Pairwise Compatibility Graphs: A Survey

Tiziana Calamoneri, Blerina Sinaimeri

PAIRWISE COMPATIBILITY GRAPHS: A SURVEY

TIZIANA CALAMONERI [*] AND BLERINA SINAIMERI[†]

Abstract. A graph $G = (V, E)$ is a *pairwise compatibility graph (PCG)* if there exists an edge-weighted tree T and two non-negative real numbers d_{min} and d_{max} such that each leaf u of T is a node of V and there is an edge $(u, v) \in E$ if and only if $d_{min} \leq d_T(u, v) \leq d_{max}$ where $d_T(u, v)$ is the sum of weights of the edges on the unique path from u to v in T.

In this paper, we survey the state of the art concerning this class of graphs and some of its subclasses.

Key words. pairwise compatibility graphs, leaf power graphs, min leaf power graphs.

AMS subject classifications. 05C90, 68R01, 68R10

1. Introduction. The reconstruction of ancestral relationships is one of the fundamental problems in computational biology as widely used to provide both evolutionary and functional insights into biological systems. The evolutionary history of a set of organisms is usually represented by a tree-like structure called phylogenetic tree which is a tree where each leaf represents a distinct known taxon and the internal nodes possible ancestors that might have led through evolution to this set of taxa. Moreover, the edges of the tree can be weighted in order to represent a sort of evolutionary distance among species. In the phylogenetic tree reconstruction problem, given a set of taxa, we want to find a phylogenetic tree that "best" explains the given data. Due to the difficulty in determining the criteria for an "optimal" tree, the performance of the reconstruction algorithms is usually evaluated experimentally by comparing the tree produced by the algorithm with the "known" tree. However, as the tree reconstruction problem is proved to be NP-hard under many criteria of optimality and as real phylogenetic trees consist usually of a large number of nodes, testing these heuristics on real data is difficult. Thus, it is interesting to find efficient ways to sample subsets of taxa from a large phylogenetic tree, subject to some biologically-motivated constraints, in order to test the reconstruction algorithms on the smaller subtrees induced by the sample. The constraints on the sample attempt to assure that the behaviour of the algorithm will not be biased by the fact it is applied on the sample instead on the whole tree. For instance, as observed in [32], very close or very distant taxa can create problems for phylogeny reconstruction algorithms. This leads to the following constraint on the sample: given two positive integers d_{min}, d_{max}, select a sample from the leaves of the tree such that the pairwise distance between any two leaves in the sample is at least d_{min} and at most d_{max}. This sampling problem was considered in [39] and polynomial algorithms were proposed. This motivates the introduction of *pairwise compatibility graphs* (PCG). Indeed, given a phylogenetic tree T, and integers d_{min}, d_{max} we can associate a graph G, called the pairwise compatibility graph of T, whose nodes are the leaves of T and for which there is an edge between two nodes if the corresponding leaves in T are at a distance within the interval $[d_{min}, d_{max}]$. While it is trivial to construct the graph G starting from T, d_{min}, d_{max}, the inverse problem is difficult.

PCGs can be seen as a generalization of the well-studied class of *leaf power graphs* (in which $d_{min} = 0$), introduced in the problem of constructing phylogenies from species similarity data [24, 41, 47]. Specifically, interspecies similarity is represented by a graph G where the nodes are the species and the adjacency relation represents evidence of evolutionary

[*]Computer Science Department, Sapienza University of Rome, Via Salaria 113, 00198 Roma, Italy.(calamoneri@di.uniroma1.it)

[†]INRIA and Université de Lyon, Université Lyon 1, LBBE, CNRS UMR558, France. (blerina.sinaimeri@inria.fr). Questions, comments, or corrections to this document may be directed to that email address.

similarity. The phylogenetic tree is then built from this graph such that the leaves correspond to nodes of the graph and two leaves that correspond to adjacent nodes are at distance at most d_{max} in the tree, where d_{max} is a chosen threshold of proximity. Although there has been a lot of work on this topic (see e.g. [47, 6, 10, 5]), a complete description of leaf power graphs is still unknown and remains an important research problem.

Another natural relaxation of the pairwise compatibility constraint is the one obtained when d_{max} is set to ∞. Thus, there is an edge (u, v) in G if and only if $d_T(u, v) \geq d_{min}$. This relaxation leads to the definition of the class of *min leaf power graphs (mLPGs)* [20]. It is worth to mention that initially the mLPG class has been defined as the complement of the LPG class, in an attempt to better understand the structure of PCGs. This is the reason why their name evokes the power of graphs although these graphs are not of this type.

In this survey we review the results on the identification of the classes PCG and mLPG, and some of the results for LPG. For more details on the characterization of the LPG class we refer the reader to the nice survey of Brandstädt [3].

This paper is organized as follows. Section 2 is devoted to some basic definitions and some preliminary results. We have included most of the definitions of the graph classes we mention. However, for further details the interested reader can consult [9]. In Section 3 we survey the main results related to the complexity of recognizing pairwise compatibility graphs. Section 4 is devoted to the graph class $LPG \cap mLPG$. In Section 5 we list the graphs classes which are known not to belong to the class PCG. Section 6 includes the state of the art on the characterization of the PCG class and its subclasses mLPG and LPG. In particular it presents the known results on the graph classes contained in PCG as well as the results concerning the characterization of the PCGs in terms of forbidden configurations. In Section 7 we consider PCGs of particular subclasses of trees, such as stars and caterpillars. In each section we also include a number of main open problems. Finally, we conclude in Section 8 with some possible research directions related to pairwise compatibility graphs.

2. Basic definitions. In this section we recall some basic definitions that we use throughout this paper.

A graph $G = (V, E)$ is a *pairwise compatibility graph* (PCG) if there exists a tree T, a positive edge weight function w on T and two non-negative real numbers d_{min} and d_{max}, $d_{min} \leq d_{max}$, such that each node $u \in V$ is uniquely associated to a leaf l_u of T and there is an edge $(u, v) \in E$ if and only if $d_{min} \leq d_{T,w}(l_u, l_v) \leq d_{max}$ where $d_{T,w}(l_u, l_v)$ is the sum of the weights of the edges on the unique path from l_u to l_v in T. In such a case, we say that G is a PCG of T for d_{min} and d_{max}; in symbols, $G = PCG(T, w, d_{min}, d_{max})$.

A graph $G(V, E)$ is called a *leaf power graph* (LPG) if there exists a tree T, a positive edge weight function w on T and a non-negative number d_{max} such that there is an edge (u, v) in E if and only if for their corresponding leaves in T, l_u, l_v, we have $d_{T,w}(l_u, l_v) \leq d_{max}$; in symbols $G = LPG(T, w, d_{max})$.

A graph $G = (V, E)$ is a *minimum leaf power graph* (mLPG) if there exists a tree T, a positive edge weight function w on T and an integer d_{min} such that there is an edge (u, v) in E if and only if for their corresponding leaves in T l_u, l_v we have $d_{T,w}(l_u, l_v) \geq d_{min}$; in symbols, $G = mLPG(T, w, d_{min})$.

We refer by *k-leaf power graph* (*k-min leaf power graph*, respectively) to a graph which is leaf power graph with $d_{max} = k$ ($d_{min} = k$, respectively). In Figure 1 examples of a PCG, an LPG and an mLPG are depicted.

A graph $G = (V, E)$ is an *exact k-leaf power* [8] if there is a weighted tree T such that each node $u \in V$ is uniquely associated to a leaf l_u of T and there is an edge $(u, v) \in E$ if and only if $d_T(l_u, l_v) = k$. It is clear that an exact k-leaf power graph is a PCG where $d_{min} = d_{max} = k$.
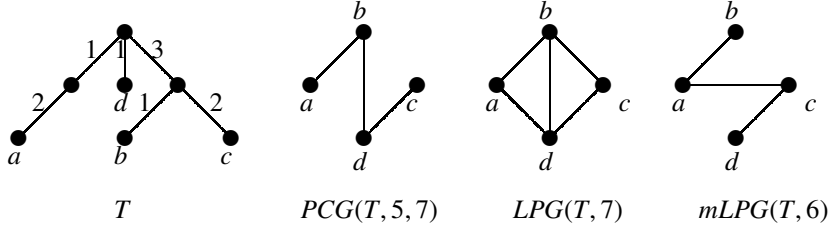
Fig. 1: An edge-weighted tree T and an example of a PCG, an LPG and an mLPG.

Observe that here we always assume that d_{min}, d_{max} and the weight of the edges of the tree of a PCG are all positive real numbers. In the original problem concerning the LPGs, these quantities were required to be natural numbers. It is proved in [5] that it is not a loss of generality to consider positive real numbers instead of naturals for LPGs. This result is extended to the general case of PCGs as follows:

LEMMA 2.1. *[16] Let $G = PCG(T, w, d_{min}, d_{max})$, where d_{min}, d_{max} are non-negative real numbers and the weight $w(e)$ of each edge e of T is a positive real number. Then it is possible to choose natural numbers $\hat{w}, \hat{d}_{min}, \hat{d}_{max}$ such that $G = PCG(T, \hat{w}, \hat{d}_{min}, \hat{d}_{max})$.*

A graph $G = (K, S, E)$ is said to be a *split* graph [33] if there is a node partition $V = K \cup S$ such that the subgraphs induced by K and S are complete and stable, respectively.

A graph $G = (V, E)$ is a *thin spider* [38] if V can be partitioned into three sets K, S and R such that:

i. K is complete, S is stable, and $|K| = |S| \geq 2$;
ii. each node in R is adjacent to each node of K and to no node in S;
iii. each node in S has a unique neighbor in K; more formally there exists a bijection $f : K \to S$ such that every node $k \in K$ is adjacent to $f(k) \in S$ and to no other node in S.

The complement of a thin spider is a *thick spider*.

The special case of these graphs in which $R = \emptyset$ is considered in [20] and they are called *n-split matching* and *n-split antimatching* graphs, respectively. Examples are shown in Figure 2. Note that the 3-split matching is sometimes called a *net* and denoted by $\bar{S_3}$ while the 3-split anti-matching is denoted by S_3 [26].

We will denote by $\mathcal{SM}$ and $\mathcal{SA}$ the class of split matching and split antimatching graphs, respectively.
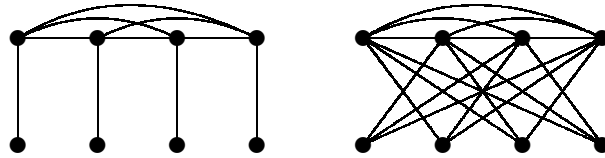

Fig. 2: A 4-split matching and a 4-split antimatching.

The *Dilworth number* [27] of a graph is the size of the largest subset of its nodes in which the closed neighborhood of no node contains the neighborhood of another.

The class of threshold graphs has been introduced many times in several contexts, with

different names and various equivalent definitions (see for example [42]). For the purposes of this paper, it is enough to say that the class of *threshold graphs* $\mathcal{T}$, is characterized as all the graphs with Dilworth number one [25]. Note that threshold graphs are split graphs. An example of a threshold graph is depicted in Figure 3(a).

A graph $G = (V, E)$ is a *threshold tolerance graph* [44] if every node v_i can be assigned a real weight a_i and a real tolerance t_i such that for every $(v_i, v_j) \in E \Leftrightarrow |a_i + a_j| \geq \min\{t_i, t_j\}$. Figure 3(b) shows a threshold tolerance graph. Threshold tolerance graphs have been introduced in [44] as a generalization of threshold graphs. Indeed, threshold graphs constitute a proper subclass obtained by considering a constant tolerance function [45].

Fig. 3: (a) A threshold graph, (b) a threshold tolerance graph.

A graph is *planar* if it can be embedded in the plane so that no two edges intersect geometrically except at the node to which both of them are incident. A graph is *outerplanar* if it has a planar embedding where all nodes are on the outer face.

A *ladder* consists of two distinct paths of the same length $u_1, \ldots, u_{n/2}$ and $v_1, \ldots, v_{n/2}$ plus the edges (u_i, v_i), $i = 1, \ldots, n/2$.

A *chord* of a cycle C is an edge not in the edge set of C whose endpoints lie in the node set of C. We say that an edge is a chord of a graph if it is a chord of some cycle in the graph.

A graph is *chordal* if every cycle of length at least 4 has a chord.

A *sun* graph [31] is a graph G on $2n$ nodes for some $n \geq 3$ whose node set can be partitioned into two sets, $W = \{w_1, \ldots, w_n\}$ and $U = \{u_1, \ldots, u_n\}$, such that U induces a clique, W is an independent set and for each i and j, w_j is adjacent to u_i if and only if $i = j$ or $i \equiv j + 1 (\mod n)$. In Figure 4(a) a sun with $n = 4$ is depicted.

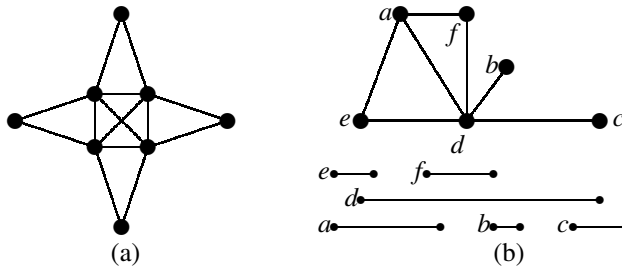*Strongly chordal graphs* [31] are sun-free chordal graphs.

Fig. 4: (a) A sun graph; (b) an interval graph.

A *t-caterpillar* [36] is a tree in which all the nodes are within distance ≤ 1 of a central path, called *spine*, constituted of t nodes.

A graph G is an *intersection graph* if its nodes correspond to a family of sets $\{S_v : v \in V(G)\}$ and (u, v) is an edge of G if and only if $S_u \cap S_v \neq \emptyset$.

We consider now some graph classes that can be defined through a intersection graphs of special objects.

An *interval graph* is the intersection graph of a set of intervals on a line (an example is shown in Figure 4(b)).

A *disk* graph is the intersection graph of disks in the plane. A graph is *grid intersection* if it is the intersection graph of horizontal and vertical line segments in the plane. A *circular arc* graph is the intersection graph of arcs of a circle.

A graph is *rectangle (square) intersection* if it has an intersection model consisting of axis-parallel rectangular (squared) boxes in the plane.

A *trapezoid* graph is the intersection graph of trapezoids between two parallel lines. A *permutation* graph is the intersection graph of straight lines between two parallels.

The following chain of inclusions holds:

interval graphs $\subseteq$ circular arc graphs $\subseteq$ permutation graphs $\subseteq$ trapezoid graphs.

A graph is a *tolerance graph* [35] if to every node v can be assigned a closed interval I_v on the real line and a tolerance t_v such that x and y are adjacent if and only if $|I_x \cap I_y| \geq \min\{t_x, t_y\}$, where $|I|$ is the length of the interval I. Tolerance graphs can be described through another intersection model, as they are equivalent to *parallelepiped* graphs, defined as the intersection graphs of *special parallelepiped* on two parallel lines.

3. Complexity of recognizing pairwise compatibility graphs. The problem of recognizing whether a graph is PCG is formally defined as follows:

Problem 1 (The PCG Recognition Problem).
INSTANCE : A graph $G = (V, E)$.
DECIDE : Is there a tree T, an edge-weight function w and two integers d_{min}, d_{max} such that $G = PCG(T, w, d_{min}, d_{max})$?

The complexity of this problem is still unknown. However, in [30] the NP-completeness is proved for the following generalization of the previous problem.

Problem 2 (The Max-Generalized-PCG Recognition Problem).
INSTANCE : A graph G, a subset S of the edges of its complement graph, and a positive integer k.
DECIDE : Is there a $G' = PCG(T, w, d_{min}, d_{max})$ such that G' contains G as a (not necessary induced) subgraph but does not contain any edge of S; and at least k edges of S have distance greater than d_{max} between their corresponding leaves in T?

Observe that when $S = \bar{E}$ then the problem becomes exactly determining whether G is a PCG.

It is worth to note that the problem of sampling a set of m leaves for a weighted tree T, such that their pairwise distance is within some interval $[d_{min}, d_{max}]$, reduces to selecting a clique of size m uniformly at random from the graph $PCG(T, w, d_{min}, d_{max})$. As the sampling problem can be solved in polynomial time on PCGs [39], it follows that the max clique problem is solved in polynomial time on this class of graphs, providing that the tree T, the weight function w and the two values d_{min}, d_{max} are known or can be found in polynomial time.

Open problem: Determine the computational complexity of the PCG Recognition problem. Durocher et al. [30] conjecture that the PCG recognition problem is NP-hard.

For what concerns LPGs, given an integer k we can formulate the following:

PROBLEM 3 (The k-LPG Recognition Problem).
INSTANCE : A graph $G = (V, E)$.
DECIDE : Is there a tree T such that $G = LPG(T, k)$?

In [29] it is shown that this problem can be solved in polynomial time if the $(k − 2)$-Steiner root problem can be solved in polynomial time. Chang and Ko [23] give a linear time algorithm for the 3-Steiner root problem, so implying that the k-leaf power recognition problem can be solved in linear time for $k = 5$.

Open problem: Determine the computational complexity of the k-LPG Recognition Problem for $k \geq 6$.

4. LPG $\cap$ mLPG. Here we give some relations between mLPG and LPG, the two main subclasses of PCG. In particular the next result holds.

PROPOSITION 4.1. *[16] The class co-LPG coincides with mLPG and, consequently, the class co-mLPG coincides with LPG.*

The relation between the classes LPG and mLPG is graphically shown in Figure 5 and it is deduced from the following considerations:

- the union of LPG and mLPG does not coincide with the whole class PCG. Indeed, the class C of cycles is in PCG but does not belong either to LPG or to mLPG [16, 55];
- the class $\mathcal{T}$ of threshold graphs belongs to $LPG \cap mLPG$ [20];
- the class $\mathcal{SM}$ of split matchings belongs to $mLPG \setminus LPG$ while the class $\mathcal{SA}$ of split antimatchings belongs to $LPG \setminus mLPG$ [20].
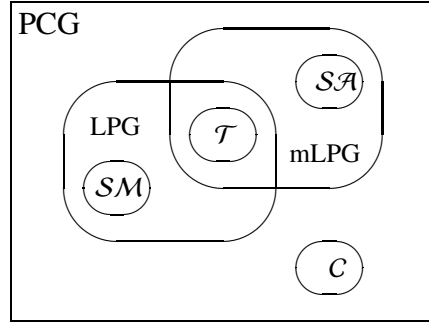

Fig. 5: Relationships between PCG, LPG and mLPG.

The previous arguments lead to the following summarizing result:

THEOREM 4.2. *For the classes of LPG and mLPG, the following relations hold: (a) $mLPG \cup LPG \subset PCG$, (b) $mLPG \cap LPG \neq \emptyset$, (c) $LPG \setminus mLPG \neq \emptyset$, (d) $mLPG \setminus LPG \neq \emptyset$.*

Particular attention in the literature has been given to the characterization of the intersection of the mLPG and LPG classes. Due to Proposition 4.1, it is clear that a self-complemented class that is included either in LPG or in mLPG is included also in $LPG \cap mLPG$. For example split permutation graphs that are the intersection class between interval and co-interval graphs, are in $LPG \cap mLPG$ as interval graphs are in LPG (see Section 6.1.). Nevertheless a complete characterization of the set of graphs in this intersection is still missing. Some graph classes that are known to be included inside $LPG \cap mLPG$ are the followings: disjoint union of cliques, that are the sole 2-leaf power graphs [3] and threshold graphs (which

have Dilworth number 1) [20]. More in general for graphs with arbitrary Dilworth number, the following result holds.

THEOREM 4.3. *[19, 18, 17]*

- *All Dilworth 1 graphs (i.e. threshold graphs) are in PCG ∩ mLPG and the witness trees are stars.*
- *All Dilworth 2 graphs are in PCG ∪ mLPG and the witness trees are 2-caterpillars. A proper subclass of Dilworth 2 graphs (properly containing threshold graphs) is in PCG ∩ mLPG.*
- *Given a t-caterpillar Γ, for any edge-weight w and any value c, the graphs LPG(Γ, w, c) and mLPG(Γ, w, c) are Dilworth t graphs; nevertheless, there are Dilworth t graphs, t ≥ 2 that do not belong to LPG ∩ mLPG. As an example, the graph depicted in Figure 6 has Dilworth number 3 and is neither a LPG nor a mLPG.*
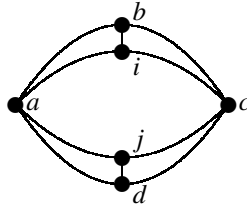

Fig. 6: A graph on 6 vertices that has Dilworth number 3 and is neither a LPG nor a mLPG.

Open problem: Characterize completely the intersection of mLPG and LPG.

5. Graphs that are not PCGs. Initially it was believed that every graph was a PCG. Indeed, first Phillips [48] proved in an exhaustive way that all graphs with less than 5 nodes are PCGs, then the result has been extended to all graphs with at most 7 nodes [14] and finally to all bipartite graphs on 8 nodes [43].

However, not all graphs are PCGs: Yanhaona et al. [54] show a bipartite graph with 15 nodes (depicted in Figure 7(a)) that is not PCG. Subsequently, Mehnaz and Rahman [43] provide a list of bipartite graphs not to be in PCG. More recently, Durochet et al. [30] prove that there exists a (not bipartite) graph with 8 nodes that is not PCG (depicted in Figure 7(b)). In view of the previously listed results, this is the smallest graph that is not PCG. The same authors provide also an example of a planar graph with 20 nodes that is not a PCG (depicted in Figure 8). As a consequence, neither bipartite nor planar graphs are included in PCG.

It is known that graph *H* depicted in Fig. 7(b) is not in PCG [51]. On the other hand, Figures 9(a), 9(b) and 9(c) show a representation of graph *H* as a disk graph, as a circular arc graph, and as square intersection graphs, respectively. This is enough to ensure that all these graph classes are not in PCG [15]. Moreover, rectangle (square) intersection graphs are a superclass of grid intersection graphs, and hence they are not in PCG. Recalling the chain of inclusions stated in the preliminaries we can deduce that trapezoid and permutation graphs are not in PCG.

Finally, in [15] it was shown that tolerance graphs are not in PCG.

Open problem: Find other graph classes that do not belong to the PCG class.

6. Graph classes in PCG. In this section we list the graph classes which are proven to belong to the PCG class. For many of these graph classes it is also known whether they belong to mLPG or LPG. Hence, for facilitating the reading we state the results concerning LPG and mLPG separately.
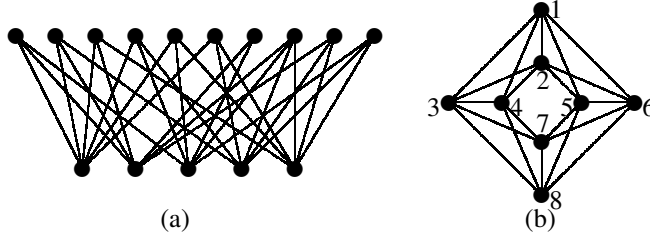
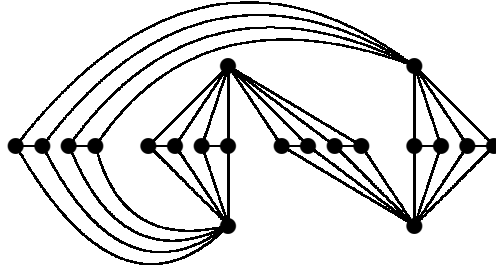**Fig. 7:** (a) The first graph proven not to be a PCG. (b) The graph of smallest size proven not to be a PCG.

Fig. 8: A planar graph with 20 nodes that is not PCG.

6.1. LPGs. Observe that trees are sun-free and chordal, and taking powers and induced subgraphs does not destroy this property [50]. It follows that every LPG is strongly chordal, i.e. (C_{n+4}, sun)–free, $n \geq 0$ [31].

However, not every strongly chordal graph is an LPG: as an example, the graph –found by Bibelnieks and Dearing [2] and shown in Figure 10– is a strongly chordal graph and is not a LPG.

Neighborhood subtree tolerance (NeST) graphs were introduced by Bibelnieks and Dearing [2] and were also studied in [37]. For the sake of brevity, we avoid defining NeST graphs here, and we only mention that Brandstädt et al. [5] show that LPG coincides with the fixed tolerance NeST graph class.

LPG is a superclass of ptolemaic graphs [4, 5], and even a superclass of directed rooted path graphs, introduced by Gavril [34]. Interval graphs are LPGs [4]; it follows that quasi-threshold graphs (that are P_4-free interval graphs) are also LPGs.

Open Problem: Characterize the graphs that are in LPG.

As we have previously mentioned, a graph is leaf power if it is k-leaf power for some integer k. Thus, it is interesting to exploit the structure of these subclasses of LPG. Obviously, a graph G is a 2-leaf power graph if and only if it is the disjoint union of cliques, that is, G does not contain a chordless path of length 2. Dom et al. [29, 28] prove that 3-leaf power graphs are exactly the graphs that do not contain an induced bull, dart, or gem (see Figure 11).

Brandstädt et al. [6] provide another characterization of 3-leaf power graphs by showing that they are exactly the graphs that result from substituting cliques into the nodes of a tree. Moreover, they give a linear time algorithm to recognize 3-leaf power graphs based on their characterization.
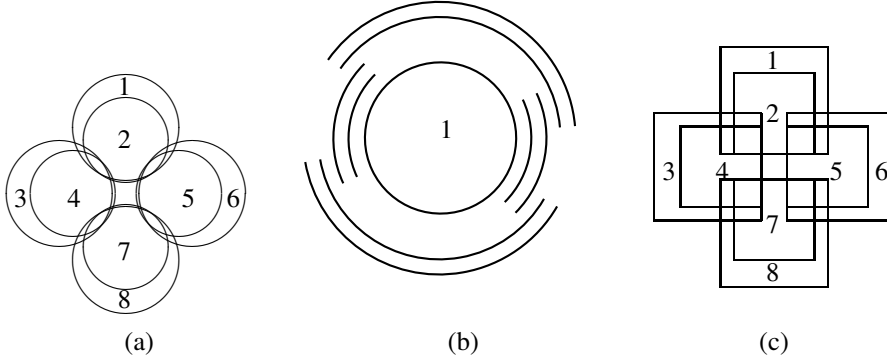
Fig. 9: Representation of *H* as: (a) a disk graph, (b) a circular arc graph, (c) a square intersection graph.
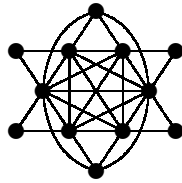

Fig. 10: A strongly chordal graph which is not in LPG.

Basic 4-leaf power graphs, i.e. the 4-leaf power graphs without true twins (two connected nodes with the same neighbourhood), are characterized by eight forbidden subgraphs [49]. It is shown that every 4-leaf power graph results from substituting cliques into the nodes of a basic 4-leaf power graph. Thus, a characterization of basic 4-leaf power graphs automatically leads to a characterization of 4-leaf power graphs in general [10].

Concerning 5-leaf power graphs, a polynomial time recognition algorithm was given in [23]. However, again no structural characterization is known, even for basic 5-leaf power graphs. Only for distance-hereditary basic 5-leaf power graphs a characterization in terms of 34 forbidden induced subgraphs has been discovered [7].

For general *k*, it is proved that *k*-leaf power graphs are not included in $(k + 1)$-leaf power graphs class [11, 12]. Beside these result, there has not been much progress towards the characterisation of these graph classes and the following problem remains open.

Open problem: Determine the structure of *k*-leaf power graphs for $k \geq 5$.

Recently, Nevries and Rosenke [46] have provided a list of 7 graphs that cannot be induced subgraphs of any LPG, and they conjecture that these are sufficient to characterize LPG in terms of forbidden subgraphs. We only remark that one of these graphs is the one already presented in Figure 10, while the other six graphs are strongly chordal graphs of smaller size. Before this work, it was conjectured that the graph of size 12 in Figure 10 was the smallest strongly chordal graph which does not belong to LPG. Nevertheless, the results in [46] imply that the smallest known strongly chordal graph that does not belong to LPG has 10 nodes.

Open problem: It remains an open problem either to prove or to disprove the conjecture stating that the LPG class can be defined as the class of graphs that does not contain any of the 7 subgraphs provided in [46]. It is important to notice that if this conjecture is true, it would imply a polynomial time recognition algorithm for LPGs.
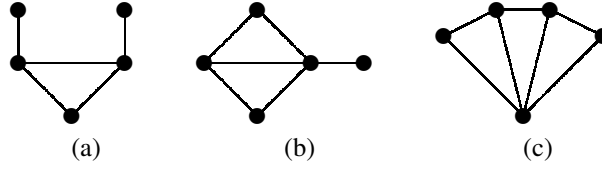
Fig. 11: (a) A bull; (b) a dart; (c) a gem.

6.2. mLPGs. A graph is $2K_2$-free if it does not contain an independent pair of edges as an induced subgraph. Recall that LPGs are chordal and hence C_4-free. Consequently, their complement, mLPG is $2K_2$-free.

Observe that a tree that is $2K_2$-free can not have a path of length greater than 3, and hence it has a diameter at most 3. It follows that every tree of diameter at least 4 does not belong in mLPG.

In [20] it is proved that split matching graphs are not in mLPG. As split matching graphs are $2K_2$-free this means that mLPG does not coincide with the $2K_2$-free graph class.

Finally, it has been proved in [22] that threshold tolerance graphs are strictly included in mLPG.

Open problem: It follows from the results presented in this survey that if G is in $LPG \cap mLPG$ then G is $(2K_2, C_{n+4}$, sun, split matching, split antimatching)–free. It would be interesting to characterize the class $LPG \cap mLPG$ in terms of forbidden subgraphs.

6.3. PCGs. Many graph classes have been proved to be in PCG: cycles, single chord cycles, cacti, tree power graphs, Steiner k-power and phylogenetic k-power graphs [54, 55]. More recently, even trees, ladder graphs, triangle-free outerplanar 3-graphs [52] and Dilworth 2 graphs [19] have been proved to be PCGs. All these graphs admit as a witness tree a caterpillar.

We have already stated that the class of bipartite graphs is not included in PCG. However, in [54] some particular subclasses of bipartite graphs are proved to be in PCG.

A *split matrogenic* graph [42] is a graph that can be constructed as a particular composition of split matchings and split anti-matchings. More formally, given a split graph $F = (V_K \cup V_S, E(F))$ and a simple graph $H = (V(H), E(H))$, their *composition* is a graph $G = (V, E) = F \circ H$ defined as follows:
- $V = V_K \cup V_S \cup V(H)$
- $E = E(F) \cup E(H) \cup \{(a, v) : a \in V_K, v \in V(H)\}$.

A split matrogenic graph is the composition of t split graphs $G_i = (K_i, S_i, E_i)$ with $i = 1, \ldots, t$ such that: either G_i is a split matching or G_i is a split antimatching or $K_i = \emptyset$ (and G_i is called *stable graph*) or $S_i = \emptyset$ (and G_i is called *clique graph*) [53].

In [21] it is proved that if the split matrogenic graph is composed using only split matching graphs or only split anti-matching graphs, then it belongs to PCG.

This result was extended to the following larger subclass of split matrogenic graphs [21].

THEOREM 6.1. *[21] Let $H = G_1 \circ \ldots \circ G_t$ be a split matrogenic graph for which there exists an index $1 \leq h \leq t$ such that $G_1, \ldots G_h$ are all split matching graphs, and $G_{h+1}, \ldots, G_t$ are all split anti-matching graphs. Then, H is in PCG.*

In fact, it seems that the order of appearance of a split matching or an split antimatching in the composition of a split matrogenic graph is somehow strictly related to the pairwise compatibility property.
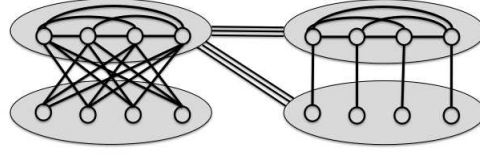
Fig. 12: The smallest split matrogenic graph for which it is still an open problem determining whether it belongs to the PCG class or not. The triple lines between the split antimatching graph and the split matching graph represent the composition operation.

Open problem: It would be interesting to understand whether the split matrogenic graph in Figure 12 with 16 nodes constituted by an 8-node split anti-matching composed with an 8-node matching is in PCG. The solution of this problem would shed some light on the possible inclusion of split matrogenic graphs in PCG.

In [16] the authors study the closure properties of the classes PCG, mLPG and LPG, under some common graph operations as: adding an isolated or universal node, adding a degree one node, adding a twin, taking the complement of a graph and taking the disjoint union of two graphs. Except for its intrinsic interest, this is also important as it is known that many graph classes can be built by means of recursive applications of particular graph operations. Using these results it was proved in [22] that the class of bipartite distance-hereditary graphs are PCGs.

Open problem: In [16] it was also proved that the classes mLPG, LPG and PCG exhibit different closure properties under a given graph operation. In particular the mLPG and LPG are not closed under the complement, however determining whether the PCG class is closed under the complement is still an open problem.

7. PCGs of a particular tree topology. Given a graph, even knowing that it is PCG, it is in general rather difficult to find the witness tree. And in fact, in the literature, most of the trees witnessing that a certain graph class is in PCG (or in LPG, or in mLPG) are very easy structures, such as stars and caterpillars. So, it seems interesting to consider the problem of characterizing subclasses of PCGs derived from a specific topology of the pairwise compatibility tree.

7.1. Stars. Stars are a very simple subclass of trees and hence it is natural to ask what are the graphs that are PCGs of a star. In [20] it is proved that threshold graphs are characterized by being LPG (and mLPG) of stars. In the same paper, this result was extended to show that PCGs of stars are in fact a special superclass of threshold graphs. In particular, the authors define the following superclass of threshold graphs:

The *vicinal preorder* $\preceq$ of a graph $G = (V, E)$ on the set of nodes V guarantees that for any two nodes $u, v \in V$, $u \preceq v$ if and only if $N(u) \subseteq N[v]$. The *dual preorder* $\preceq^*$ is defined by: $u \preceq^* v$ if and only if $v \preceq u$.

A graph $G = (V, E)$ is *nearly three-threshold* if it is possible to partition the set of nodes V into three classes V_K, V_{S_1}, V_{S_2} so that:

   a) The subgraph induced by $K \cup S_1$ is a threshold graph.,
   b) The subgraph induced by $K \cup S_2$ is a threshold graph.
   c) The subgraph induced by $S_1 \cup S_2$ is a bipartite graph.

Furthermore, the total vicinal preorder related to the graph induced by $K \cup S_2$ is the dual of the total vicinal preorder defined by the graph induced by $K \cup S_1$ (see Figure 13(a)).
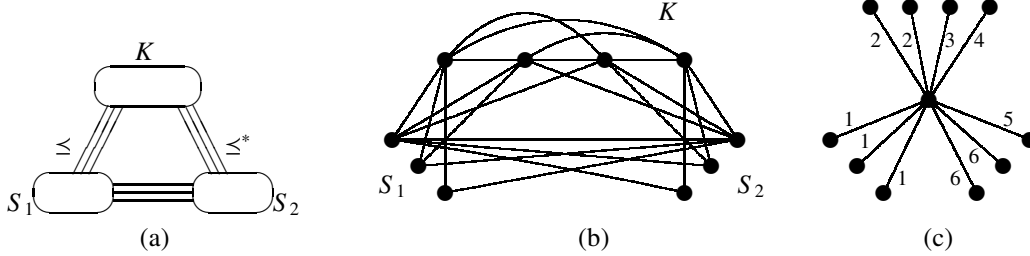

Fig. 13: (a) The structure of a PCG generated by a star; (b) the PCG generated by the star depicted in (c).

THEOREM 7.1. *[20] If a graph G is a PCG of a star then G is a nearly three-threshold graph.*

Open problem: Determine whether the class of graphs that are PCGs of a star coincides with the class of nearly three-threshold graphs.

7.2. Caterpillars. Another important considered tree structure is the caterpillar. PCGs of caterpillars are very general graphs, so we first consider a simplified model, i.e. we assume that $w(e) = 1$ for each edge of the tree. Observe that this restriction is natural as in many papers (e.g. see [3, 39]) the tree is not weighted and the distance is defined as the number of edges on the (unique) path connecting two leaves.

The problem of characterizing PCGs of unit weight caterpillars has been considered in [4] in the special case of LPGs, providing the following result:

THEOREM 7.2. *[4] Let G be an n node connected graph and Γ_n be a unit weight n leaf caterpillar. Then the following statements are equivalent:*

1. *$G = LPG(\Gamma_n, d_{max})$;*
2. *G is a unit interval graph.*

In [13], the authors generalize the previous result to PCGs of unit weight caterpillars:

THEOREM 7.3. *[13] Let G be an n node connected graph, Γ_n be a unit weight n leaf caterpillar. Then the following statements are equivalent:*

1. *$G = PCG(\Gamma_n, d_{min}, d_{max})$;*
2. *$G = P_n^{d_{max}-2} - P_n^{d_{min}-3}$ if $d_{min} > 3$ and $G = P_n^{d_{max}-2}$ otherwise, where P_n^i is the $i-th$ power of the n-node path.*

The authors of [13] generalize then the model to general weighted caterpillars, giving some properties of the resulting PCG. In particular, they give some conditions on the weight function w and on d_{max} so that $PCG(\Gamma_n, w, d_{min}, d_{max})$ is either triangle free or has an induced clique.

Unfortunately, we are far from giving a characterization. So the following open problem arises:

Open problem: Give a complete characterization of PCGs of caterpillars.

7.3. Other trees. It is worth to mention that the 7 node wheel W_7 is proved to be a PCG but not PCG of a caterpillar [13], and the witness tree is shown in Figure 14(b) [14]. As a consequence, caterpillars cannot generate all PCGs, and this fact makes even more significant the last open problem of the previous section.
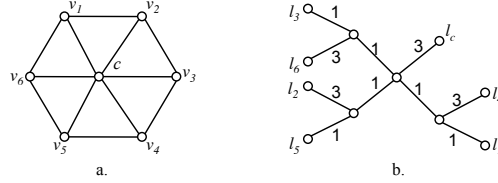

Fig. 14: (a) The wheel W_7 and (b) the edge-weighted tree T such that $W_7 = PCG(T, w, 5, 7)$.

Open problem: It is not known whether wheels on at least eight nodes are PCGs or not.

Given a graph known to be in PCG, finding its witness tree it is far from trivial. A brute force approach is unfeasible as there are too many n-leaf trees to check (and, on each of them, it is necessary to check all possible edge weights). The following result goes in the direction of simplifying the search of the topology providing a unifying tree structure.

THEOREM 7.4. *[14] Let G be a graph, and T a tree. If $G = PCG(T, w, d_{min}, d_{max})$, then there always exist a full binary tree Λ, a new edge-weight function w', and a new value d'_{max} such that $G = PCG(\Lambda, w', d_{min}, d'_{max})$.*

Unfortunately, the previous theorem does not guarantee to have a unique tree, but it is anyway an improvement in practice for the pairwise compatibility tree construction problem, as it leads to consider only a particular subclass of all the n leaf trees.

8. Conclusions. Pairwise compatibility graphs were introduced in the context of phylogenetics and generalize the well-studied class of leaf power graphs. Much attention has been dedicated to them in the literature. However, as shown by this survey, many problems remain open and we are still far from a complete characterization of the PCG class. Any progress towards the latter problem would be interesting not only from a graph theory perspective but also could help in the design of better sampling algorithms for phylogenetic trees. Finally, we conclude by observing that lately it is becoming more and more evident that phylogenetic networks may provide an alternative to phylogenetic trees and may be more suitable for datasets where evolution involves significant amounts of reticulate events such as hybridization, horizontal gene transfer or recombination [1, 40]. Thus, except for the many already existing open problems in this area, it could be interesting to consider possible extensions of these problems and concepts to network graphs.

REFERENCES

[1] E. Bapteste, M. O'Malley, R. Beiko, M. Ereshefsky, J. P. Gogarten, L. Franklin-Hall, F-J. Lapointe, J. Dupre, T. Dagan, Y. Boucher, and W. Martin. Prokaryotic evolution and the tree of life are two different things. *Biology Direct*, 4(1):34, 2009.

[2] E. Bibelnieks and P.M. Dearing. Neighborhood subtree tolerance graphs. *Discrete Applied Mathematics*, 43(1):13 – 26, 1993.

[3] A. Brandstädt. On leaf powers. *Technical report, University of Rostock*, 2010.

[4] A. Brandstädt and C. Hundt. Ptolemaic graphs and interval graphs are leaf powers. In EduardoSany Laber, Claudson Bornstein, LoanaTito Nogueira, and Luerbio Faria, editors, *LATIN 2008: Theoretical Informatics*, volume 4957 of *Lecture Notes in Computer Science*, pages 479–491. Springer Berlin Heidelberg, 2008.

[5] A. Brandstädt, C. Hundt, F. Mancini, and P. Wagner. Rooted directed path graphs are leaf powers. *Discrete Mathematics*, 310(4):897–910, 2010.

[6] A. Brandstädt and V. B. Le. Structure and linear time recognition of 3-leaf powers. *Information Processing Letters*, 98(4):133–138, May 2006.

[7] A. Brandstädt, V. B. Le, and D. Rautenbach. Distance-hereditary 5-leaf powers. *Discrete Mathematics*, 309:3843–3852, 2009.

[8] A. Brandstädt, V. B. Le, and D. Rautenbach. Exact leaf powers. *Theoretical Computer Science*, 411(31–33):2968–2977, 2010.

[9] A. Brandstädt, V. B. Le, and J. Spinrad. *Graph Classes: A Survey*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1999.

[10] A. Brandstädt, V. B. Le, and R. Sritharan. Structure and linear-time recognition of 4-leaf powers. *ACM Trans. Algorithms*, 5(1):11:1–11:22, December 2008.

[11] A. Brandstädt and P. Wagner. On *k*- versus (*k* + 1)-leaf powers. In Boting Yang, Ding-Zhu Du, and CaoAn Wang, editors, *Combinatorial Optimization and Applications*, volume 5165 of *Lecture Notes in Computer Science*, pages 171–179. Springer Berlin Heidelberg, 2008.

[12] A. Brandstädt and P. Wagner. The complete inclusion structure of leaf power classes. *Theoretical Computer Science*, 410:5505–5514, 2009.

[13] T. Calamoneri, A. Frangioni, and B. Sinaimeri. Pairwise compatibility graphs of caterpillars. *Comput. J.*, 57(11):1616–1623, 2014.

[14] T. Calamoneri, D. Frascaria, and B. Sinaimeri. All graphs with at most seven vertices are pairwise compatibility graphs. *Comput. J.*, 56(7):882–886, 2013.

[15] T. Calamoneri, M. Gastaldello, and B. Sinaimeri. On pairwise compatibility of some graph (super)classes. arXiv:1504.06454, 2015.

[16] T. Calamoneri, E. Montefusco, R. Petreschi, and B. Sinaimeri. Exploring pairwise compatibility graphs. *Theoretical Computer Science*, 468:23–26, 2013.

[17] T. Calamoneri and R. Petreschi. Corrigendum to "On pairwise compatibility graphs having Dilworth number two [Theor. Comput. Sci. 524 (2013) 34-40]. *Theoretical Computer Science*. (to appear).

[18] T. Calamoneri and R. Petreschi. On pairwise compatibility graphs having Dilworth number *k*. *Theoretical Computer Science*, 547:82 –89, 2014.

[19] T. Calamoneri and R. Petreschi. On pairwise compatibility graphs having Dilworth number two. *Theoretical Computer Science*, 524:34 –40, 2014.

[20] T. Calamoneri, R. Petreschi, and B. Sinaimeri. On relaxing the constraints in pairwise compatibility graphs. In Md.Saidur Rahman and Shin-ichi Nakano, editors, *WALCOM: Algorithms and Computation*, volume 7157 of *Lecture Notes in Computer Science*, pages 124–135. Springer Berlin Heidelberg, 2012.

[21] T. Calamoneri, R. Petreschi, and B. Sinaimeri. Pairwise compatibility property of some superclasses of threshold graphs. *Discrete Mathematics, Algorithms and Applications*, 05(02):1360002, 2013.

[22] T. Calamoneri and B. Sinaimeri. Relating threshold tolerance graphs to other graph classes. In *Proceedings of the 15th Italian Conference on Theoretical Computer Science, Perugia, Italy, September 17-19, 2014.*, pages 73–79, 2014.

[23] M.-S. Chang and M.-T. Ko. The 3-steiner root problem. In A. Brandstädt, D. Kratsch, and H. Müller, editors, *WG*, volume 4769 of *Lecture Notes in Computer Science*, pages 109–120. Springer, 2007.

[24] Z.-Z. Chen, T. Jiang, and G. Lin. Computing phylogenetic roots with bounded degrees and errors. *SIAM Journal of Computing*, 32(4):864–879, 2003.

[25] V. Chvatal and P.L. Hammer. Aggregation of inequalities in integer programming. In B.H. Korte P.L. Hammer, E.L. Johnson and G.L. Nemhauser, editors, *Studies in Integer Programming*, volume 1 of *Annals of Discrete Mathematics*, pages 145 – 162. Elsevier, 1977.

[26] H.N. de Ridder et al. Information System on Graph Classes and their Inclusions(ISGCI). http://www.graphclasses.org.

[27] R. P. Dilworth. A decomposition theorem for partially ordered sets. *Annals of Mathematics*, 51:161–166, 1950.

[28] M. Dom, J. Guo, F. Hüffner, and R. Niedermeier. Error compensation in leaf root problems. In Rudolf Fleischer and Gerhard Trippen, editors, *Algorithms and Computation*, volume 3341 of *Lecture Notes in Computer Science*, pages 389–401. Springer Berlin Heidelberg, 2005.

[29] M. Dom, J. Guo, F. Hüffner, and R. Niedermeier. Extending the tractability border for closest leaf powers. In D. Kratsch, editor, *WG*, volume 3787 of *Lecture Notes in Computer Science*, pages 397–408. Springer, 2005.

[30] S. Durocher, D. Mondal, and Md. S. Rahman. On graphs that are not PCGs. *Theoretical Computer Science*, 2015. in press.

[31] M. Farber. Characterizations of strongly chordal graphs. *Discrete Mathematics*, 43(2V3):173–189, 1983.

[32] J. Felsenstein. Cases in which parsimony or compatibility methods will be positively misleading. *Systematic Zoology*, 27:401–410, 1978.

[33] S. Földes and P.L. Hammer. Split graphs. *8th South–Eastern Conf. on Combinatorics, Graph Theory and Computing (F. HOFFMAN et al., eds.) Louisiana StateUniv. Baton Rouge, Louisiana, Congressus Numerantium*, 19:311–315, 1977.

[34] F. Gavril. A recognition algorithm for the intersection graphs of directed paths in directed trees. *Discrete Mathematics*, 13(3):237 – 249, 1975.

[35] M. C. Golumbic, C. L. Monma, and W. T. Trotter. Tolerance graphs. *Discrete Applied Mathematics*, 9(2):157–170, 1984.

[36] F. Harary and A.J. Schwenk. The number of caterpillars. *Discrete Mathematics*, 6(4):359–365, 1973.

[37] R. B. Hayward, P. E. Kearney, and A. Malton. Nest graphs. *Discrete Applied Mathematics*, 121(1–3):139–153, 2002.

[38] B. Jamison and S. Olariu. Recognizing P_4-sparse graphs in linear time. *SIAM J. Comput.*, 21(2):381–406, 1992.

[39] P.E. Kearney, J. I. Munro, and D. Phillips. Efficient generation of uniform samples from phylogenetic trees. In Gary Benson and RodericD.M. Page, editors, *Algorithms in Bioinformatics*, volume 2812 of *Lecture Notes in Computer Science*, pages 177–189. Springer Berlin Heidelberg, 2003.

[40] F. J. Lapointe, P. Lopez, Y. Boucher, J. Koenig, and E. Bapteste. Clanistics: a multi-level perspective for harvesting unrooted gene trees. *Trends Microbiology*, 18(8):341–347, 2010.

[41] G. Lin, P. E. Kearney, and T. Jiang. Phylogenetic k-root and steiner k-root. In Gerhard Goos, Juris Hartmanis, Jan van Leeuwen, D.T. Lee, and Shang-Hua Teng, editors, *Algorithms and Computation*, volume 1969 of *Lecture Notes in Computer Science*, pages 539–551. Springer Berlin Heidelberg, 2000.

[42] N.V.R. Mahadev and U.N. Peled. *Threshold graphs and related topics*. Annals of Discrete Mathematics. Elsevier, Burlington, MA, 1995.

[43] S. Mehnaz and M.S. Rahman. Pairwise compatibility graphs revisited. In *Informatics, Electronics Vision (ICIEV), 2013 International Conference on*, pages 1–6, 2013.

[44] C.L. Monma, B. Reed, and W.T. Trotter Jr. A generalization of threshold graphs with tolerance. *Congressus Numerantium*, 55:187–197, 1986.

[45] C.L. Monma, B. Reed, and W.T. Trotter Jr. Threshold tolerance graphs. *J. Graph Theory*, 12:343–362, 1988.

[46] R. Nevries and C. Rosenke. Towards a characterization of leaf powers by clique arrangements. In *SOFSEM 2015: Theory and Practice of Computer Science*, volume 8939 of *Lecture Notes in Computer Science*, pages 364–376. Springer Berlin Heidelberg, 2015.

[47] N. Nishimura, P. Ragde, and D. M. Thilikos. On graph powers for leaf-labeled trees. *Journal of Algorithms*, 42(1):69–108, 2002.

[48] D. Phillips. Uniform sampling from phylogenetic trees. *Master Degree - Univ. of Waterloo, Canada*, 2002.

[49] D. Rautenbach. Some remarks about leaf roots. *Discrete Mathematics*, 306(13):1456 – 1461, 2006.

[50] A. Raychaudhuri. On powers of strongly chordal and circular graphs. *Ars Combin.*, 34:147–160, 1992.

[51] D. Mondal S. Durocher and M.S. Rahman. On graphs that are not pcgs. In Subir Kumar Ghosh and Takeshi Tokuyama, editors, *WALCOM: Algorithms and Computation*, volume 7748 of *Lecture Notes in Computer Science*, pages 310–321. Springer Berlin Heidelberg, 2013.

[52] S.A. Salma, Md. S. Rahman, and Md. I. Hossain. Triangle-free outerplanar 3-graphs are pairwise compatibility graphs. *J. Graph Algorithms Appl.*, 17(2):81–102, 2013.

[53] R. Tyshkevich. Decomposition of graphical sequences and unigraphs. *Discrete Mathematics*, 220(1C3):201 – 238, 2000.

[54] M.N. Yanhaona, Md. S. Bayzid, and Md. S. Rahman. Discovering pairwise compatibility graphs. *Discrete Math., Alg. and Appl.*, 2(4):607–624, 2010.

[55] M.N. Yanhaona, K.S.M. T. Hossain, and Md. S. Rahman. Pairwise compatibility graphs. *J. Appl. Math. Comput.*, 30:479–503, 2009.