds.), Stud. Logic Found. Math., vol. 71, North-Holland, 1973, pp. 427–441.
135. H. T. Hall, *Counterexamples in discrete geometry*, Ph.D. thesis, University of California, 2004.
136. R. Hammack, W. Imrich, and S. Klavžar, *Handbook of product graphs*, 2nd ed., Discrete Math. Appl., CRC press, Boca Raton, 2011.
137. A. Hatcher, *Algebraic topology*, Cambridge Univ. Press, Cambridge, 2002.
138. P. Hell and I. Rival, *Absolute retracts and varieties of reflexive graphs*, Canad. J. Math. **39** (1987), no. 3, 544–567.
139. J. M. Hendrickx, *Views in a graph: To which depth must equality be checked?*, IEEE Trans. Parallel Distrib. Syst. **25** (2014), no. 7, 1907–1912.
140. N. Hoda, *Crystallographic Helly groups*, in preparation.
141. C. A. Holzmann, P. G. Norton, and M. D. Tobey, *A graphical representation of matroids*, SIAM J. Appl. Math. **25** (1973), no. 4, 618–627.
142. J. E. Hopcroft, *An $n \log n$ algorithm for minimizing states in a finite automaton*, Theory of machines and computations, 1971, pp. 189–196.
143. J. Huang and D. Osajda, *Helly meets Garside and Artin*, arXiv preprint **1904.09060** (2019).
144. J. R. Isbell, *Six theorems about injective metric spaces*, Comment. Math. Helv. **39** (1964), no. 1, 65–76.
145. ———, *Median algebra*, Trans. Amer. Math. Soc. **260** (1980), no. 2, 319–362.
146. T. Januszkiewicz and J. Świątkowski, *Simplicial nonpositive curvature*, Publ. Math. Inst. Hautes Études Sci. **104** (2006), 1–85.
147. J. Kari and P. Papasoglu, *Deterministic aperiodic tile sets*, Geom. Funct. Anal. **9** (1999), no. 2, 353–369.

148. M. Katsev, A. Yershova, B. Tovar, R. Ghrist, and S. M. LaValle, *Mapping and pursuit-evasion strategies for a simple wall-following robot*, IEEE Trans. Robotics **27** (2011), no. 1, 113–128.
149. W. S. Kennedy, I. Saniee, and O. Narayan, *On the hyperbolicity of large-scale networks and its estimation*, BigData 2016, IEEE, 2016, pp. 3344–3351.
150. A. Kosowski, *Time and space-efficient algorithms for mobile agents in an anonymous network*, HDR, Université de Bordeaux, 2013.
151. ———, *A $\widetilde{O}(n^2)$ time-space trade-off for undirected $s-t$-connectivity*, SODA 2013, SIAM, 2013, pp. 1873–1883.
152. M. Koucký, *Universal traversal sequences with backtracking*, J. Comput. System Sci. **65** (2002), no. 4, 717–726.
153. L. Kozma and S. Moran, *Shattering, graph orientations, and connectivity*, Electron. J. Combin. **20** (2013), no. 3, P44.
154. D. Kuske and M. Lohrey, *Logical aspects of Cayley-graphs: the group case*, Ann. Pure Appl. Logic **131** (2005), no. 1-3, 263–286.
155. D. Kuzmin and M. K. Warmuth, *Unlabeled compression schemes for maximum classes*, J. Mach. Learn. Res. **8** (2007), 2047–2081.
156. U. Lang, *Injective hulls of certain discrete metric spaces and groups*, J. Topol. Anal. **5** (2013), no. 3, 297–331.
157. D. B. Lange and M. Oshima, *Seven good reasons for mobile agents*, Commun. ACM **42** (1999), no. 3, 88–89.
158. F. Larrión, M. A. Pizaña, and R. Villarroel-Flores, *The fundamental group of clique-Helly graphs*, Mat. Contem. **39** (2010), 5–7.
159. J. F. Lawrence, *Lopsided sets and orthant-intersection of convex sets*, Pacific J. Math. **104** (1983), no. 1, 155–173.
160. F. T. Leighton, *Finite common coverings of graphs*, J. Combin. Theory Ser. B **33** (1982), no. 3, 231–238.
161. J. Lin, A. S. Morse, and B. D. O. Anderson, *The multi-agent rendezvous problem. part 1: The synchronous case*, SIAM J. Control Optim. **46** (2007), no. 6, 2096–2119.
162. ———, *The multi-agent rendezvous problem. part 2: The asynchronous case*, SIAM J. Control Optim. **46** (2007), no. 6, 2120–2147.
163. N. Littlestone and M. K. Warmuth, *Relating data compression and learnability*, Tech. report, Department of Computer and Information Sciences, Santa Cruz, CA, 1986.
164. V. Lukkarila, *The 4-way deterministic tiling problem is undecidable*, Theoret. Comput. Sci. **410** (2009), no. 16, 1516–1533.
165. R. C. Lyndon and P. E. Schupp, *Combinatorial group theory*, Classics Math., Springer-Verlag, Berlin, 2001, Reprint of the 1977 edition.
166. P. Madhusudan, *Model-checking trace event structures*, LICS 2003, IEEE Computer Society, 2003, pp. 371–380.
167. H. A. Masur and Y. N. Minsky, *Geometry of the complex of curves I: Hyperbolicity*, Invent. Math. **138** (1999), no. 1, 103–149.
168. J. Matoušek, *The number of unique-sink orientations of the hypercube*, Combinatorica **26** (2006), no. 1, 91–99.
169. S. B. Maurer, *Matroid basis graphs I*, J. Combin. Theory Ser. B **14** (1973), no. 3, 216–240.
170. A. W. Mazurkiewicz, *Distributed enumeration*, Inform. Process. Lett. **61** (1997), no. 5, 233–239.
171. T. Mészáros and L. Rónyai, *Shattering-extremal set systems of VC dimension at most 2*, Electron. J. Combin. **21** (2014), no. 4, P4.30.
172. E. F. Moore, *Gedanken-experiments on sequential machines*, Automata studies, Ann. of Math. Stud., no. 34, Princeton University Press, Princeton, N. J., 1956, pp. 129–153.
173. S. Moran, *Shattering-extremal systems*, arXiv preprint **1211.2980** (2012).
174. S. Moran and M. K. Warmuth, *Labeled compression schemes for extremal classes*, ALT 2016, Lecture Notes in Comput. Sci., vol. 9925, Springer, 2016, pp. 34–49.
175. S. Moran and A. Yehudayoff, *Sample compression schemes for VC classes*, J. ACM **63** (2016), no. 3, 21:1–21:10.
176. R. Morin, *Concurrent automata vs. asynchronous systems*, MFCS 2005, Lecture Notes in Comput. Sci., vol. 3618, Springer, 2005, pp. 686–698.
177. H. M. Mulder, *The interval function of a graph*, Mathematical Centre tracts, vol. 132, Mathematisch Centrum, Amsterdam, 1980.
178. D. E. Müller and P. E. Schupp, *The theory of ends, pushdown automata, and second order logic*, Theoret. Comput. Sci. **37** (1985), 51–75.
179. S. B. Myers, *Riemannian manifolds with positive mean curvature*, Duke Math. J. **8** (1941), no. 2, 401–404.
180. O. Narayan and I. Saniee, *Large-scale curvature of networks*, Phys. Rev. E **84** (2011), 066108.

181. A. Naudin, *Impact des connaissances initiales sur la calculabilité distribuée*, Ph.D. thesis, Aix-Marseille Université, 2017.
182. G. A. Niblo and L. D. Reeves, *The geometry of cube complexes and the complexity of their fundamental groups*, Topology **37** (1998), no. 3, 621–633.
183. M. Nielsen, G. D. Plotkin, and G. Winskel, *Petri nets, event structures and domains, part I*, Theoret. Comput. Sci. **13** (1981), no. 1, 85–108.
184. M. Nielsen, G. Rozenberg, and P. S. Thiagarajan, *Transition systems, event structures and unfoldings*, Inform. and Comput. **118** (1995), no. 2, 191–207.
185. M. Nielsen and P. S. Thiagarajan, *Regular event structures and finite Petri nets: the conflict-free case*, ICATPN 2002, Lecture Notes in Comput. Sci., vol. 2360, Springer, 2002, pp. 335–351.
186. N. Norris, *Universal covers of graphs: isomorphism to depth $n-1$ implies isomorphism to all depths*, Discrete Appl. Math. **56** (1995), no. 1, 61–74.
187. P. S. Novikov, *On the algorithmic unsolvability of the word problem in group theory*, Tr. Mat. Inst. Steklova **44** (1955), 3–143 (Russian).
188. R. Nowakowski and P. M. Winkler, *Vertex-to-vertex pursuit in a graph*, Discrete Math. **43** (1983), no. 2-3, 235–239.
189. A. Yu. Olshanskii, *Hyperbolicity of groups with subquadratic isoperimetric inequality*, Internat. J. Algebra Comput. **1** (1991), no. 3, 281–289.
190. D. Osajda, *Connectedness at infinity of systolic complexes and groups*, Groups Geom. Dyn. **1** (2007), no. 2, 183–203.
191. ———, *A combinatorial non-positive curvature I: Weak systolicity*, arXiv preprint **1305.4661** (2013).
192. ———, *Combinatorial negative curvature and triangulations of three-manifolds*, Indiana Univ. Math. J. **64** (2015), no. 3, 943–956.
193. D. Osajda and P. Przytycki, *Boundaries of systolic groups*, Geom. Topol. **13** (2009), no. 5, 2807–2880.
194. J. Oxley, *Matroid theory*, 2nd ed., Oxf. Grad. Texts Math., vol. 21, Oxford Univ. Press, Oxford, 2011.
195. A. Pajor, *Sous-espaces ℓ_1^n des espaces de Banach*, Travaux en Cours, Hermann, Paris, 1985.
196. Dömötör Pálvölgyi and Gábor Tardos, *Unlabeled compression schemes exceeding the VC-dimension*, Discrete Appl. Math. (2019), to appear.
197. P. Panaite and A. Pelc, *Exploring unknown undirected graphs*, J. Algorithm **33** (1999), no. 2, 281–295.
198. A. Papadopoulos, *Metric spaces, convexity and nonpositive curvature*, 2nd ed., IRMA Lect. Math. Theor. Phys., vol. 6, Eur. Math. Soc., 2014.
199. P. Papasoglu, *Strongly geodesically automatic groups are hyperbolic*, Invent. Math. **121** (1995), no. 2, 323–334.
200. ———, *An algorithm detecting hyperbolicity*, Geometric and Computational Perspectives on Infinite Groups (Minneapolis, MN and New Brunswick, NJ, 1994), DIMACS Ser. Discrete Math. Theoret. Comput. Sci., vol. 25, DIMACS, Amer. Math. Soc., 1996, pp. 193–200.
201. A. Pawlik, J. Kozik, T. Krawczyk, M. Lasoń, P. Micek, W. T. Trotter, and B. Walczak, *Triangle-free intersection graphs of line segments with large chromatic number*, J. Combin. Theory Ser. B **105** (2014), 6–10.
202. D. Peleg, *Distributed computing: A locality-sensitive approach*, SIAM, Philadelphia, 2000.
203. N. Polat, *On infinite bridged graphs and strongly dismantlable graphs*, Discrete Math. **211** (2000), no. 1-3, 153–166.
204. ———, *Convexity and fixed-point properties in Helly graphs*, Discrete Math. **229** (2001), no. 1-3, 197–211, Combinatorics, graph theory, algorithms and applications.
205. E. Prisner, *Convergence of iterated clique graphs*, Discrete Math. **103** (1992), no. 2, 199–207.
206. P. Przytycki, *The fixed point theorem for simplicial nonpositive curvature*, Math. Proc. Cambridge Philos. Soc. **144** (2008), no. 3, 683–695.
207. ———, *$\underline{E}G$ for systolic groups*, Comment. Math. Helv. **84** (2009), no. 1, 159–169.
208. A. Quilliot, *Problèmes de jeux, de point fixe, de connectivité et de représentation sur des graphes, des ensembles ordonnés et des hypergraphes*, Thèse de doctorat d'état, Université de Paris VI, 1983.
209. K. Reidemeister, *Einführung in die kombinatorische topologie*, F. Vieweg & Sohn, Braunschweig, 1932.
210. O. Reingold, *Undirected connectivity in log-space*, J. ACM **55** (2008), no. 4, 17:1–17:24.
211. N. Robertson and P. D. Seymour, *Graph minors II: Algorithmic aspects of tree-width*, J. Algorithms **7** (1986), no. 3, 309–322.
212. ———, *Graph minors V: Excluding a planar graph*, J. Combin. Theory Ser. B **41** (1986), no. 1, 92–114.

213. R. M. Robinson, *Undecidability and nonperiodicity for tilings of the plane*, Invent. Math. **12** (1971), no. 3, 177–209.
214. M. Roller, *Poc sets, median algebras and group actions*, Tech. report, Univ. of Southampton, 1998.
215. B. Rozoy and P. S. Thiagarajan, *Event structures and trace monoids*, Theoret. Comput. Sci. **91** (1991), no. 2, 285–313.
216. B. I. P. Rubinstein and J. H. Rubinstein, *A geometric approach to sample compression*, J. Mach. Learn. Res. **13** (2012), 1221–1261.
217. B. I. P. Rubinstein, J. H. Rubinstein, and P. L. Bartlett, *Bounding embeddings of VC classes into maximum classes*, Measures of Complexity. Festschrift for Alexey Chervonenkis (V. Vovk, H. Papadopoulos, and A. Gammerman, eds.), Springer, Cham, 2015, pp. 303–325.
218. M. Sageev, *Ends of group pairs and non-positively curved cube complexes*, Proc. London Math. Soc. **s3-71** (1995), no. 3, 585–617.
219. ———, *CAT(0) cube complexes and groups*, Geometric Group Theory (M. Bestvina, M. Sageev, and K. Vogtmann, eds.), IAS/Park City Math. Ser., vol. 21, Amer. Math. Soc., Inst. Adv. Study, 2012, pp. 6–53.
220. R. Samei, B. Yang, and S. Zilles, *Generalizing labeled and unlabeled sample compression to multi-label concept classes*, ALT 2014, Lecture Notes in Comput. Sci., vol. 8776, Springer, 2014, pp. 275–290.
221. L. Santocanale, *A nice labelling for tree-like event structures of degree 3*, Inform. and Comput. **208** (2010), no. 6, 652–665.
222. N. Sauer, *On the density of families of sets*, J. Combin. Theory Ser. A **13** (1972), no. 1, 145–147.
223. V. Schmitt, *Stable trace automata vs. full trace automata*, Theoret. Comput. Sci. **200** (1998), no. 1-2, 45–100.
224. A. D. Scott, *Induced trees in graphs of large chromatic number*, J. Graph Theory **24** (1997), no. 4, 297–311.
225. D. Seese, *The structure of models of decidable monadic theories of graphs*, Ann. Pure Appl. Logic **53** (1991), no. 2, 169–195.
226. S. Shalev-Shwartz and S. Ben-David, *Understanding machine learning: From theory to algorithms*, Cambridge Univ. Press, 2014.
227. Y. Shavitt and T. Tankel, *Hyperbolic embedding of Internet graph for distance estimation and overlay construction*, IEEE/ACM Trans. Netw. **16** (2008), no. 1, 25–36.
228. S. Shelah, *A combinatorial problem, stability and order for models and theories in infinitary languages*, Pacific J. Math. **41** (1972), no. 1, 247–261.
229. E. E. Shult, *Points and lines: Characterizing the classical geometries*, Universitext, Springer, Heidelberg, 2011.
230. V. P. Soltan, *Introduction to the axiomatic theory of convexity*, Stiinţa, Chişinău, 1984, (in Russian).
231. V. P. Soltan and V. Chepoi, *Conditions for invariance of set diameters under d-convexification in a graph*, Kibernetika (Kiev) **6** (1983), 14–18 (Russian, with English summary).
232. M. Soto, *Quelques propriétés topologiques des graphes et applications à Internet et aux réseaux*, Ph.D. thesis, Université Paris Diderot, 2011.
233. E. W. Stark, *Connections between a concrete and an abstract model of concurrent systems*, MFPS 1989, Lecture Notes in Comput. Sci., vol. 442, Springer, 1989, pp. 53–79.
234. S. Suri, E. Vicari, and P. Widmayer, *Simple robots with minimal sensing: From local visibility to global geometry*, Int. J. of Robot. Res. **27** (2008), no. 9, 1055–1067.
235. I. Suzuki and M. Yamashita, *Distributed anonymous mobile robots: Formation of geometric patterns*, SIAM J. Comput. **28** (1999), no. 4, 1347–1363.
236. T. Szabó and E. Welzl, *Unique sink orientations of cubes*, FOCS 2001, IEEE Computer Society, 2001, pp. 547–555.
237. M. Tancer, *Recognition of collapsible complexes is NP-complete*, Discrete Comput. Geom. **55** (2016), no. 1, 21–38.
238. S. Tani, *Compression of view on anonymous networks - folded view -*, IEEE Trans. Parallel Distrib. Syst. **23** (2012), no. 2, 255–262.
239. P. S. Thiagarajan, *Regular trace event structures*, Technical Report BRICS RS-96-32, Computer Science Department, Aarhus University, Aarhus, Denmark, 1996.
240. ———, *Regular event structures and finite Petri nets: A conjecture*, Formal and Natural Computing, Lecture Notes in Comput. Sci., vol. 2300, Springer, 2002, pp. 244–256.
241. P. S. Thiagarajan and S. Yang, *Rabin's theorem in the concurrency setting: a conjecture*, Theoret. Comput. Sci. **546** (2014), 225–236.
242. J. Tits, *Buildings of spherical type and finite BN-pairs*, Lecture Notes in Math., vol. 386, Springer-Verlag, Berlin, 1974.
243. J. Ueberberg, *Foundations of incidence geometry*, Springer Monogr. Math., Springer, Heidelberg, 2011, Projective and polar spaces.

244. M. van de Vel, *Matching binary convexities*, Topology Appl. **16** (1983), no. 3, 207–235.
245. V. N. Vapnik and A. Y. Chervonenkis, *On the uniform convergence of relative frequencies of events to their probabilities*, Theory Probab. Appl. **16** (1971), no. 2, 264–280.
246. K. Verbeek and S. Suri, *Metric embedding, hyperbolic space, and social networks*, SoCG 2014, ACM, 2014, pp. 501–510.
247. M. K. Warmuth, *Compressing to VC dimension many points*, COLT/Kernel 2003, Lecture Notes in Comput. Sci., vol. 2777, Springer, 2003, pp. 743–744.
248. G. M. Weetman, *A construction of locally homogeneous graphs*, J. London Math. Soc. (2) **50** (1994), no. 1, 68–86.
249. E. Welzl, *Complete range spaces*, (1987), Unpublished notes.
250. D. H. Wiedemann, *Hamming geometry*, Ph.D. thesis, University of Waterloo, 1986, re-typeset July, 2006.
251. A. Wigderson, *Mathematics and computation*, Princeton Univ. Press, 2019.
252. P. M. Winkler, *Isometric embedding in products of complete graphs*, Discrete Appl. Math. **7** (1984), no. 2, 221–225.
253. G. Winskel, *Events in computation*, Ph.D. thesis, Edinburgh University, 1980.
254. G. Winskel and M. Nielsen, *Models for concurrency*, Handbook of Logic in Computer Science (S. Abramsky, Dov M. Gabbay, and T. S. E. Maibaum, eds.), vol. 4, Oxford Univ. Press, 1995, pp. 1–148.
255. D. T. Wise, *Non-positively curved squared complexes, aperiodic tilings, and non-residually finite groups*, Ph.D. thesis, Princeton University, 1996.
256. ———, *Sixtolic complexes and their fundamental groups*, (2003), unpublished manuscript.
257. ———, *Complete square complexes*, Comment. Math. Helv. **82** (2007), no. 4, 683–724.
258. ———, *From riches to raags: 3-manifolds, right-angled Artin groups, and cubical geometry*, CBMS Reg. Conf. Ser. Math., vol. 117, Amer. Math. Soc., Providence, RI, 2012.
259. M. Yamashita and T. Kameda, *Computing on anonymous networks: Part I - Characterizing the solvable cases*, IEEE Trans. Parallel Distrib. Syst. **7** (1996), no. 1, 69–89.
260. ———, *Leader election problem on networks in which processor identity numbers are not distinct*, IEEE Trans. Parallel Distrib. Syst. **10** (1999), no. 9, 878–887.
261. G. M. Ziegler, *Lectures on polytopes*, Grad. Texts in Math., vol. 152, Springer–Verlag, New York, 1995.