The Recognition of Simple-Triangle Graphs and of Linear-Interval Orders is Polynomial∗ †

George B. Mertzios

School of Engineering and Computing Sciences, Durham University, UK.
Email: george.mertzios@durham.ac.uk

Abstract. Intersection graphs of geometric objects have been extensively studied, both due to their interesting structure and their numerous applications; prominent examples include interval graphs and permutation graphs. In this paper we study a natural graph class that generalizes both interval and permutation graphs, namely simple-triangle graphs. Simple-triangle graphs – also known as PI graphs (for Point-Interval) – are the intersection graphs of triangles that are defined by a point on a line $L_1$ and an interval on a parallel line $L_2$. They lie naturally between permutation and trapezoid graphs, which are the intersection graphs of line segments between $L_1$ and $L_2$ and of trapezoids between $L_1$ and $L_2$, respectively. Although various efficient recognition algorithms for permutation and trapezoid graphs are well known to exist, the recognition of simple-triangle graphs has remained an open problem since their introduction by Corneil and Kamula three decades ago. In this paper we resolve this problem by proving that simple-triangle graphs can be recognized in polynomial time. As a consequence, our algorithm also solves a longstanding open problem in the area of partial orders, namely the recognition of linear-interval orders, i.e. of partial orders $P = P_1 \cap P_2$, where $P_1$ is a linear order and $P_2$ is an interval order. This is one of the first results on recognizing partial orders $P$ that are the intersection of orders from two different classes $P_1$ and $P_2$. In complete contrast to this, partial orders $P$ which are the intersection of orders from the same class $P$ have been extensively investigated, and in most cases the complexity status of these recognition problems has been already established.

Keywords: Intersection graphs, PI graphs, recognition problem, partial orders, polynomial algorithm.

1 Introduction

A graph $G$ is the intersection graph of a family $\mathcal{F}$ of sets if we can bijectively assign sets of $\mathcal{F}$ to vertices of $G$ such that two vertices of $G$ are adjacent if and only if the corresponding sets have a non-empty intersection. It turns out that many graph classes with important applications can be described as intersection graphs of set families that are derived from some kind of geometric configuration. One of the most prominent examples is that of interval graphs, i.e. the intersection graphs of intervals on the real line, which have natural applications in several fields, including bioinformatics and involving the physical mapping of DNA and the genome reconstruction.

Generalizing the intersections on the real line, consider two parallel horizontal lines on the plane, $L_1$ (the upper line) and $L_2$ (the lower line). A graph $G$ is a simple-triangle graph if it is the intersection graph of triangles that have one endpoint on $L_1$ and the other two on $L_2$. Furthermore, $G$ is a triangle graph if it is the intersection graph of triangles with endpoints on $L_1$ and $L_2$, but now there is no restriction on which line contains one endpoint of every triangle and which contains the other two. Simple-triangle and triangle graphs are also known as PI and PT graphs, respectively [3, 6, 22], where PI stands for “Point-Interval”. Such representations of simple-triangle and of triangle graphs are called simple-triangle (or PI) and triangle (or PT∗) representations, respectively. Simple-triangle and triangle graphs lie naturally between permutation graphs (i.e. the intersection graphs of line segments with one endpoint on $L_1$ and one on $L_2$) and trapezoid graphs (i.e. the intersection graphs of trapezoids with one interval on $L_1$ and the opposite interval on $L_2$) [3, 22]. Note that, using the notation PI for simple-triangle graphs, permutation graphs are PP (for “Point-Point”) graphs, while trapezoid graphs are II (for “Interval-Interval”) graphs [6].

A partial order is a pair $P = (U, R)$, where $U$ is a finite set and $R$ is an irreflexive transitive binary relation on $U$. Whenever $(x, y) \in R$ for two elements $x, y \in U$, we write $x <_P y$. If $x <_P y$ or $y <_P x$, then $x$ and $y$ are comparable, otherwise they are incomparable. $P$ is a linear order if every pair of elements in $U$...
are comparable. Furthermore, \( P \) is an interval order if each element \( x \in U \) is assigned to an interval \( I_x \) on the real line such that \( x < y \) if and only if \( I_x \) lies completely to the left of \( I_y \). One of the most fundamental notions on partial orders is dimension. For any partial order \( P \) and any class \( \mathcal{P} \) of partial orders (e.g. linear order, interval order, semiorder, etc.), the \( \mathcal{P}\)-dimension of \( P \) is the smallest \( k \) such that \( P \) is the intersection of \( k \) orders from \( \mathcal{P} \). In particular, when \( \mathcal{P} \) is the class of linear orders, the \( \mathcal{P}\)-dimension of \( P \) is known as the dimension of \( P \). Although in most cases we can efficiently recognize whether a partial order belongs to a class \( \mathcal{P} \), this is not the case for higher dimensions. Due to a classical result of Yannakakis [23], it is NP-complete to decide whether the dimension, or the interval dimension, of a partial order is at most \( k \), where \( k \geq 3 \).

There is a natural correspondence between graphs and partial orders. For a partial order \( P = (U, R) \), the comparability (resp. incomparability) graph \( G(P) \) of \( P \) has elements of \( U \) as vertices and an edge between every pair of comparable (resp. incomparable) elements. A graph \( G \) is a (co)comparability graph if \( G \) is the (in)comparability graph of a partial order \( P \). There has been a long line of research in order to establish the complexity of recognizing partial orders of \( \mathcal{P}\)-dimension at most 2 (e.g. where \( \mathcal{P} \) is linear orders or interval orders [13]). In particular, since permutation (resp. trapezoid) graphs are the incomparability graphs of partial orders with dimension (resp. interval dimension) at most 2 [12,22], permutation and trapezoid graphs can be recognized efficiently by the corresponding partial order algorithms [15,22].

In contrast, not much is known so far for the recognition of partial orders \( P \) that are the intersection of orders from different classes \( P_1 \) and \( P_2 \). One of the longstanding open problems in this area is the recognition of linear-interval orders \( P \), i.e. of partial orders \( P = P_1 \cap P_2 \), where \( P_1 \) is a linear order and \( P_2 \) is an interval order. In terms of graphs, this problem is equivalent to the recognition of simple-triangle (i.e. PI) graphs, since PI graphs are the incomparability graphs of linear-interval orders; this problem is well known and remains open since the introduction of PI graphs in 1987 [8] (cf. for instance the books [3,22]).

**Our contribution.** In this article we establish the complexity of recognizing simple-triangle (PI) graphs, and therefore also the complexity of recognizing linear-interval orders. Given a graph \( G \) with \( n \) vertices, such that its complement \( \bar{G} \) has \( m \) edges, we provide an algorithm with running time \( O(n^2m) \) that either computes a PI representation of \( G \), or it announces that \( G \) is not a PI graph. Equivalently, given a partial order \( P = (U, R) \) with \( |U| = n \) and \( |R| = m \), our algorithm either computes in \( O(n^2m) \) time a linear order \( P_1 \) and an interval order \( P_2 \) such that \( P = P_1 \cap P_2 \), or it announces that such orders \( P_1, P_2 \) do not exist. Surprisingly, it turns out that the seemingly small difference in the definition of simple-triangle (PI) graphs and triangle (PI*) graphs results in a very different behavior of their recognition problems; only recently it has been proved that the recognition of triangle graphs is NP-complete [18]. In addition, our polynomial time algorithm is in contrast to the recognition problems for the related classes of bounded tolerance (i.e. parallelogram) graphs [19] and of max-tolerance graphs [14], which have already been proved to be NP-complete.

As the main tool for our algorithm we introduce the notion of a linear-interval cover of bipartite graphs. As a second tool we identify a new tractable subclass of 3SAT, called gradually mixed formulas, for which we provide a linear time algorithm. The class of gradually mixed formulas is hybrid, i.e. it is characterized by both relational and structural restrictions on the clauses. Then, using the notion of a linear-interval cover, we are able to reduce our problem to the satisfiability problem of gradually mixed formulas.

Our algorithm proceeds as follows. First, it computes from the given graph \( G \) a bipartite graph \( \tilde{G} \), such that \( G \) is a PI graph if and only if \( \tilde{G} \) has a linear-interval cover. Second, it computes a gradually mixed Boolean formula \( \phi \) such that \( \phi \) is satisfiable if and only if \( \tilde{G} \) has a linear-interval cover. This formula \( \phi \) can be written as \( \phi = \phi_1 \land \phi_2 \), where every clause of \( \phi_1 \) has 3 literals and every clause of \( \phi_2 \) has 2 literals. The construction of \( \phi_1 \) and \( \phi_2 \) is based on the fact that a necessary condition for \( \tilde{G} \) to admit a linear-interval cover is that its edges can be colored with two different colors (according to some restrictions). Then the edges of \( \tilde{G} \) correspond to literals of \( \phi \), while the two edge colors encode the truth value of the corresponding variables. Furthermore every clause of \( \phi_1 \) corresponds to the edges of an alternating cycle in \( \tilde{G} \) (i.e. a closed walk that alternately visits edges and non-edges) of length 6, while the clauses of \( \phi_2 \) correspond to specific pairs of edges of \( \tilde{G} \) that are not allowed to receive the same color. Finally, the equivalence between the existence of a linear-interval cover of \( \tilde{G} \) and a satisfying truth assignment for \( \phi \) allows us to use our linear algorithm to solve satisfiability on gradually mixed formulas in order to complete our recognition algorithm.

**Organization of the paper.** We present in Section 2 the class of gradually mixed formulas and a linear time algorithm to solve satisfiability on this class. In Section 3 we provide the necessary notation and preliminaries on threshold graphs and alternating cycles. Then in Section 4 we introduce the notion of a linear-interval cover of bipartite graphs to characterize PI graphs, and in Section 5 we translate the linear-interval cover problem to the satisfiability problem on a gradually mixed formula. Finally, in Section 6 we present our PI graph recognition algorithm.
2 A tractable subclass of 3SAT

In this section we introduce the class of \textit{gradually mixed} formulas and we provide a linear time algorithm for solving satisfiability on this class. Any gradually mixed formula \( \phi \) is a mix of binary and ternary clauses. That is, there exist a 3-CNF formula \( \phi_1 \) (i.e. a formula in conjunctive normal form with at most 3 literals per clause) and a 2-CNF formula \( \phi_2 \) (i.e. with at most 2 literals per clause) such that \( \phi = \phi_1 \land \phi_2 \), while \( \phi \) satisfies some constraints among its clauses. Before we define gradually mixed formulas (cf. Definition 2), we first define dual clauses.

**Definition 1.** Let \( \phi_1 \) be a 3-CNF formula. If \( \alpha = (\ell_1 \lor \ell_2 \lor \ell_3) \) is a clause of \( \phi_1 \), then the \( \overline{\alpha} = (\overline{\ell_1} \lor \overline{\ell_2} \lor \overline{\ell_3}) \) is the dual clause of \( \alpha \).

Note by Definition 1 that, whenever \( \alpha \) is a clause of a formula \( \phi_1 \), the dual clause \( \overline{\alpha} \) of \( \alpha \) may belong, or may not belong, to \( \phi_1 \).

**Definition 2.** Let \( \phi_1 \) and \( \phi_2 \) be CNF formulas with 3 literals and 2 literals in each clause, respectively. The mixed formula \( \phi = \phi_1 \land \phi_2 \) is gradually mixed if the next two conditions are satisfied:

1. Let \( \alpha \) and \( \beta \) be two clauses of \( \phi_1 \). Then \( \alpha \) does not share exactly one literal with either the clause \( \beta \) or the clause \( \overline{\beta} \).

2. If \( \alpha = (\ell_1 \lor \ell_2 \lor \ell_3) \) is a clause of \( \phi_1 \) and \( (\ell_0 \lor \ell_1) \) is a clause of \( \phi_2 \), then \( \phi_2 \) contains also (at least) one of the clauses \( \{(\ell_0 \lor \ell_2), (\ell_0 \lor \ell_3)\} \).

As an example of a gradually mixed formula, consider the formula \( \phi = \phi_1 \land \phi_2 \), where \( \phi_1 = (x_1 \lor \overline{x_2} \lor x_3) \land (\overline{x_1} \lor x_2 \lor x_4) \land (x_5 \lor x_6 \lor \overline{x_7}) \) and \( \phi_2 = (x_2 \lor \overline{x_7}) \land (x_8 \lor x_1) \land (x_8 \lor x_4) \land (\overline{x_2} \lor x_9) \land (x_5 \lor x_{10}) \land (\overline{x_6} \lor x_{10}) \).

Note by Definition 2 that the class of gradually mixed formulas contains 2SAT as a proper subclass, since every 2-CNF formula \( \phi_2 \) can be written as a gradually mixed formula \( \phi = \phi_1 \land \phi_2 \) where \( \phi_1 = \emptyset \). Furthermore the class of gradually mixed formulas \( \phi \) is a hybrid class, since the conditions of Definition 2 concern simultaneously relational restrictions (i.e. where the clauses are restricted to be of certain types) and structural restrictions (i.e. where there are restrictions on how different clauses interact with each other).

The intuition for the term \textit{gradually mixed} in Definition 2 is that, whenever the sub-formulas \( \phi_1 \) and \( \phi_2 \) share more variables, the number of clauses of \( \phi_2 \) that are imposed by condition 2 of Definition 2 increases. In the next theorem we use resolution to prove that satisfiability can be solved in linear time on gradually mixed formulas.

**Theorem 1.** There exists a linear time algorithm which decides whether a given gradually mixed formula \( \phi \) is satisfiable and computes a satisfying truth assignment of \( \phi \), if one exists.

**Proof.** Let \( \phi = \phi_1 \land \phi_2 \), where \( \phi_1 \) is a 3-CNF formula and \( \phi_2 \) is a 2-CNF formula. We first scan through all clauses of \( \phi \) to remove all tautologies, i.e. all clauses which contain both a literal and its negation, since such clauses are always satisfiable. Furthermore we eliminate all double literal occurrences in each clause.

In the remainder of the proof we denote by \( \phi \) the resulting formula after the removal of tautologies and the elimination of double literal occurrences in the clauses. Note that, during this elimination procedure, some clauses of \( \phi_1 \) may become 2-CNF clauses. In the resulting formula we denote by \( \phi'_1 \) the conjunction of the clauses that have 3 literals each, and by \( \phi''_1 \) the conjunction of the clauses of \( \phi_1 \) that remain with 1 or 2 literals each. In particular, since also in every clause of \( \phi_1 \) no literal is the negation of another one (as we removed from \( \phi \) all tautologies), the literals of every clause in \( \phi'_1 \) correspond to three distinct variables.

Then we compute a 2-CNF formula \( \phi_0 \) (in time linear to the size of \( \phi \)) as follows. Initially \( \phi_0 \) is empty. First we mark all literals \( \ell \) for which the 2-CNF formula \( \phi''_1 \land \phi_2 \) includes the clause \( \ell \). Then we scan through all clauses of the 3-CNF formula \( \phi'_1 \). For every clause \( (\ell_1 \lor \ell_2 \lor \ell_3) \) of \( \phi'_1 \), such that the literal \( \overline{\ell_1} \) (resp. \( \overline{\ell_2} \) or \( \overline{\ell_3} \)) has been marked, we add to \( \phi' \) the clause \( (\ell_2 \lor \ell_3) \) (resp. the clause \( (\ell_1 \lor \ell_3) \) or \( (\ell_1 \lor \ell_2) \)).

If \( \phi \land \phi_0 \) is satisfiable then clearly \( \phi \) is also satisfiable as a sub-formula of \( \phi \land \phi_0 \). Conversely, suppose that \( \phi \) is satisfiable by the truth assignment \( \tau \). Let \( \gamma = (\ell_1 \lor \ell_2) \) be an arbitrary clause of \( \phi_0 \). The existence of \( \gamma \) in \( \phi_0 \) implies the existence of some clauses \( \alpha = (\overline{\ell_1}) \) and \( \beta = (\ell_1 \lor \ell_2 \lor \ell_3) \) in \( \phi \). Therefore, since \( \alpha = \beta = 1 \) in \( \tau \) by assumption, it follows that \( \ell_3 = 0 \) in \( \tau \). Thus the clause \( \beta \) equals \( (\ell_1 \lor \ell_2) \) in \( \tau \), and therefore \( \gamma = 1 \) in \( \tau \). That is, \( \tau \) satisfies also \( \phi_0 \). Therefore \( \phi \) is satisfiable if and only if \( \phi \land \phi_0 \) is satisfiable.

In the remainder of the proof, we prove that \( \phi \land \phi_0 \) is satisfiable if and only if the 2-CNF formula \( \phi''_1 \land \phi_2 \land \phi_0 \) is satisfiable. The one direction is immediate, i.e. if \( \phi \land \phi_0 \) is satisfiable then \( \phi''_1 \land \phi_2 \land \phi_0 \) is also satisfiable as a sub-formula of \( \phi \land \phi_0 \). Conversely, suppose that \( \phi''_1 \land \phi_2 \land \phi_0 \) is satisfiable and let \( \tau \) be a
satisfying truth assignment of this formula. If \( \tau \) satisfies all clauses of \( \phi_1 \), then clearly \( \tau \) is also a satisfying truth assignment of \( \phi \wedge \phi_0 \). Otherwise let \( \alpha = (\ell_1 \lor \ell_2 \lor \ell_3) \) be a clause of \( \phi_1 \) that is not satisfied by \( \tau \). Then \( \ell_1 = \ell_2 = \ell_3 = 0 \) in \( \tau \). In this case, we construct the truth assignment \( \tau' \) from \( \tau \) by flipping the value of one (arbitrary) literal of \( \{\ell_1, \ell_2, \ell_3\} \) in \( \tau \). Assume without loss of generality that the value of \( \ell_1 \) flips from \( \tau \) to \( \tau' \), while the values of all other variables remain the same in both \( \tau \) and \( \tau' \). Recall that the literals \( \{\ell_1, \ell_2, \ell_3\} \) correspond to three distinct variables, since we eliminated all double occurrences of literals in all clauses in \( \phi_1 \). Therefore \( \ell_1 = \ell_2 = \ell_3 = 1 \) in \( \tau' \), and thus \( \alpha = 1 \) in \( \tau' \).

Suppose that there exists a clause \( \beta = (\ell_4 \lor \ell_5 \lor \ell_6) \) of \( \phi_1 \) where \( \beta = 1 \) in \( \tau \) and \( \beta = 0 \) in \( \tau' \). Then clearly one of the literals of \( \beta \) equals \( \ell_1 \), since \( \ell_1 \) is the only literal whose value changes in \( \tau' \) from 1 to 0. Assume without loss of generality that \( \ell_4 = \ell_1 \), i.e. \( \alpha \) shares at least one literal with \( \beta = (\ell_1 \lor \ell_5 \lor \ell_6) \). Therefore, since \( \phi \) is a gradually mixed formula by assumption, it follows by Definition 2 that \( \alpha \) shares at least one more literal with \( \beta \). Assume without loss of generality that \( \ell_2 = \ell_5 \). Then, since by assumption \( \ell_2 = 0 \) in both \( \tau \) and \( \tau' \), it follows that the clause \( \beta = (\ell_4 \lor \ell_5 \lor \ell_6) = (\ell_1 \lor \ell_6 \lor \ell_6) \) is satisfied in \( \tau' \), which is a contradiction to our assumption. Therefore for every clause \( \beta \) of \( \phi_1' \), if \( \beta = 1 \) in \( \tau \) then also \( \beta = 1 \) in \( \tau' \).

We now prove that all clauses of the 2-CNF formula \( \phi_1'' \wedge \phi_2 \wedge \phi_0 \) remain satisfied in \( \tau' \). First consider an arbitrary clause \( \gamma \) of \( \phi_0 \) that contains one of the literals \( \{\ell_1, \ell_1\} \). If \( \gamma \) contains the literal \( \ell_1 \), then \( \gamma = 1 \) in \( \tau' \), since \( \ell_1 = 1 \) in \( \tau' \). Let \( \gamma \) contain the literal \( \ell_1 \), and let \( \gamma = (\ell_1 \lor \ell_4) \). Then it follows by the construction of the formula \( \phi_0 \) that there exists a literal \( \ell_2 \), such that \( (\ell_1 \lor \ell_4) \) is a clause of \( \phi_1 \) by assumption, and \( \phi_0 \) contains (by construction) the clause \( (\ell_2 \lor \ell_4) \). Then, since \( \tau \) satisfies \( \phi_0 \) by assumption, it follows that \( \ell_2 = 1 \) or \( \ell_2 = 1 \) in \( \tau \). This is a contradiction, since \( \ell_1 = \ell_2 = \ell_3 = 0 \) in \( \tau \). Therefore \( \ell_4 \notin \{\ell_1, \ell_4\} \). Thus, since \( \phi \) is a gradually mixed formula by assumption, it follows by Definition 2 that \( \phi_0 \) has also one of the clauses \( \{\ell_1 \lor \ell_4, \ell_4 \lor \ell_4\} \). Assume without loss of generality that \( \phi_0 \) has the clause \( (\ell_4 \lor \ell_2) \). Then, since \( \tau \) satisfies \( \phi_2 \) by assumption and \( \ell_2 = 0 \) in \( \tau \), it follows that \( \ell_4 = 1 \) in \( \tau \). Furthermore, since \( \ell_4 \notin \{\ell_1, \ell_4\} \), it remains \( \ell_4 = 1 \) in \( \tau' \), and thus \( \gamma = (\ell_1 \lor \ell_4) = 1 \) in \( \tau' \). That is, all clauses \( \gamma \) of \( \phi_2 \) remain satisfied in the assignment \( \tau' \).

Finally consider a clause \( \gamma \) of \( \phi_0'' \) that contains one of the literals \( \{\ell_1, \ell_1\} \). If \( \gamma \) contains \( \ell_1 \) then \( \gamma = 1 \) in \( \tau \), since \( \ell_1 = 1 \) in \( \tau \), and \( \ell_1 = 1 \) in \( \tau' \). Let \( \gamma \) contain the literal \( \ell_1 \), and let \( \gamma = (\ell_1 \lor \ell_4) \). Note that \( \ell_1 = 1 \), since we removed all tautologies from \( \phi \). Suppose that \( \ell_4 = \ell_1 \), i.e. \( \gamma = (\ell_1 \lor \ell_4) \). Then, since \( \alpha = (\ell_1 \lor \ell_2 \lor \ell_3) \) is a clause of \( \phi_1 \) by assumption, the formula \( \phi_0 \) contains (by construction) the clause \( (\ell_2 \lor \ell_4) \). Thus, since \( \tau \) satisfies \( \phi_0 \) by assumption, it follows that \( \ell_2 = 1 \) or \( \ell_2 = 1 \) in \( \tau \). This is a contradiction, since \( \ell_1 = \ell_2 = \ell_3 = 0 \) in \( \tau \). Therefore \( \ell_4 \notin \{\ell_1, \ell_4\} \). Thus, since \( \phi \) is a gradually mixed formula by assumption, it follows by Definition 2 that \( \phi_0 \) has also one of the clauses \( \{\ell_1 \lor \ell_4, \ell_4 \lor \ell_4\} \). Assume without loss of generality that \( \phi_0 \) has the clause \( (\ell_4 \lor \ell_2) \). Then, since \( \tau \) satisfies \( \phi_2 \) by assumption and \( \ell_2 = 0 \) in \( \tau \), it follows that \( \ell_4 = 1 \) in \( \tau \). Furthermore, since \( \ell_4 \notin \{\ell_1, \ell_4\} \), it remains \( \ell_4 = 1 \) in \( \tau' \), and thus \( \gamma = (\ell_1 \lor \ell_4) = 1 \) in \( \tau' \). That is, all clauses \( \gamma \) of \( \phi_2 \) remain satisfied in the assignment \( \tau' \).

Summarizing, all clauses of the 2-CNF formula \( \phi_1'' \wedge \phi_2 \wedge \phi_0 \) remain satisfied in \( \tau' \). Furthermore, \( \alpha = 1 \) in \( \tau \), while for every clause \( \beta \) of \( \phi_1' \), if \( \beta = 1 \) in \( \tau \) then also \( \beta = 1 \) in \( \tau' \). Thus, according to the above transition from \( \tau \) to \( \tau' \), we can modify iteratively the truth assignment \( \tau \) to a truth assignment \( \tau'' \) that satisfies all clauses of \( \phi \wedge \phi_0 \). Therefore \( \phi \wedge \phi_0 \) is satisfiable if and only if the 2-CNF formula \( \phi_1'' \wedge \phi_2 \wedge \phi_0 \) is satisfiable.

Since the transition from the assignment \( \tau \) to the assignment \( \tau' \) can be done in constant time (we only need to flip locally the value of one literal \( \ell_1 \) in the clause \( \alpha = (\ell_1 \lor \ell_2 \lor \ell_3) \) of \( \phi_1' \)), the computation of \( \tau'' \) from \( \tau \) can be done in time linear to the size of \( \phi \wedge \phi_0 \). Therefore, since a satisfying truth assignment \( \tau \) of the 2-CNF formula \( \phi_1'' \wedge \phi_2 \wedge \phi_0 \) (if one exists) can be computed in linear time using any standard linear time algorithm for the 2-SAT problem (e.g. [8]), a satisfying truth assignment \( \tau'' \) of \( \phi \wedge \phi_0 \) (if one exists) can
be also computed in time linear to the size of \( \phi \land \phi_0 \) (and thus also in time linear to the size of \( \phi \)). This completes the proof of the theorem. \( \square \)

The conditions of Definition 2 which guarantee the tractability of gradually mixed formulas are minimal, in the sense that, if we remove any of these two conditions, the resulting subclass of 3SAT is NP-complete.

Indeed, assume that we impose only the first condition of Definition 2 to the mixed formula \( \phi = \phi_1 \land \phi_2 \). Then we can reduce 3SAT to this subclass as follows. Let \( \phi \) be an instance of 3SAT. We define \( \phi_1 \) to be the formula obtained by \( \phi_0 \) if we replace every literal \( \ell \) of \( \phi_0 \) by a new variable \( x_\ell \). For every two of these new variables \( x_\ell \) and \( x_{\ell'} \) in \( \phi_1 \), we add to \( \phi_2 \) the clauses \( (x_\ell \lor \ell') \land (\ell \lor x_{\ell'}) \) if \( \ell = \ell' \) in \( \phi_0 \), and we add to \( \phi_2 \) the clauses \( (x_\ell \lor x_{\ell'}) \land (\ell \lor \ell') \) if \( \ell = \ell' \) in \( \phi_0 \). Then \( \phi = \phi_1 \land \phi_2 \) satisfies the first condition of Definition 2 (since no two clauses of \( \phi \) share any variable), while \( \phi_0 \) is satisfiable if and only if \( \phi \) is satisfiable.

On the other hand, assume that we impose only the second condition of Definition 2 to the mixed formula \( \phi = \phi_1 \land \phi_2 \). Then, by setting \( \phi_2 = \emptyset \), we can include in the resulting class every 3-CNF formula, and thus this class is NP-complete.

3 Preliminaries

3.1 Notation

In the remainder of this article we consider finite, simple, and undirected graphs. Given a graph \( G \), we denote by \( V(G) \) and \( E(G) \) the sets of its vertices and edges, respectively. An edge between two vertices \( u \) and \( v \) of a graph \( G = (V, E) \) is denoted by \( uv \), and in this case \( u \) and \( v \) are said to be adjacent. The neighborhood of a vertex \( u \in V \) is the set \( N(u) = \{ v \in V \mid uv \in E \} \) of its adjacent vertices. The complement of \( G \) is denoted by \( \overline{G} \), i.e. \( \overline{G} = (V, \overline{E}) \), where \( uv \in \overline{E} \) if and only if \( uv \notin E \). For any subset \( E_0 \subseteq E \) of the edges of \( G \), we denote for simplicity \( G - E_0 = (V, E \setminus E_0) \). A subset \( S \subseteq V \) of its vertices induces an independent set in \( G \) if \( uv \notin E \) for every pair of vertices \( u, v \in S \). Furthermore, \( S \) induces a clique in \( G \) if \( uv \in E \) for every pair \( u, v \in S \). For two graphs \( G_1 = (V, E_1) \) and \( G_2 = (V, E_2) \), we denote \( G_1 \subseteq G_2 \) whenever \( E_1 \subseteq E_2 \). Moreover, we denote for simplicity by \( G_1 \cup G_2 \) and \( G_1 \cap G_2 \) the graphs \( (V, E_1 \cup E_2) \) and \( (V, E_1 \cap E_2) \), respectively. A graph \( G \) is a split graph if its vertices can be partitioned into a clique \( K \) and an independent set \( I \). Furthermore, \( G = (V, E) \) is a threshold graph if we can assign to each vertex \( v \in V \) a real weight \( a_v \), such that \( uv \in E \) if and only if \( a_u + a_v \geq 1 \).

A proper \( k \)-coloring of a graph \( G \) is an assignment of \( k \) colors to the vertices of \( G \), such that adjacent vertices are assigned different colors. The smallest \( k \) for which there exists a proper \( k \)-coloring of \( G \) is the chromatic number of \( G \), denoted by \( \chi(G) \). If \( \chi(G) = 2 \) then \( G \) is a bipartite graph; in this case the vertices of \( G \) are partitioned into two independent sets, the color classes. A bipartite graph \( G \) is denoted by \( G = (U, V, E) \), where \( U \) and \( V \) are its color classes and \( E \) is the set of edges between them. For a bipartite graph \( G = (U, V, E) \), its bipartite complement is the graph \( \overline{G} = (U, V, \overline{E}) \), where for every two vertices \( u \in U \) and \( v \in V \), \( uv \in \overline{E} \) if and only if \( uv \notin E \). A bipartite graph \( G = (U, V, E) \) is a chain graph if the vertices of each color class can be ordered by inclusion of their neighborhoods, i.e. \( N(u) \subseteq N(v) \) or \( N(v) \subseteq N(u) \) for any two vertices \( u, v \) in the same color class. Note that chain graphs are closed under bipartite complementation, i.e. \( G \) is a chain graph if and only if \( \overline{G} \) is a chain graph.

For any graph \( G = (V, E) \) and any graph class \( \mathcal{G} \), the \( \mathcal{G} \)-cover number of \( G \) is the smallest \( k \) such that \( E = \bigcup_{i=1}^{k} E_i \), where \( E_i = (V, E_i) \in \mathcal{G} \), \( 1 \leq i \leq k \); in this case the graphs \( \{G_i\}_{i=1}^{k} \) are a \( \mathcal{G} \)-cover of \( G \). For several graph classes \( \mathcal{G} \) it is NP-complete to decide whether the \( \mathcal{G} \)-cover number of a graph is at most \( k \), where \( k \geq 3 \), see e.g. [23]. Throughout the paper, whenever a set of the chain graphs \( \{G_i\}_{i=1}^{k} \) form a chain-cover of a bipartite graph \( G \), then all these graphs are assumed to have the same color classes as \( G \).

For any partial order \( P = (U, R) \), we denote by \( \overline{P} = (U, \overline{R}) \) the inverse partial order of \( P \), i.e. for any two elements \( u, v \in U \), \( u \prec_P v \) if and only if \( v \prec_P u \). For any two partial orders \( P_1 = (U, R_1) \) and \( P_2 = (U, R_2) \), we denote \( P_1 \subseteq P_2 \) whenever \( R_1 \subseteq R_2 \). Moreover, we denote for simplicity \( P_1 \cup P_2 \) and \( P_1 \cap P_2 \) for the partial orders \( (U, R_1 \cup R_2) \) and \( (U, R_1 \cap R_2) \), respectively. If \( P_2 \) is a linear order and \( P_1 \subseteq P_2 \), then \( P_2 \) is a linear extension of \( P_1 \). The orders \( P_1 \) and \( P_2 \) contradict each other if there exist two elements \( u, v \in U \) such that \( u \prec_P v \) and \( v \prec_P u \). The linear-interval dimension of a partial order \( P \) is the lexicographically smallest pair \( (k, \ell) \) such that \( P = \bigcap_{i=1}^{k} P_i \), where \( \{P_i\}_{i=1}^{k} \) are interval orders and exactly \( \ell \) among them are not linear orders. In particular, \( P \) is a linear-interval order if its linear-interval dimension is at most \((2, 1)\), i.e. \( P = P_1 \cap P_2 \), where \( P_1 \) is a linear order and \( P_2 \) is an interval order.
3.2 Threshold graphs and alternating cycles

In this section we provide preliminary definitions and known results on alternating cycles and on threshold graphs, which will be useful for the remainder of the paper.

**Definition 3.** Let $G = (V, E)$ be a graph, $\tilde{E} \subseteq E$ be an edge subset, and $k \geq 2$. A set of $2k$ (not necessarily distinct) vertices $v_1, v_2, \ldots, v_{2k} \in V$ builds an alternating cycle $AC_{2k}$ in $\tilde{E}$, if $v_iv_{i+1} \in \tilde{E}$ whenever $i$ is even and $v_iv_{i+1} \notin \tilde{E}$ whenever $i$ is odd (where indices are mod $2k$). Furthermore, we say that $G$ has an alternating cycle $AC_{2k}$, whenever $G$ has an $AC_{2k}$ in the edge set $\tilde{E} = E$.

For instance, for $k = 3$, there exist two different possibilities for an $AC_6$, which are illustrated in Figures 1(a) and 1(b). These two types of an $AC_6$ are called an alternating path of length 5 or of length 6, respectively ($AP_5$ and $AP_6$ for short, respectively). In an $AP_5$ on vertices $v_1, v_2, v_3, v_4, v_5$, if there exist the edges $v_1v_3$ and $v_2v_6$ (or, symmetrically, the edges $v_3v_5$ and $v_4v_2$, or the edges $v_5v_1$ and $v_6v_4$), then this $AP_5$ is called a double $AP_6$, cf. Figure 1(c).

**Definition 4.** Let $G = (V, E)$ be a graph and $v_1, \ldots, v_6$ be the vertices of an $AP_6$. Then the non-edge $v_1v_2$ (resp. the non-edge $v_3v_4, v_5v_6$) is a base of the $AP_6$ and the edge $v_4v_5$ (resp. the edge $v_6v_1$, $v_2v_3$) is the corresponding ceiling of this $AP_6$.

![Fig. 1. All possibilities for an $AC_6$: (a) an alternating path $AP_5$ of length 5, (b) an alternating path $AP_6$ of length 6, and (c) a double $AP_6$. The solid lines denote edges of the graph and the dashed lines denote non-edges of the graph.](image)

Furthermore, note that for $k = 2$, a set of four vertices $v_1, v_2, v_3, v_4 \in V$ builds an alternating cycle $AC_4$ if $v_1v_2, v_3v_4 \in E$ and $v_1v_4, v_2v_3 \notin E$. There are three possible graphs on four vertices that build an alternating cycle $AC_4$, namely $2K_2$, $P_4$, and $C_4$, which are illustrated in Figure 2.

![Fig. 2. The three possible $AC_4$'s: (a) a $2K_2$, (b) a $P_4$, and (c) a $C_4$.](image)

Alternating cycles can be used to characterize threshold and chain graphs. In particular, threshold graphs are the graphs with no induced $AC_4$, and chain graphs are the bipartite graphs with no induced $2K_2$ [16].

We define now for any bipartite graph $G$ the associated split graph of $G$, which we will use extensively in the remainder of the paper.

**Definition 5.** Let $G = (U, V, E)$ be a bipartite graph. The associated split graph of $G$ is the split graph $H_G = (U \cup V, E')$, where $E' = E \cup (V \times V)$, i.e. $H_G$ is the split graph made by $G$ by replacing the independent set $V$ of $G$ by a clique.

**Observation 1** Let $G$ be a bipartite graph and $H_G$ be the associated split graph of $G$. Then, $G$ has an induced $2K_2$ if and only if $H_G$ has an induced $AC_4$, and in this case this $AC_4$ is a $P_4$. 

The next lemma connects the chain cover number $ch(G)$ of a bipartite graph $G$ with the threshold cover number $t(H_G)$ of the associated split graph $H_G$ of $G$. Recall that the problem of deciding whether a graph $G$ has threshold cover number at most a given number $k$ is NP-complete for $k \geq 3$ \cite{23}, while it is polynomial for $k = 2$ \cite{21}.

**Lemma 1** \cite{16}. Let $G = (U, V, E)$ be a bipartite graph. Then $chi(G) = t(H_G)$.

The next two definitions of a *conflict* between two edges and the *conflict graph* are essential for our results.

**Definition 6.** Let $G = (V, E)$ be a graph and $e_1, e_2 \in E$. If the vertices of $e_1$ and $e_2$ build an $AC_2$ in $G$, then $e_1$ and $e_2$ are in conflict, and in this case we denote $e_1 \parallel e_2$ in $G$. Furthermore, an edge $e \in E$ is committed if there exists an edge $e' \in E$ such that $e \parallel e'$; otherwise $e$ is uncommitted.

**Definition 7** \cite{21}. Let $G = (V, E)$ be a graph. The conflict graph $G^* = (V^*, E^*)$ of $G$ is defined by

- $V^* = E$ and
- for every $e_1, e_2 \in E$, $e_1e_2 \in E^*$ if and only if $e_1 \parallel e_2$ in $G$.

**Observation 2** Let $G = (V, E)$ be a graph and let $e \in E$. If $e$ is uncommitted, then $e$ is an isolated vertex in the conflict graph $G^*$ of $G$.

**Observation 3** Let $G = (V, E)$ be a split graph. Let $K$ and $I$ be a partition of $V$, such that $K$ is a clique and $I$ is an independent set (such a partition always exists for split graphs). Then, every edge of $K$ is uncommitted.

**Lemma 2.** Let $G$ be a graph and let the vertices $v_1, \ldots, v_6$ of $G$ build an $AP_6$ (an alternating path of length 6). Assume that among the three edges $\{v_2v_3, v_4v_5, v_6v_1\}$ of this $AP_6$, no pair of edges is in conflict. Then the edges $v_3v_6, v_4v_1, v_5v_2$ exist in $G$ and $v_4v_5 \parallel v_3v_6, v_2v_3 \parallel v_1v_4, v_6v_1 \parallel v_5v_2$.

**Proof.** Suppose that $v_3v_6$ is not an edge of $G$. Then the edges $v_2v_3$ and $v_6v_1$ are in conflict, since $v_1v_2$ is not an edge of $G$ (cf. Figure 1(b)), which is a contradiction to the assumption of the lemma. Therefore $v_3v_6$ is an edge of $G$. By symmetry, it follows that also $v_4v_1$ and $v_5v_2$ are edges in $G$. Note now that the edges $v_1v_4$ and $v_5v_6$ are not edges of $G$. By symmetry, it follows that also $v_2v_3 \parallel v_1v_4, v_6v_1 \parallel v_5v_2.$

Note that the threshold cover number $t(G)$ of a graph $G = (V, E)$ equals the smallest $k$, such that the edge set $E$ of $G$ can be partitioned into $k$ sets $E_1, E_2, \ldots, E_k$, each having a threshold completion in $G$ (that is, there exists for every $i = 1, 2, \ldots, k$ an edge set $E'_i$, such that $E_i \subseteq E'_i \subseteq E$ and $(V, E'_i)$ is a threshold graph). The following characterization of subgraphs that admit a threshold completion in a given graph $G$ has been proved in \cite{12}.

**Lemma 3** \cite{12}. Let $H$ be a subgraph of a graph $G = (V, E)$. Then $H$ has a threshold completion in $G$ if and only if $G$ has no $AC_2k$, $k \geq 2$, on the edges of $H$.

If the conditions of Lemma 3 are satisfied, then such a threshold completion of $H$ in $G$ can be computed in linear time, as the next lemma states.

**Lemma 4** \cite{21}. If a subgraph $H$ of $G = (V, E)$ has a threshold completion in $G$, then it can be computed in $O(|V| + |E|)$ time.

**Corollary 1.** Let $G = (V, E)$ be a graph. Then, $t(G) = 1$ if and only if $G$ has no $AC_2k$, $k \geq 2$. Furthermore, $t(G) \leq 2$ if and only if the set $E$ of edges can be partitioned into two sets $E_1$ and $E_2$, such that $G$ has no $AC_2k$, $k \geq 2$, in each $E_i$, $i = 1, 2$.

**Proof.** First note that $t(G) = 1$ if and only if $G$ is a threshold graph. Therefore, Lemma 3 implies that $t(G) = 1$ if and only if $G$ has no $AC_2k$, $k \geq 2$.

Recall that the threshold cover number $t(G)$ of a graph $G = (V, E)$ equals the smallest $k$, such that the edge set $E$ of $G$ can be partitioned into $k$ sets $E_1, E_2, \ldots, E_k$, each having a threshold completion in $G$. Therefore, if $t(G) \leq 2$, Lemma 3 implies that $E$ can be partitioned into two such sets $E_1$ and $E_2$, such that $G$ has no $AC_2k$, $k \geq 2$, in each $E_i$, $i = 1, 2$. Note that, in the case where $t(G) = 1$ (i.e. $G$ is a threshold graph), we can set $E_1 = E$ and $E_2 = \emptyset$. Conversely, suppose that $E$ can be partitioned into two such sets $E_1$ and $E_2$. Then Lemma 3 implies that both graphs $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$ have a threshold completion in $G$, where $G_1 \cup G_2 = G$. Therefore $t(G) \leq 2$. \qed
For every graph $G$, it can be easily proved that the chromatic number $\chi(G^*)$ of its conflict graph $G^*$ provides a lower bound for the threshold cover number $t(G)$ of $G$, as the text lemma states.

**Lemma 5** ([16]). Let $G$ be a graph. Then $\chi(G^*) \leq t(G)$.

Lemma 5 immediately implies that a necessary condition for a graph $G$ to have threshold cover number $t(G) \leq 2$ is that $\chi(G^*) \leq 2$, i.e. that $G^*$ is a bipartite graph. The main result of [21] is the next theorem, which proves that this is also a sufficient condition for graphs $G$ with $\chi(G^*) \leq 2$.

**Theorem 2** ([21]). If the conflict graph $G^*$ of a graph $G = (V, E)$ is bipartite (i.e. $\chi(G^*) \leq 2$), then $t(G) \leq 2$. Moreover, $E$ can be partitioned in $O(|E|(|V| + |E|))$ time into two sets $E_1$ and $E_2$, such that $G$ has no $AC_{2k}$, $k \geq 2$, in each $E_i$, $i = 1, 2$.

Due to the next theorem, it suffices for bipartite conflict graphs $G^*$ to consider only small alternating cycles $AC_{2k}$ with $k \leq 3$.

**Theorem 3** ([12]). Suppose that the conflict graph $G^*$ of a graph $G = (V, E)$ is bipartite (i.e. $\chi(G^*) \leq 2$), with (vertex) color classes $E_1$ and $E_2$. If $G$ has an $AC_{2k}$ on the edges of $E_1$ (resp. of $E_2$), where $k \geq 3$, then $G$ has also an $AC_6$ in $E_1$ (resp. of $E_2$).

**Lemma 6** ([13]). Let $G = (V, E)$ be a split graph. Let $K$ and $I$ be a partition of $V$ such that $K$ induces a clique and $I$ induces an independent set in $G$. Assume that the vertices $v_1, \ldots, v_6$ build an $AP_6$ in $G$. Then either $v_1, v_3, v_5 \in K$ and $v_2, v_4, v_6 \in I$, or $v_1, v_3, v_5 \in I$ and $v_2, v_4, v_6 \in K$.

**Lemma 7.** Any split graph $G$ does not contain any $AP_5$ or any double $AP_6$.

*Proof.* The fact that a split graph $G$ does not contain any $AP_5$ has been proved in [13]. Let now $K$ and $I$ be a partition of the vertices $V$ of $G$, such that $K$ induces a clique and $I$ induces an independent set in $G$ (such a partition exists by definition, since $G$ is a split graph). Suppose that $G$ has an $AP_6$ on the vertices $v_1, v_2, v_3, v_4, v_5, v_6$, cf. Figure 1(a). Then, Lemma 6 implies that either $v_1, v_3, v_5 \in K$ and $v_2, v_4, v_6 \in I$, or $v_1, v_3, v_5 \in I$ and $v_2, v_4, v_6 \in K$. In both cases, none of the pairs of edges $\{v_1v_3, v_2v_6\}$, $\{v_1v_5, v_4v_2\}$, and $\{v_5v_1, v_6v_4\}$ can exist simultaneously in $G$. Therefore, $G$ has no double $AP_6$. This completes the proof of the lemma.

4 Linear-Interval covers of bipartite graphs

In this section we introduce the crucial notion of a linear-interval cover of bipartite graphs (cf. Definition 10). Then we use linear-interval covers to provide a new characterization of PI graphs (cf. Theorem 9), which is one of the main tools for our PI graph recognition algorithm. First we provide in the next theorem the characterization of PI graphs using linear orders and interval orders.

**Theorem 4.** Let $G = (V, E)$ be a comparability graph and $P$ be a partial order of $G$. Then $G$ is a PI graph if and only if $P = P_1 \cap P_2$, where $P_1$ is a linear order and $P_2$ is an interval order.

*Proof.* For the purposes of the proof, a partial order $P = (U, R)$ is called a PI order [5], if there exists a PI representation (i.e. a simple-triangle representation) $R$, such that for any two $u, v \in U$, $u \prec_P v$ if and only if the triangle associated to $u$ lies in $R$ entirely to the left of the triangle associated to $v$.

Suppose that $P = P_1 \cap P_2$ for two partial orders $P_1$ and $P_2$, where $P_1$ is a linear order and $P_2$ is an interval order. Then $P$ is a PI order [5], and thus $G$ is a PI graph. Conversely, suppose that $G$ is a PI graph. Equivalently, $P$ is a PI order, and thus $\text{lidim}(P) \leq (2, 1)$ [5]. That is, $P = P_1 \cap P_2$ for two partial orders $P_1$ and $P_2$, where $P_1$ is a linear order and $P_2$ is an interval order. Moreover, whenever we are given a partial order $P$ such that $P = P_1 \cap P_2$, where $P_1$ is a linear order and $P_2$ is an interval order, it is straightforward to compute a PI model for $P$ (cf. [5]). Equivalently, we can easily construct in this case a PI representation of the incomparability graph $G$ of $P$ (cf. lines 13-15 of Algorithm 1 below).

For every partial order $P$ we define now the domination bipartite graph $C(P)$, which has been used to characterize interval orders [15]. Here “$<$” stands for “Comparable”, since the definition of $C(P)$ uses the comparable elements of $P$.

**Definition 8** ([15]). Let $P = (U, R)$ be a partial order, where $U = \{u_1, u_2, \ldots, u_n\}$. Furthermore let $V = \{v_1, v_2, \ldots, v_n\}$. The domination bipartite graph $C(P) = (U, V, E)$ is defined such that $u_i v_j \in E$ if and only if $u_i \prec_P u_j$. 

8
Lemma 8 ([15]). Let $P = (U, R)$ be a partial order. Then, $P$ is an interval order if and only if $C(P)$ is a chain graph.

Extending the notion of $C(P)$, we now introduce the bipartite graph $NC(P)$ to characterize linear orders (cf. Lemma 9). Here “NC” stands for “Non-strictly Comparable”. Namely, this graph can be obtained by adding to the graph $C(P)$ the perfect matching $\{u_i v_i \mid i = 1, 2, \ldots, n\}$ on the vertices of $U$ and $V$.

Definition 9. Let $P = (U, R)$ be a partial order, where $U = \{u_1, u_2, \ldots, u_n\}$. Furthermore let $V = \{v_1, v_2, \ldots, v_n\}$. Then, $NC(P) = (U, V, E)$ is the bipartite graph, such that $u_i v_j \in E$ if and only if $u_i \leq_P u_j$.

Lemma 9. Let $P = (U, R)$ be a partial order. Then, $P$ is a linear order if and only if $NC(P)$ is a chain graph.

Proof. Let $U = \{u_1, u_2, \ldots, u_n\}$. Suppose that $P$ is a linear order, i.e. $u_1 <_P u_2 <_P \ldots <_P u_n$. Then, by Definition 9, the set of neighbors of a vertex $u_i \in U$ in the graph $NC(P)$ is $N(u_i) = \{v_i, v_{i+1}, \ldots, v_n\}$. Therefore, $N(u_n) \subseteq N(u_{n-1}) \subseteq \ldots \subseteq N(u_1)$, and thus $NC(P)$ is a chain graph.

Suppose now that $NC(P)$ is a chain graph. Then the sets of neighbors of the vertices of $U$ in the graph $NC(P)$ can be linearly ordered by inclusion. Let without loss of generality $N(u_1) \subseteq N(u_2) \subseteq \ldots \subseteq N(u_n)$. Therefore, since $v_i \in N(u_i)$ in $NC(P)$ for every $i = 1, 2, \ldots, n$, it follows that $v_i \in N(u_j)$ in $NC(P)$ whenever $i < j$. Therefore, by Definition 9, $u_j <_P u_i$ whenever $i < j$. That is, $u_n <_P u_{n-1} <_P \ldots <_P u_1$, i.e. $P$ is a linear order.

We introduce now the notion of a linear-interval cover of a bipartite graph. This notion is crucial for our main result of this section, cf. Theorem 5.

Definition 10. Let $G = (U, V, E)$ be a bipartite graph, where $U = \{u_1, u_2, \ldots, u_n\}$ and $V = \{v_1, v_2, \ldots, v_n\}$. Let $E_0 = \{u_i v_i \mid 1 \leq i \leq n\}$ and suppose that $E_0 \subseteq E$. Then, $G$ is linear-interval coverable if there exist two chain graphs $G_1 = (U, E_1)$ and $G_2 = (U, E_2)$, such that $G = G_1 \cup G_2$ and $E_0 \subseteq E_2 \setminus E_1$. In this case, the sets $\{E_1, E_2\}$ are a linear-interval cover of $G$.

Before we proceed with Theorem 5, we first provide the next auxiliary lemma.

Lemma 10. Let $Q_1 = (U, R_1)$ be an interval order and $Q_2 = (U, R_2)$ be a partial order, such that $Q_1$ and $Q_2$ do not contradict each other. Then there exists a linear order $Q_0$ that is a linear extension of both $Q_1$ and $Q_2$.

Proof. Let $U = \{u_1, u_2, \ldots, u_n\}$ be the ground set of $Q_1$ and $Q_2$. Furthermore let $C(Q_1) = (U, V, E_1)$ be the domination bipartite graph of $Q_1$, where $V = \{v_1, v_2, \ldots, v_n\}$, cf. Definition 8. Since $Q_1$ is an interval order by assumption, $C(Q_1)$ is a chain graph by Lemma 8, i.e. $C(Q_1)$ does not contain an induced $2K_2$. Consider now two edges $u_i v_j$ and $u_k v_l$ of $C(Q_1)$, where $\{i, j\} \cap \{k, \ell\} = \emptyset$. Then $u_i <_{Q_1} u_j$ and $u_k <_{Q_1} u_\ell$ by Definition 8. Furthermore, at least one of the edges $u_i v_j$ and $u_k v_\ell$ exists in $C(Q_1)$, since otherwise the edges $u_i v_j$ and $u_k v_\ell$ induce a $2K_2$ in $C(Q_1)$, which is a contradiction. Therefore $u_i <_{Q_1} u_j$ or $u_k <_{Q_1} u_\ell$. Since $Q_1$ and $Q_2$ do not contradict each other by assumption, we can define the simple directed graph $G_0 = (U, E)$, such that $\overrightarrow{u_i u_j} \in E$ if and only if $u_i <_{Q_1} u_j$, or $u_i <_{Q_2} u_j$. We will prove that $G_0$ is acyclic. Suppose otherwise that $G_0$ has at least one directed cycle, and let $C$ be a directed cycle of $G_0$ with the smallest possible length. Assume first that $C$ has length 3, and let its edges be $\overrightarrow{u_i u_j}, \overrightarrow{u_j u_k}$, and $\overrightarrow{u_k u_i}$. Then at least two of these edges belong to $Q_1$ or to $Q_2$. Let without loss of generality $\overrightarrow{u_i u_j}$ and $\overrightarrow{u_j u_k}$ belong to $Q_1$, i.e. $u_i <_{Q_1} u_j$ and $u_j <_{Q_1} u_k$. Then also $u_i <_{Q_1} u_k$, since $Q_1$ is transitive, and thus $\overrightarrow{u_i u_k} \notin E$. This contradicts the assumption that $\overrightarrow{u_i u_k}$ is an edge of $C$. Assume now that $C$ has length greater than 3. Suppose that two consecutive edges $\overrightarrow{u_i u_j}$ and $\overrightarrow{u_j u_k}$ of $C$ belong to $Q_1$, i.e. $u_i <_{Q_1} u_j$ and $u_j <_{Q_1} u_k$. Then also $u_i <_{Q_1} u_k$, since $Q_1$ is transitive, and thus $\overrightarrow{u_i u_k} \notin E$. Therefore we can replace in $C$ the edges $\overrightarrow{u_i u_j}$ and $\overrightarrow{u_j u_k}$ by the edge $\overrightarrow{u_i u_k}$, obtaining thus a smaller directed cycle than $C$, which is a contradiction by the assumption on $C$. Thus no two consecutive edges of $C$ belong to $Q_1$. Similarly, no two consecutive edges of $C$ belong to $Q_2$, and thus the edges of $C$ belong alternately to $Q_1$ and $Q_2$. In particular, $C$ has even length.

Consider now three consecutive edges $\overrightarrow{u_i u_j}, \overrightarrow{u_j u_k}, \overrightarrow{u_k u_i}$ of $C$, where $\overrightarrow{u_i u_j}$ and $\overrightarrow{u_k u_i}$ belong to $Q_1$, then $u_i <_{Q_1} u_j$ and $u_k <_{Q_1} u_i$, where $\{i, j\} \cap \{k, \ell\} = \emptyset$, and thus $u_i <_{Q_1} u_j$ or $u_k <_{Q_1} u_i$, as we proved above. That is, $\overrightarrow{u_i u_j} \in E$ or $\overrightarrow{u_k u_i} \in E$. Therefore, since we assumed that $\overrightarrow{u_k u_i}$ is an edge of $C$, it follows that $\overrightarrow{u_i u_j} \notin E$, and thus $\overrightarrow{u_i u_j} \in E$. Therefore, in particular, $\overrightarrow{u_i u_j} \notin E$, and thus $C$ does not have length 4, i.e. it has length at least 6. Thus we can replace in $C$ the edges $\overrightarrow{u_i u_j}, \overrightarrow{u_j u_k}, \overrightarrow{u_k u_i}$ by the edge $\overrightarrow{u_i u_k}$, obtaining thus a smaller directed cycle than $C$, which is a contradiction by the assumption on $C$. 

9
Proof. (a) ⇒ (b). Since $P_1$ is a linear order, it follows by Lemma 8 that $NC(P_1)$ is a chain graph. Furthermore, since $P_2$ is an interval order, it follows by Lemma 8 that $C(P_2)$ is a chain graph. Therefore, since the class of chain graphs is closed under bipartite complementation, it follows that $\overline{NC}(P_1)$ and $C(P_2)$ are chain graphs.

Let $u_1,u_j \in U$ such that $u_1v_j E(C(P))$. Then $u_1 <_{P_1} u_j$ by Definition 8. Furthermore, since $P = P_1 \cap P_2$ by assumption, it follows that $u_1 <_{P_1} u_j$ and $u_1 <_{P_2} u_j$, and thus also $u_1v_j \in E(\overline{NC}(P_1))$ and $u_1v_j \in E(C(P_2))$ by Definitions 8 and 9 respectively. Therefore $C(P) \subseteq NC(P_1) \cap C(P_2)$.

Let now $u_1,u_j \in U$ such that $u_1v_j \in E(\overline{NC}(P_1))$ and $u_1v_j \in E(C(P_2))$. Then, it follows in particular that $u_1 \neq u_j$. Suppose otherwise $u_1v_j \notin E(C(P_2))$, a contradiction. Thus, $u_1 <_{P_1} u_j$ and $u_1 <_{P_2} u_j$ by Definitions 8 and 9. Therefore, since $P = P_1 \cap P_2$ by assumption, it follows that $u_1 <_{P_1} u_j$ and thus $u_1v_j \in E(C(P))$ by Definition 8. That is, $\overline{NC}(P_1) \cap C(P_2) \subseteq C(P)$. Summarizing, $C(P) = \overline{NC}(P_1) \cap C(P_2)$, and thus also $\overline{C}(P) = \overline{NC}(P_1) \cup \overline{C}(P_2)$.

(b) ⇒ (a). Since $\overline{C}(P) = \overline{NC}(P_1) \cup \overline{C}(P_2)$, it follows that $C(P) = NC(P_1) \cap C(P_2)$. Let $u_1,u_j \in U$ such that $u_1 <_{P_1} u_j$. Then $u_1v_j \in E(C(P))$ by Definition 8. Therefore, since $C(P) = NC(P_1) \cap C(P_2)$, it follows that also $u_1v_j \in E(\overline{NC}(P_1))$ and $u_1v_j \in E(C(P_2))$. Thus, in particular, $u_1 \neq u_j$. Suppose otherwise $u_1v_j \notin E(C(P_2))$, a contradiction. Therefore $u_1 <_{P_1} u_j$ and $u_1 <_{P_2} u_j$ by Definitions 8 and 9. That is, $P \subseteq P_1 \cap P_2$.

Let now $u_1,u_j \in U$ such that $u_1 <_{P_1} u_j$ and $u_1 <_{P_2} u_j$. Then $u_1v_j \in E(\overline{NC}(P_1))$ and $u_1v_j \in E(C(P_2))$ by Definitions 8 and 9. Therefore, since $C(P) = NC(P_1) \cap C(P_2)$, it follows that also $u_1v_j \in E(C(P_2))$. Thus $u_1 <_{P_1} u_j$ by Definition 8. That is, $P_1 \cap P_2 \subseteq P$. Summarizing, $P = P_1 \cap P_2$. Furthermore, since by assumption $\overline{NC}(P_1)$ and $\overline{C}(P_2)$ are chain graphs, it follows that also $NC(P_1)$ and $C(P_2)$ are chain graphs. Therefore $P_1$ is a linear order and $P_2$ is an interval order by Lemmas 8 and 9 respectively.

(c) ⇒ (b). Define $G_1 = \overline{NC}(P_1)$ and $G_2 = \overline{C}(P_2)$. Then, it follows by (b) that $G_1$ and $G_2$ are chain graphs and that $\overline{C}(P) = G_1 \cup G_2$. Note now by Definitions 8 and 9 that $E_0 \cap E(C(P)) = \emptyset$ and that $E_0 \subseteq E(\overline{NC}(P_1))$, respectively. Therefore $E_0 \subseteq E(\overline{C}(P_2)) \subseteq E(\overline{NC}(P_1))$. Thus, since $E_2 = E(G_2) = E(C(P_2))$ and $E_1 = E(G_1) = E(\overline{NC}(P_1))$, it follows that $E_0 \subseteq E_2 \setminus E_1$. That is, $\overline{C}(P)$ is linear-interval coverable by Definition 10.

(c) ⇒ (b). We will construct from the edge sets $E_1$ and $E_2$ of $G_1$ and $G_2$, respectively, a linear order $P_1$ and an interval order $P_2$, such that $\overline{C}(P) = \overline{NC}(P_1) \cup \overline{C}(P_2)$. Note first the bipartite complement $\overline{G}_2$ of $G_2$ as $\overline{G}_2 = (U,V,\overline{E}_2)$. Note that $\overline{G}_2$ is a chain graph, since $G_2$ is also a chain graph by assumption.

The interval order $P_2$. We define $P_2$, such that $u_1 <_{P_2} u_j$, if and only if $u_1v_j \in \overline{E}_2$. We will now prove that $P_2$ is a partial order. Recall that $E_0 \subseteq E_2$ by assumption, and thus $E_0 \cap \overline{E}_2 = \emptyset$. That is, $u_1v_j \notin \overline{E}_2$ for every $i = 1,2,\ldots,n$. Furthermore, $\overline{G}_2$ is a chain graph, since $G_2$ is a chain graph by assumption. Therefore, for two distinct indices $i,j$, at most one of the edges $u_iv_j$ and $u_jv_i$ belongs to $\overline{E}_2$, since otherwise these two edges would induce a $2K_2$ in $\overline{G}_2$, which is a contradiction. Thus, according to our definition of $P_2$, whenever $i \neq j$, it holds that either $u_i <_{P_2} u_j$, or $u_j <_{P_2} u_i$, or $u_i$ and $u_j$ are incomparable in $P_2$. Suppose that $u_1 <_{P_2} u_j$ and $u_j <_{P_2} u_k$ for three indices $i,j,k$. That is, $u_jv_k \notin \overline{E}_2$ by definition of $P_2$. Since $\overline{G}_2 = (U,V,\overline{E}_2)$ is a chain graph, the edges $u_jv_k$ and $u_kv_j$ do not build a $2K_2$ in $\overline{G}_2$. Therefore, since $u_jv_k \notin \overline{E}_2$, it follows that $u_jv_k \in \overline{E}_2$, i.e. $u_i <_{P_2} u_k$. That is, $P_2$ is transitive, and thus $P_2$ is a partial order. Furthermore, note by the definition of $P_2$ and by Definition 8 that $\overline{G}_2 = C(P_2)$. Therefore, since $\overline{G}_2$ is a chain graph, it follows by Lemma 8 that $P_2$ is an interval order.

In order to define the linear order $P_1$, we first define two auxiliary orders $Q_1$ and $Q_2$, as follows.
The interval order $Q_1$. We define $Q_1$, such that $u_i <_{Q_1} u_j$ if and only if $u_i v_j \in E_1$. We will prove that $Q_1$ is a partial order. Recall that $E_0 \cap E_1 = \emptyset$ by assumption. That is, $u_i v_j \notin E_1$ for every $i = 1, 2, \ldots, n$. Furthermore, for two distinct indices $i, j$, at most one of the edges $u_i v_j$ and $u_j v_i$ belongs to $E_1$. Indeed, otherwise these two edges would induce a $2K_2$ in $G_1$, which is a contradiction since $G_1$ is a chain graph by assumption. Thus, according to our definition of $Q_1$, whenever $i \neq j$, it follows that either $u_i <_{Q_1} u_j$, or $u_j <_{Q_1} u_i$, or $u_i$ and $u_j$ are incomparable in $Q_1$. Suppose that $u_i <_{Q_1} u_j$ and $u_j <_{Q_1} u_k$ for three indices $i, j, k$. That is, $u_i v_j, u_j v_k \in E_1$ by definition of $Q_1$. Since $G_1$ is a chain graph by assumption, the edges $u_i v_j$ and $u_j v_k$ do not build a $2K_2$ in $G_1$. Therefore, since $u_i v_j \notin E_1$, it follows that $u_i v_k \notin E_1$, i.e. $u_i <_{Q_1} u_k$. That is, $Q_1$ is transitive, and thus $Q_1$ is a partial order. Furthermore, note by the definition of $Q_1$ and by Definition 8 that $G_1 = C(Q_1)$. Therefore $Q_1$ is an interval order by Lemma 8, since $G_1$ is a chain graph by assumption.

The partial order $Q_2$. We define the partial order $Q_2$ as the inverse partial order $P$ of $P$. That is, $u_i <_{Q_2} u_j$ if and only if $u_j <_{Q_1} u_i$. Note that $Q_2$ is transitive, since $P$ is transitive.

Before we define the linear order $P_1$, we first prove that the partial orders $Q_1$ and $Q_2$ do not contradict each other. Suppose otherwise that $u_i <_{Q_1} u_j$ and $u_j <_{Q_2} u_i$, for some pair $u_i, u_j$. Then, since $u_i <_{Q_1} u_j$, it follows that $u_i v_j \in E_1$ by definition of $Q_1$. Therefore $u_i v_j \in E(\hat{C}(P))$, since $\hat{C}(P) = G_1 \cup G_2$ by assumption. On the other hand, since $u_i <_{Q_2} u_j$, it follows that $u_j < P u_i$ by definition of $Q_2$. Therefore $u_i v_j \in E(\hat{C}(P))$ by Definition 8, and thus $u_i v_j \notin E(\hat{C}(P))$, which contradicts that $\hat{C}(P) = G_1 \cup G_2$ by assumption. Furthermore recall that $G_2 = C(P_2)$ as we proved above, and thus $G_2 = \hat{C}(P_2)$. Therefore, since $G_1 \leq \hat{NC}(P_1)$, it follows that $\hat{C}(P) \leq \hat{NC}(P_1) \cup \hat{C}(P_2)$.

Finally we prove that $\hat{C}(P) \leq NC(P_1) \cap C(P_2)$. Consider now an edge $u_i v_j \in E(\hat{C}(P))$. Then $u_i < P u_j$ by Definition 8, and thus $u_j <_{Q_2} u_i$ by the definition of $Q_2$. Furthermore $u_i <_{P_1} u_j$ by the definition of $P_1$, and thus $u_i v_j \in E(\hat{NC}(P_1))$ by Definition 8. Note now that $\hat{C}(P) = \hat{G}_1 \cap \hat{G}_2$, since $\hat{C}(P) = G_1 \cup G_2$ by assumption. Therefore, since $u_i v_j \in E(\hat{C}(P))$ by assumption, it follows that $u_i v_j \in \hat{E}_2$. That is, if $u_i v_j \in E(\hat{C}(P))$ then $u_i v_j \in E(\hat{NC}(P_1))$ and $u_i v_j \in \hat{E}_2$. Therefore, since $\hat{G}_2 = C(P_2)$, it follows that $\hat{C}(P) \leq NC(P_1) \cap C(P_2)$.

Summarizing, since $\hat{C}(P) \leq \hat{NC}(P_1) \cup \hat{C}(P_2)$ and $C(P) \leq NC(P_1) \cap C(P_2)$, it follows that $\hat{C}(P) = NC(P_1) \cup C(P_2)$. This completes the proof of the theorem. □

The next corollary follows now easily by Theorems 4 and 5.

Corollary 2. Let $G = (V, E)$ be a cocomparability graph and $P$ be a partial order of $\overline{G}$. Then, $G$ is a PI graph if and only if the bipartite graph $\hat{C}(P)$ is linear-interval coverable.

We now present Algorithm 1, which constructs a PI representation $R$ of a cocomparability graph $G$ by a linear-interval cover $\{E_1, E_2\}$ of the bipartite graph $\hat{C}(P)$ (cf. Definition 10). Since $E_0 \subseteq E_2 \setminus E_1$ by Definition 10, where $E_0 = \{u_i v_j \mid 1 \leq i \leq n\}$ and $n$ is the number of vertices of $G$, note that $i \neq j$ during the execution of each of the lines 6, 8, and 10 of Algorithm 1.

Theorem 6. Let $G$ be a cocomparability graph with $n$ vertices and $P$ be the partial order of $\overline{G}$. Let $\{E_1, E_2\}$ be a linear-interval cover of $\hat{C}(P)$. Then Algorithm 1 constructs in $O(n^2)$ time a PI representation $R$ of $G$.

Proof. Since $\hat{C}(P)$ admits a linear-interval cover $\{E_1, E_2\}$, Corollary 2 implies that $G$ is a PI graph. Furthermore, it follows by the proof of the implication $(c) \Rightarrow (b)$ in Theorem 3 that the partial orders $P_1$ and $P_2$ are constructed in lines 12, 13, 12 of Algorithm 1 are a linear order and an interval order, respectively, such that $\hat{C}(P) = \hat{NC}(P_1) \cup \hat{C}(P_2)$. Furthermore, it follows by the proof of the implication $(b) \Rightarrow (a)$ in Theorem 5 that $P = P_1 \cap P_2$ for these two partial orders. Since we have computed in lines 12, 13, 12 the linear order $P_1$ and the interval order $P_2$, for which $P = P_1 \cap P_2$, it is now straightforward to construct a PI
Algorithm 1 Construction of a PI representation, given a linear-interval cover

Input: A cocomparability graph $G$, a partial order $P$ of $\overline{C}$, the domination bipartite graph $C(P) = (U, V, E)$, and a linear-interval cover $(E_1, E_2)$ of $\overline{C}(P)$

Output: A PI representation $R$ of $G$

1: Let $U = \{u_1, u_2, \ldots, u_n\}$, $V = \{v_1, v_2, \ldots, v_n\}$
2: $Q_1 \leftarrow \emptyset$, $Q_2 \leftarrow \emptyset$, $P_2 \leftarrow \emptyset$
3: for $i = 1, 2, \ldots, n$ do (construction of the partial orders $Q_1, Q_2, P_2$)
   4: for $j = 1, 2, \ldots, n$ do
   5: if $u_i v_j \notin E_2$ then $\{i \neq j\}$
   6: $u_i <_{P_2} u_j$
   7: if $u_i v_j \in E_1$ then $\{i \neq j\}$
   8: $u_i <_{Q_2} u_j$
   9: if $u_i <_{P} v_j$ then $\{i \neq j\}$
   10: $u_i <_{Q_2} u_j$
11: Compute a linear extension $Q_0$ of $Q_1 \cup Q_2$
12: $P_1 \leftarrow Q_0$
13: Place the elements of $U$ on a line $L_1$ according to the linear order $P_1$
14: Place a set of $n$ intervals on a line $L_2$ (parallel to $L_1$) according to the interval order $P_2$
15: Build the PI representation $R$ of $G$ by connecting the endpoints of the intervals on $L_2$ with the corresponding points on $L_1$
16: return $R$

representation $R$ of $G$ as follows (cf. also \cite{5} and the proof of Theorem 4). We arrange a set of $n$ points (resp. $n$ intervals) on a line $L_1$ (resp. on a line $L_2$, parallel to $L_1$) according to the linear order $P_1$ (resp. to the interval order $P_2$). Then we connect the endpoints of the intervals on $L_2$ with the corresponding points on $L_1$. Regarding the time complexity, each of the lines 5-10 of Algorithm 1 can be executed in constant time, and thus the lines $[3, 10]$ can be executed in total $O(n^2)$ time. Furthermore, since the lines $[11, 15]$ can be executed in a trivial way in at most $O(n^2)$ time each, it follows that the running time of Algorithm 1 is $O(n^2)$.

5 Detecting linear-interval covers using Boolean satisfiability

The natural algorithmic question that arises from the characterization of PI graphs using linear-interval covers in Corollary \cite{3} is the following: “Given a cocomparability graph $G$ and a partial order $P$ of $\overline{G}$, can we efficiently decide whether the bipartite graph $\overline{C}(P)$ has a linear-interval cover?” We will answer this algorithmic question in the affirmative in Section 9. In this section we translate every instance of this decision problem (i.e. whether the bipartite graph $\overline{C}(P)$ has a linear-interval cover) to a restricted instance of 3SAT (cf. Theorem \cite{7}). That is, for every such a bipartite graph $\overline{C}(P)$, we construct a Boolean formula $\phi$ in conjunctive normal form (CNF), with size polynomial on the size of $\overline{C}(P)$ (and thus also on $G$), such that $\overline{C}(P)$ has a linear-interval cover if and only if $\phi$ is satisfiable. In particular, this formula $\phi$ can be written as $\phi = \phi_1 \wedge \phi_2$, where $\phi_1$ has three literals in every clause and $\phi_2$ has two literals in every clause. Moreover, as we will prove in Section 6, the satisfiability problem can be efficiently decided on the formula $\phi$, by exploiting an appropriate sub-formula of $\phi$ which is gradually mixed (cf. Definition \cite{2}).

In the remainder of the paper, given a cocomparability graph $G$ and a partial ordering $P$ of its complement $\overline{G}$, we denote by $\tilde{G} = \overline{C}(P)$ the bipartite complement of the domination bipartite graph $C(P)$ of $P$. Furthermore we denote by $H$ the associated split graph of $\tilde{G}$ and by $H^*$ the conflict graph of $H$. Moreover, we assume in the remainder of the paper without loss of generality that $\chi(H^*) \leq 2$, i.e. that $H^*$ is bipartite. Indeed, as we formally prove in Lemma 11, if $\chi(H^*) > 2$ then $\tilde{G}$ does not have a linear-interval cover, i.e. $\tilde{G}$ is not a PI graph. Note that every proper 2-coloring of the vertices of the conflict graph $H^*$ corresponds to exactly one 2-coloring of the edges of $H$ that includes no monochromatic $AC_4$. We assume in the following that a proper 2-coloring (with colors blue and red) of the vertices of $H^*$ is given as input; note that $\chi_0$ can be computed in polynomial time.

Lemma 11. Let $G$ be a cocomparability graph and $P$ be a partial order of $\overline{G}$. Let $\tilde{G} = \overline{C}(P)$, $H$ be the associated split graph of $\tilde{G}$, and $H^*$ be the conflict graph of $H$. If $\tilde{G}$ is linear-interval coverable, then $\chi(H^*) \leq 2$. 

12
Algorithm 2 Construction of the 3-CNF Boolean formula $\phi_1$

**Input:** The bipartite graph $\bar{G} = \bar{G}(P)$, the associated split graph $H$ of $\bar{G}$, its conflict graph $H^*$, and a proper 2-coloring $\chi_0$ of the vertices of $H^*$

**Output:** The 3-CNF Boolean formula $\phi_1$

1: $\phi_1 \leftarrow \emptyset$
2: for all triples of edges $\{e, e', e''\} \subseteq E(H)$, such that $\{e, e', e''\}$ build an AC6 in $E(H)$ do {note that this is an AC6 in the graph $H$ itself and not in a color subclass of its edges}
3: if $\ell_e \neq \ell_{e'}$, $\ell_{e'} \neq \ell_e$, and $\ell_e \neq \ell_{e''}$ then
4: if $\phi_1$ does not contain $(\ell_e \lor \ell_{e'} \lor \ell_{e''})$ and $(\ell_{e'} \lor \ell_e \lor \ell_{e''})$ then
5: $\phi_1 \leftarrow \phi_1 \land (\ell_e \lor \ell_{e'} \lor \ell_{e''}) \land (\ell_{e'} \lor \ell_e \lor \ell_{e''})$
6: return $\phi_1$

**Proof.** Suppose otherwise that $\chi(H^*) > 2$. Then $t(H) > 2$, since $\chi(H^*) \leq t(H)$ by Lemma 5. Therefore, Lemma 1 implies that $\chi(\bar{G}) > 2$, and thus $G$ is not a trapezoid graph [15]. Therefore $G$ is clearly not a PI graph, and thus $\bar{G}$ is not linear-interval coverable by Corollary 2, which is a contradiction to the assumption of the lemma. Therefore $\chi(H^*) \leq 2$.

Let $C_1, C_2, \ldots, C_k$ be the connected components of $H^*$. Some of these components of $H^*$ may be isolated vertices, which correspond to uncommitted edges in $H$. We assign to every component $C_i$, where $1 \leq i \leq k$, the Boolean variable $x_i$. Since $H^*$ is bipartite by assumption, the vertices of each connected component $C_i$ of $H^*$ can be partitioned into two color classes $S_{i,1}$ and $S_{i,2}$. Without loss of generality, we assume that $S_{i,1}$ (resp. $S_{i,2}$) contains the vertices of $C_i$ that are colored red (resp. blue) in $\chi_0$. Note that, since vertices of $H^*$ correspond to edges of $H$ (cf. Definition 7), for every two edges $e$ and $e'$ of $H$ that are in conflict (i.e. $e \parallel e'$) there exists an index $i \in \{1, 2, \ldots, k\}$ such that one of these edges belongs to $S_{i,1}$ and the other belongs to $S_{i,2}$. We now assign a literal $\ell_e$ to every edge $e$ of $H$ as follows: if $e \in S_{i,1}$ for some $i \in \{1, 2, \ldots, k\}$, then $\ell_e = x_i$; otherwise, if $e \in S_{i,2}$, then $\ell_e = \overline{x_i}$. Note that, by construction, whenever two edges are in conflict in $H$, their assigned literals are one the negation of the other.

**Observation 4** Every truth assignment $\tau$ of the variables $x_1, x_2, \ldots, x_k$ corresponds bijectively to a proper 2-coloring $\chi_\tau$ (with colors blue and red) of the vertices of $H^*$, as follows: $x_i = 0$ in $\tau$ (resp. $x_i = 1$ in $\tau$), if and only if all vertices of the component $C_i$ have in $\chi_\tau$ the same color as in $\chi_0$ (resp. opposite color than in $\chi_0$). In particular, $\tau = (0, 0, \ldots, 0)$ corresponds to the coloring $\chi_0$.

We now present the construction of the Boolean formulas $\phi_1$ and $\phi_2$ from the graphs $H$ and $H^*$, cf. Algorithms 2 and 3 respectively.

**Description of the 3-CNF formula $\phi_1$:** Consider an AC6 in the split graph $H$, and let $e, e', e''$ be its three edges in $H$, such that no two literals among $\{\ell_e, \ell_{e'}, \ell_{e''}\}$ are one the negation of the other. According to Algorithm 2, the Boolean formula $\phi_1$ has for this triple $\{e, e', e''\}$ of edges exactly the two clauses $\alpha = (\ell_{e'} \lor \ell_e \lor \ell_{e''})$ and $\alpha' = (\ell_{e'} \lor \ell_{e''} \lor \ell_e)$. It is easy to check by the assignment of literals to edges that the clause $\alpha$ (resp. the clause $\alpha'$) of $\phi_1$ is false in a truth assignment $\tau$ of the variables if and only if all edges $\{e, e', e''\}$ are colored red (resp. blue) in the 2-edge-coloring $\chi_\tau$ of $H$ (cf. Observation 4), as the following observation states.

**Observation 5** Let $\tau$ be any truth assignment of the variables $x_1, x_2, \ldots, x_k$. Let $\{e_1, e_2, e_3\}$ be the edges of an AC6 in $H$ and let $\alpha = (\ell_{e_1} \lor \ell_{e_2} \lor \ell_{e_3})$ and $\alpha' = (\ell_{e_1} \lor \ell_{e_2} \lor \ell_{e_3})$ be a the corresponding clauses in $\phi_1$. This AC6 is monochromatic in the coloring $\chi_\tau$ if and only if $\alpha = 0$ or $\alpha' = 0$ in $\tau$.

Consider now another AC6 of $H$ on the edges $\{e_1, e_2, e_3\}$, in which at least one literal among $\{\ell_{e_1}, \ell_{e_2}, \ell_{e_3}\}$ is the negation of another literal, for example $\ell_{e_1} = \ell_{e_2}$. Then, for any proper 2-coloring of the vertices of $H^*$, the edges $e$ and $e'$ of $H$ receive different colors, and thus this AC6 is not monochromatic. Thus the next observation follows by Observation 5.

**Observation 6** The formula $\phi_1$ is satisfied by a truth assignment $\tau$ if and only if the corresponding 2-coloring $\chi_\tau$ of the edges of $H$ does not contain any monochromatic AC6.

**Description of the 2-CNF formula $\phi_2$:** Denote for simplicity $H = (U, V, E_H)$, where $U = \{u_1, u_2, \ldots, u_n\}$ and $V = \{v_1, v_2, \ldots, v_n\}$. Furthermore denote $E_0 = \{u_i v_i \mid 1 \leq i \leq n\}$. Let $E' = E_H \setminus E_0$
Algorithm 3 Construction of the 2-CNF Boolean formula $\phi_2$

**Input:** The bipartite graph $G = \tilde{C}(P)$, the associated split graph $H$ of $G$, its conflict graph $H^*$, and a proper 2-coloring $\chi_\tau$ of the vertices of $H$.

**Output:** The 2-CNF Boolean formula $\phi_2$

1: Let $H = (U, V, E_H)$, where $U = \{u_1, u_2, \ldots, u_n\}$ and $V = \{v_1, v_2, \ldots, v_n\}$
2: $E_0 \leftarrow \{u_iv_i \mid 1 \leq i \leq n\}$; $E' \leftarrow E_H \setminus E_0$; $H' \leftarrow H - E_0$
3: $\phi_2 \leftarrow \emptyset$
4: for every pair $\{i, j\} \subseteq \{1, 2, \ldots, n\}$ with $u_iv_j \notin E'$ do
   5: for $t = 1, 2, \ldots, n$ do
     6: if $u_iv_t, u_jv_j \in E'$ then $\{\text{the edges } u_iv_t, u_jv_j \text{ are in conflict in } H' \text{ but not in } H\}$
   7: $e \leftarrow u_iv_i$; $e' \leftarrow u_jv_j$; $\phi_2 \leftarrow \phi_2 \land (\ell_e \lor \ell_{e'})$
8: return $\phi_2$

and $H' = H - E_0$, i.e. $H'$ is the split graph that we obtain if we remove from $H$ all edges of $E_0$. Consider now a pair of edges $e = u_iv_i$ and $e' = u_jv_j$ of $E'$, such that $u_iv_j \notin E'$. Note that $i$ and $j$ may be equal. However, since $E' \cap E_0 = \emptyset$, it follows that $i \neq t$ and $t \neq j$. Moreover, since the edge $u_iv_t$ belongs to $E_H$ but not to $E'$, it follows that the edges $e$ and $e'$ are in conflict in $H'$ but not in $H$ (for both cases where $i = j$ and $i \neq j$). That is, although $e$ and $e'$ are two non-adjoining vertices in the conflict graph $H^*$ of $H$, they are adjacent vertices in the conflict graph of $H'$. For both cases where $i = j$ and $i \neq j$, an example of such a pair of edges $\{e, e'\}$ is illustrated in Figure 3. According to Algorithm 3 for every such pair $\{e, e'\}$ of edges in $H$, the Boolean formula $\phi_2$ has the clause $(\ell_e \lor \ell_{e'})$. It is easy to check by the assignment of literals to edges of $H$ that this clause $(\ell_e \lor \ell_{e'})$ of $\phi_2$ is false in the truth assignment $\tau$ if and only if both $e$ and $e'$ are colored red in the 2-edge coloring $\chi_\tau$ of $H$.

![Figure 3](image-url)

**Fig. 3.** Two edges $e = u_iv_i$ and $e' = u_jv_j$ of $H$, for which the formula $\phi_2$ has the clause $(\ell_e \lor \ell_{e'})$, in the case where (a) $i \neq j$ and (b) $i = j$.

Now we provide the main result of this section in Theorem 7, which relates the existence of a linear-interval cover in $\tilde{C}(P)$ with the Boolean satisfiability of the formula $\phi_1 \land \phi_2$. Before we present Theorem 7 we first provide two auxiliary lemmas.

**Lemma 12.** Let $G$ be a cocomparability graph and $P$ be a partial order of $G$. Let $\tilde{G} = \tilde{C}(P)$, $H$ be the associated split graph of $G$, and $H^*$ be the conflict graph of $H$. Denote $\tilde{G} = (U, V, E_\tilde{G})$ and $E_0 = \{u_iv_i \mid 1 \leq i \leq n\}$. Then, every $e \in E_0$ is an isolated vertex of $H^*$.

**Proof.** Note by Definition 5 that $H = (U \cup V, E_H)$, where $E_H = E_\tilde{G} \setminus (V \times V)$. Furthermore all edges of $V \times V$ in $E_H$ correspond to isolated vertices in the conflict graph $H^*$ of $H$ by Observations 2 and 3. Therefore all non-isolated vertices in $H^*$ correspond to edges of $\tilde{G}$ (i.e. they do not belong to $V \times V$). Consider now an edge $e_i = u_iv_i \in E_0 \subseteq E_\tilde{G}$, where $1 \leq i \leq n$. Suppose that $e_i$ is not an isolated vertex in the conflict graph $H^*$. Then the edge $e_i$ of $\tilde{G}$ builds with another edge $e = u_jv_k$ an induced $AC_4$ in $H$, i.e. $e_i = u_iv_i$ and $e = u_jv_k$ induce a $2K_2$ in $\tilde{G}$. Therefore $u_jv_k, u_kv_k \notin E_\tilde{G}$, i.e. $u_jv_k, u_kv_k \in E(C(P))$. Thus $u_j <_P u_i$ and $u_i <_P u_k$ by Definition 8. Therefore, since $P$ is transitive (as a partial order), it follows that $u_j <_P u_k$, and thus $u_jv_k \in E(C(P))$, i.e. $u_jv_k \notin E_\tilde{G}$. This is a contradiction, since we assumed that $e = u_jv_k$ is an edge of $\tilde{G}$, i.e. $u_jv_k \in E_\tilde{G}$. Therefore, $e_i = u_iv_i$ is an isolated vertex of $H^*$. □

**Lemma 13.** Let $H$ be a split graph and $H^*$ be the conflict graph of $H$, where $H^*$ is bipartite with color classes $E_1$ and $E_2$. Let the vertices $v_1, \ldots, v_6$ of $H$ build an $AC_6$ on the edges of $E_i$, where $i \in \{1, 2\}$. Then the edges $v_3v_6, v_4v_1, v_5v_2$ exist in $H$ and $v_4v_5 \mid v_3v_6, v_2v_3 \mid v_4v_1$, and $v_6v_1 \mid v_5v_2$. 

14
Proof. Since $H$ is a split graph, Lemma 2 implies that $H$ does not contain any $AP_5$ or any double $AP_4$. Therefore, the $AC_6$ of $H$ is an $AP_6$, i.e. an alternating path of length 6, cf. Figure 1(b). Since $E_1$ and $E_2$ are the two color classes of $H^*$, any two vertices $e$ and $e'$ of $H^*$ in the set $E_i$, where $i \in \{1, 2\}$, are not adjacent in $H^*$. Equivalently, any two edges $e$ and $e'$ of $H$ in the set $E_i$ are not in conflict, where $i \in \{1, 2\}$. Therefore, since by assumption, all edges $\{v_2v_3, v_4v_5, v_6v_1\}$ of this $AC_6$ belong to the same color class $E_i$ for some $i \in \{1, 2\}$, it follows that no pair of these edges is in conflict in $H$. Thus Lemma 2 implies that the edges $v_2v_6, v_2v_1, v_2v_2$ exist in $H$ and that $v_2v_3 || v_2v_6, v_2v_5 || v_2v_1$, and $v_2v_1 || v_2v_2$. □

We are now ready to provide Theorem 7.

Theorem 7. $\tilde{G} = \tilde{G}(P)$ is linear-interval colorable if and only if $\phi_1 \land \phi_2$ is satisfiable. Given a satisfying assignment $\tau$ of $\phi_1 \land \phi_2$, Algorithm 5 computes a linear-interval cover of $\tilde{G}$ in $O(n^2)$ time.

Proof. Denote $\tilde{G} = (U, V, E), U = \{u_1, u_2, \ldots, u_n\}$ and $V = \{v_1, v_2, \ldots, v_n\}$. Furthermore denote $H = (U, V, E_H)$, where $E_H = \tilde{E} \cup (V \times V)$, cf. Definition 6. Let $E_0 = \{u_i, v_i \mid 1 \leq i \leq n\}$. Since $\tilde{G} = \tilde{G}(P)$, note by Definition 6 that $E_0 \subseteq \tilde{E} \subseteq E_H$. Let $\chi_0$ be the 2-coloring of the vertices of $H^*$ (i.e. the edges of $H$) that is given as input to Algorithms 2 and 3. Moreover, let $C_1, C_2, \ldots, C_k$ be the connected components of $H^*$.

$(\Rightarrow)$ Suppose that $\tilde{G}$ is linear-interval colorable. That is, there exist by Definition 10 two chain graphs $G_1 = (U, V, E_1)$ and $G_2 = (U, V, E_2)$, such that $\tilde{G} = G_1 \cup G_2$ and $E_0 \subseteq E_2 \setminus E_1$. Let $H_1 = (U, V, E_{H_1})$ and $H_2 = (U, V, E_{H_2})$ be the associated split graphs of $G_1$ and $G_2$, respectively. Note that $H = H_1 \cup H_2$ and $E_0 \subseteq E_{H_2} \setminus E_{H_1}$. Since $G_1$ and $G_2$ are chain graphs, i.e. $ch(G_1) = ch(G_2) = 1$, Lemma 1 implies that $t(H_1) = t(H_2) = 0$, i.e. $H_1$ and $H_2$ are threshold graphs. Therefore, neither $H_1$ nor $H_2$ includes an $AC_4$

Recall that the formulas $\phi_1$ and $\phi_2$ have one Boolean variable $x_i$ for every connected component $C_i$ of $H^*$, $i \in \{1, 2, \ldots, k\}$. We construct a 2-coloring $\chi$ of the edges of $H$ as follows. For every edge $e$ of $H$ (i.e. a vertex of $H^*$), if $e \in E_{H_1}$, then we color $e$ red in $\chi_H$; otherwise, if $e \in E_{H_2} \setminus E_{H_1}$, then we color $e$ blue in $\chi_H$. Recall that $E_0 \subseteq E_{H_2} \setminus E_{H_1}$, and thus all edges of $E_0$ are colored blue in $\chi_H$. Since both $H_1$ and $H_2$ do not include any $AC_4$, it follows by the definition of $\chi_H$ that there exists no monochromatic $AC_4$ in $\chi_H$. Therefore, every two edges $e$ and $e'$ of $H$, which correspond to adjacent vertices in $H^*$, have different colors in $\chi_H$, and thus $\chi_H$ constitutes a proper 2-coloring of the vertices of $H^*$. Therefore the coloring $\chi_H$ of the edges of $H$ (i.e. vertices of $H^*$) defines a truth assignment $\tau$ of the variables $x_1, x_2, \ldots, x_k$ as follows (cf. Observation 4). For every connected component $C_i$ of $H^*$, where $1 \leq i \leq k$, we define $x_i = 0$ (resp. $x_i = 1$) if and only if all vertices of $C_i$ have in $\chi_H$ different (resp. the same) color as in $\chi_0$. We will now prove that $\tau$ satisfies both formulas $\phi_1$ and $\phi_2$.

**Satisfaction of the Boolean formula $\phi_1$.** Let $\alpha$ be a clause of $\phi_1$. Recall that $\alpha$ corresponds to some triple $(e, e', e'')$ of edges of $H$ that builds an $AC_6$ in $H$ (cf. lines 245 of Algorithm 2). In particular, either $\alpha = (e \lor e' \lor e'')$ or $\alpha = (e'' \lor e' \lor e'')$, where $e, e', e''$ are the literals that have been assigned to the edges $e, e', e''$, respectively. Then, it follows from the description of the formula $\phi_1$ (cf. also Observation 5) that the clause $(e \lor e' \lor e'')$ (resp. the clause $(e'' \lor e' \lor e'')$) is not satisfied in the truth assignment $\tau$ if and only if the edges $e, e', e''$ of $H$ are all red (resp. all blue) in $\chi_H$.

Let $\alpha = (e \lor e' \lor e'')$ (resp. $\alpha = (e'' \lor e' \lor e'')$). Suppose that $\alpha$ is not satisfied by $\tau$, and thus the edges $e, e', e''$ of $H$ are all red (resp. blue) in $\chi_H$. Therefore all edges $e, e', e''$ belong to $E_{H_1}$ (resp. to $E_{H_2} \setminus E_{H_1}$, and thus to $E_{H_2}$) by the definition of $\chi_H$. Thus $H$ has an $AC_6$ on the edges $e, e', e''$, which belong to $H_1$ (resp. to $H_2$). Therefore $H_1$ (resp. $H_2$) does not have a threshold completion in $H$ by Lemma 3. This is a contradiction, since $H_1$ (resp. $H_2$) is a threshold graph. Therefore the clause $\alpha = (e \lor e' \lor e'')$ of $\phi_1$ is satisfied by the truth assignment $\tau$, and thus $\tau$ satisfies $\phi_1$.

**Satisfaction of the Boolean formula $\phi_2$.** Let $\alpha = (e \lor e')$ be a clause of $\phi_2$. Recall that $\alpha$ corresponds to some pair of edges $e = u_1v_1$ and $e' = u_4v_2$ of $E_{H_2} \setminus E_0$, such that $u_2, v_2 \not\in \tilde{E} \setminus E_0$ (cf. lines 4 of Algorithm 3). Therefore, since $u_1v_1 \in E_0$, it follows that the edges $(e, e')$ build an $AC_4$ in $H - E_0$ but not in $H$. Suppose that the clause $\alpha = (e \lor e')$ of $\phi_2$ is not satisfied by the truth assignment $\tau$, i.e. $\ell_2 = \ell_3 = 0$ in $\tau$. Then it follows from the description of the formula $\phi_2$ that both $e$ and $e'$ are colored red in the 2-edge coloring $\chi_H$ of $H$. Therefore both edges $e$ and $e'$ belong to $H_1$ by the definition of $\chi_H$. However, as we noticed above, the edges $(e, e')$ build an $AC_4$ in $H - E_0$, and thus they also build an $AC_4$ in $H \subseteq H - E_0$. This is a contradiction by Corollary 1 since $H_1$ is a threshold graph. Therefore the clause $\alpha = (e \lor e')$ of $\phi_2$ is satisfied by the truth assignment $\tau$, and thus $\tau$ satisfies $\phi_2$.

$(\Leftarrow)$ Suppose that $\phi_1 \land \phi_2$ is satisfiable, and let $\tau$ be a satisfying truth assignment of $\phi_1 \land \phi_2$. Recall that the formulas $\phi_1$ and $\phi_2$ have one Boolean variable $x_i$ for every connected component $C_i$ of $H^*$, $i \in \{1, 2, \ldots, k\}$. First, given the truth assignment $\tau$, we construct the 2-coloring $\chi_\tau$ of the vertices of $H^*$ according to
Observation[4] This 2-coloring of the vertices of $H^*$ defines also a corresponding 2-coloring of the edges of $H$. Since $\phi_1$ is satisfied by $\tau$, it follows by Observation[5] that, in the coloring $\chi_\tau$ of its edges, $H$ does not contain any monochromatic $AC_6$. Therefore Theorem[6] implies that $H$ does not contain any monochromatic $AC_{2k}$ in $\chi_\tau$, where $k \geq 3$.

The vertex coloring $\chi_\tau'$ of $H^*$. Now we modify the coloring $\chi_\tau$ to the coloring $\chi_\tau'$, as follows. For every trivial connected component $C_i$ of $H^*$ (i.e. when $C_i$ has exactly one vertex), we color the vertex of $C_i$ blue in $\chi_\tau'$, regardless of the color of $C_i$ in $\chi_\tau$. On the other hand, for every non-trivial connected component $C_i$ of $H^*$ (i.e. when $C_i$ has at least two vertices), the vertices of $C_i$ have the same color in both $\chi_\tau$ and $\chi_\tau'$. This new 2-coloring of the vertices of $H^*$ defines also a corresponding 2-coloring of the edges of $H$. Note in particular by Lemma[12] that all edges of $E_0$ are colored blue in $\chi_\tau'$. Denote by $E_{H_1}$ and $E_{H_2}$ the sets of red and blue edges of $H$ in $\chi_\tau'$, respectively. Note that $E_0 \subseteq E_{H_2}$. Moreover note that $H$ does not have any $AC_4$ on the vertices of $E_{H_1}$, or on the vertices of $E_{H_2}$, since $\chi_\tau'$ is a proper 2-coloring of the vertices of $H^*$. Define the subgraphs $H_1 = (U, V, E_{H_1})$ and $H_2 = (U, V, E_{H_2})$ of $H$. Note that $H = H_1 \cup H_2$.

$H_2$ has a threshold completion in $H$. Suppose now that $H$ has an $AC_{2k}$ on the edges of $E_{H_2}$, for some $k \geq 3$. Then Theorem[5] implies that $H$ also has an $AC_6$ on the edges of $E_{H_1}$, i.e. $H$ has an $AC_6$, in which all three colors are blue in $\chi_\tau'$. Since $H$ does not have any monochromatic $AC_6$ in $\chi_\tau$, it follows that for at least one of the edges $e$ of the blue $AC_6$ of $H$ in $\chi_\tau'$, the color of $e$ is different in $\chi_\tau$ and in $\chi_\tau'$. Therefore, it follows by the construction of $\chi_\tau'$ from $\chi_\tau$ that the vertex $e$ of $H^*$ that corresponds to $e$ is an isolated vertex in $H^*$. That is, the edge $e$ is uncommitted in $H$. This is a contradiction by Lemma[13] since $e$ has been assumed to be an edge of a monochromatic $AC_6$ of $H$ in $\chi_\tau'$. Therefore $H$ does not have any $AC_{2k}$ on the edges of $E_{H_2}$, where $k \geq 3$. Thus, since $H$ does not have any $AC_4$ on the vertices of $E_{H_1}$, it follows that $H$ does not have any $AC_{2k}$ on the edges of $E_{H_2}$, where $k \geq 2$. Therefore $H_2$ has a threshold completion in $H$ by Lemma[3].

$H_1$ has a threshold completion in $H - E_0$. Denote now $H' = H - E_0$. We will prove that $H_1$ has a threshold completion in $H'$. To this end, it suffices to prove by Lemma[3] that $H'$ does not have any $AC_{2k}$ on the edges of $E_{H_1}$, where $k \geq 2$.

For the sake of contradiction, suppose that $H'$ includes an $AC_4$ on the edges of $E_{H_1}$. That is, there exist two edges $e, e' \in E_{H_1}$, that are in conflict in $H'$. Note by the definition of $E_{H_1}$ that the edges $e$ and $e'$ are colored red in $\chi_\tau$, and thus they are also colored red in $\chi_\tau'$. If the edges $\{e, e'\}$ also build an $AC_4$ in $H$ (i.e. before the removal of $E_0$), then the vertices $e$ and $e'$ are adjacent in $H^*$, and thus the edges $e$ and $e'$ have different colors in $\chi_\tau$, which is a contradiction. Thus the edges $\{e, e'\}$ are in conflict in $H'$ but not in $H$. Recall now that for every such a pair $\{e, e'\}$ of edges of $H'$ there exists a clause $\alpha = (\ell_e \lor \ell_{e'})$ in the formula $\phi_2$ (cf. lines[14] of Algorithm[6]). It follows from the description of the formula $\phi_2$ that the clause $\alpha$ is satisfied by the truth assignment $\tau$ if and only if both edges $e, e'$ are in $\chi_\tau$ in red colors. However, since $\tau$ is a satisfying assignment of $\phi_2$, every clause of $\phi_2$ is satisfied by $\tau$. Therefore at least one of the edges $e$ and $e'$ is colored blue in $\chi_\tau$, which is a contradiction. Therefore $H'$ does not include any $AC_4$ on the edges of $E_{H_1}$.

Suppose now that $H'$ includes an $AC_{2k}$ on the edges of $E_{H_1}$, where $k \geq 3$. Consider the smallest such $AC_{2k}$ on the edges of $E_{H_1}$, i.e. an $AC_{2k}$ with the smallest $k \geq 3$. Let $w_1, w_2, \ldots, w_{2k}$ be the vertices of $H'$ that build this $AC_{2k}$. Note by the definition of $E_{H_1}$ that all edges of this $AC_{2k}$ are colored red in the coloring $\chi_\tau'$, and thus they are also colored red in the coloring $\chi_\tau$. However, as we proved above, in the coloring $\chi_\tau$ of its edges, $H$ does not contain any monochromatic $AC_{2k}$, where $k \geq 3$. Therefore, at least one of the non-edges of the $AC_{2k}$ in the graph $H'$ is an edge of $E_0$ in the graph $H$. Assume without loss of generality that this edge of $E_0$ is $w_1 w_2$. That is, assume that $w_1 w_2 \in E_0$, i.e. $w_{2i} w_{2i+1} = u_i v_i$ for some $i \in \{1, 2, \ldots, n\}$.

Suppose that $w_3 w_{2k}$ is not an edge of $H'$. Then, since $w_1 w_2 \in E_0$, there exists (similarly to above) a clause $\alpha$ in the formula $\phi_2$ such that $\alpha$ is satisfied by the truth assignment $\tau$ if and only if both edges $w_3 w_2$ and $w_{2k} w_1$ are colored red in $\chi_\tau$. However, $\tau$ is a satisfying truth assignment of $\phi_2$ by assumption, and thus at least one edge of $w_3 w_2$ and $w_{2k} w_1$ is colored blue in $\chi_\tau$, which is a contradiction. Therefore $w_3 w_{2k}$ is an edge of $H'$. Suppose now that the edge $w_3 w_{2k}$ of $H'$ is colored red in $\chi_\tau'$, and thus $w_{2k} w_1 \in E_{H_1}$ by the definition of $E_{H_1}$. Then the vertices $w_3, w_4, \ldots, w_{2k}$ build an $AC_{2k-2}$ in $H'$ on the edges of $E_{H_1}$, which is a contradiction to the minimality assumption of the $AC_{2k}$ in $H'$. Therefore the edge $w_3 w_{2k}$ of $H'$ is colored blue in $\chi_\tau'$, and thus $w_3 w_{2k} \in E_{H_1}$.

Recall now that both the edges $w_3 w_2$ and $w_{2k} w_1$ of $H'$ are red in $\chi_\tau'$. Therefore, by the definition of the coloring $\chi_\tau'$, it follows that each of the edges $w_3 w_2$ and $w_{2k} w_1$ participates to at least one $AC_4$ in $H$ (or equivalently the corresponding vertices of $w_3 w_2$ and $w_{2k} w_1$ in $H^*$ are not isolated vertices). Let the edges $w_3 w_2$ and $w_{2k} w_1$ form an $AC_4$ in $H$, for some vertices $w_2'$ and $w_1'$, where $w_2 w_2'$ and $w_{2k} w_{2k}'$ are not edges in $H$. Similarly, let the edges $w_3 w_2$ and $w_{2k} w_1$ form an $AC_4$ in $H$, for some vertices $w_2'$ and $w_1'$, where $w_2 w_2'$ and $w_1 w_1'$ are not edges in $H$. Note that some of the vertices $\{w_2', w_3, w_{2k}', w_1'\}$ may coincide with
each other, as well as with some of the vertices \{w_2, w_3, w_{2k}, w_1\}. Recall that \(\chi'_x\) is a proper 2-coloring of the vertices of \(H^*\). Therefore, since \(w_2w_3\) and \(w_{2k}w_1\) are colored red in \(\chi'_x\), it follows that \(w'_2w'_3\) and \(w'_{2k}w'_1\) are colored blue in \(\chi'_x\). Therefore the vertices \(w_1, w_2, w'_2, w'_3, w_{2k}, w'_{2k}, w'_1\) build an \(AC_5\) in \(H^*\) on the edges of \(E_{H_2}\). This is a contradiction, since we proved above that \(H\) does not have any \(AC_2k\) on the edges of \(E_{H_2}\), where \(k \geq 2\).

Therefore, it follows that \(H^*\) does not include any \(AC_2k\) on the edges of \(E_{H_1}\), where \(k \geq 3\). Thus, since we already proved that \(H^*\) does not include any \(AC_2k\) on the edges of \(E_{H_1}\), it follows that \(H^*\) does not include any \(AC_2k\) on the edges of \(E_{H_1}\), where \(k \geq 2\). Therefore \(H_1\) has a threshold completion in \(H' = H - E_0\) by Lemma 6.

Summarizing, \(H_1\) has a threshold completion in \(H' = H - E_0\), and \(H_2\) has a threshold completion in \(H\). Furthermore all edges of \(E_0\) belong to the graph \(H\), and \(H = H_1 \cup H_2\). Let \(\bar{H}_1\) be the threshold completion of \(H_1\) in \(H - E_0\), and let \(\bar{H}_2\) be the threshold completion of \(H_2\) in \(H\). Then \(H_1\) and \(\bar{H}_2\) are two threshold graphs, i.e. they do not include any \(AC_4\). Furthermore, let \(\tilde{G}_1 = (U, V, \tilde{E}_1)\) and \(\tilde{G}_2 = (U, V, \tilde{E}_2)\) be the bipartite graphs obtained by \(\tilde{H}_1\) and \(\tilde{H}_2\), respectively, by removing from them all possible edges of \(V \times V\). Note that \(E_0 \subseteq \tilde{E}_2 \setminus \tilde{E}_1\), since every edge of \(E_0\) belongs to \(H_2\) and not to \(\tilde{H}_1\). Furthermore, neither \(\tilde{G}_1\) nor \(\tilde{G}_2\) include any induced \(2K_2\), since \(\tilde{H}_1\) and \(\tilde{H}_2\) do not include any \(AC_4\). Therefore both \(\tilde{G}_1\) and \(\tilde{G}_2\) are chain graphs. Moreover, since \(H = H_1 \cup H_2\), it follows that also \(H = \tilde{H}_1 \cup \tilde{H}_2\) and \(\tilde{G} = \tilde{G}_1 \cup \tilde{G}_2\). Thus, since \(E_0 \subseteq \tilde{E}_2 \setminus \tilde{E}_1\), it follows that \(\tilde{G}\) is linear-interval coverable by Definition 10 and \(\{\tilde{E}_1, \tilde{E}_2\}\) is a linear-interval cover of \(\tilde{G}\). This construction of \(\{\tilde{E}_1, \tilde{E}_2\}\) from the satisfying truth assignment \(\tau\) of \(\phi_1 \land \phi_2\) is shown in Algorithm 4.

Running time of Algorithm 4. First note that, since \(|U| = |V| = n\), the split graph \(H\) has \(O(n^2)\) edges. Therefore, since each edge of \(H\) is processed exactly once in the execution of lines 11\(\,\)and 12 in Algorithm 4, these lines are executed in \(O(n^2)\) time in total. Similarly, each of the lines 9, 10, and 13 is executed in \(O(n^2)\) time. Now, each of the lines 11 and 12 is executed by Lemma 4 in time linear to the size of \(H\), i.e. in \(O(n^2)\) time each. Therefore the total running time of Algorithm 4 is \(O(n^2)\). This completes the proof of the theorem.

6 The recognition of linear-interval orders and PI graphs

In this section we investigate the structure of the formula \(\phi_1 \land \phi_2\) that we computed in Section 5. In particular, we first prove in Section \(\frac{6.1}{()}\) some fundamental structural properties of \(\phi_1 \land \phi_2\), which allow us to find an appropriate sub-formula of \(\phi_1 \land \phi_2\) which is gradually mixed (cf. Definition 2). Then we exploit this sub-formula of \(\phi_1 \land \phi_2\) in order to provide in Section \(\frac{6.2}{()}\) an algorithm that solves the satisfiability problem on \(\phi_1 \land \phi_2\) in time linear to its size, cf. Theorem 8. Finally, using this satisfiability algorithm, we combine our results of Sections 4 and 5 in order to recognize efficiently PI graphs and linear-interval orders in Section \(\frac{6.2}{()}\).
6.1 Structural properties of the formula $\phi_1 \land \phi_2$

The three main structural properties of $\phi_1 \land \phi_2$ are proved in Lemmas 16, 17, and 18 respectively. We first provide two auxiliary technical lemmas.

**Lemma 14.** Let $\alpha = (\ell_1 \lor \ell_2 \lor \ell_3)$ be a clause of $\phi_1$. Assume that $\alpha$ corresponds to the $\text{AP}_k$ of $H$ on the vertices $v_1, \ldots, v_6$, which has the literals $\ell_1, \ell_2, \ell_3$ on its edges (in this order). Then, for every edge $e$ of $H$ with literal $\ell_e = \ell_2$, there exists an $\text{AP}_k$ in $H$ with $v_1v_2$ as its base and $e$ as its ceiling, which has the literals $\ell_1, \ell_2, \ell_3$ on its edges (in this order).

**Proof.** First note that by the construction of $\phi_1$ (cf. Section 5) no two literals among $\{\ell_1, \ell_2, \ell_3\}$ are one the negation of the other, i.e. $\ell_1 \neq \ell_2$, $\ell_1 \neq \ell_3$, and $\ell_2 \neq \ell_3$. Therefore also no pair among the edges of the $\text{AP}_k$ on the vertices $v_1, \ldots, v_6$ is in conflict. Denote for simplicity $e' = v_4v_5$. Since $\ell_{e'} = \ell_2 = \ell_2$, the edges $e'$ and $e$ of $H$ correspond to two vertices of the conflict graph $H^*$ that lie in the same connected component of $H^*$. Thus there exists a path between these two vertices of $H^*$. That is, there exists a sequence of edges $e_1, e_2, \ldots, e_t$ in $H$, where $e_1 = e'$ and $e_t = e$, such that $e_i || e_{i+1}$ for every $i \in \{1, 2, \ldots, t-1\}$. Note that $\ell_{e_i} \in \{\ell_2, \ell_2\}$ for all these edges $e_i$. For every $1 \leq i \leq t$ denote $e_i = u_iv_i$, where $u_1 = v_4$ and $v_1 = v_5$. Furthermore let $u_1u_{i+1}$ and $w_{i+1}w_{i+1}$ be the non-edges between $e_i$ and $e_{i+1}$, where $1 \leq i \leq t-1$. For simplicity of the presentation, denote $u_0 = v_3$ and $w_0 = v_6$.

We will prove by induction that for every $i \in \{1, 2, \ldots, t\}$ there exists an $\text{AP}_k$ in $H$ on the vertices $v_1, v_2, u_{i-1}, u_i, w_{i-1}, w_i-1$ (if $i$ is odd), or on the vertices $v_1, v_2, u_i, u_{i-1}, w_{i-1}, w_{i}$ (if $i$ is even), which has the literals $\ell_1, \ell_2, \ell_3$ on its edges (in this order). The induction basis (i.e. the case where $i = 1$) follows immediately by the assumption of the lemma.

For the induction step, let first $i \geq 2$ be even. Then $i - 1$ is odd, and thus there exists by the induction hypothesis an $\text{AP}_k$ in $H$ on the vertices $v_1, v_2, u_{i-2}, u_{i-1}, w_{i-1}, w_i-1$ which has the literals $\ell_1, \ell_2, \ell_3$ on its edges (in this order). That is, $\ell_{v_iu_{i-2}} = \ell_1$, $\ell_{u_{i-2}u_{i-1}} = \ell_2$, and $\ell_{w_{i-2}w_{i-1}} = \ell_3$. Therefore, since $\ell_{v_iu_{i-2}} \in \{\ell_2, \ell_3\}$ and $u_{i-1}w_{i-1} || u_{i-2}w_{i-1}$ by assumption, it follows that $\ell_{u_{i-1}w_{i-1}} = \ell_2$. Furthermore, since no pair among the edges of the $\text{AP}_k$ is in conflict, Lemma 2 implies in particular that the edges $v_1u_{i-2}$ and $v_2w_{i-1}$ exist in $H$ and that $\ell_{v_1u_{i-2}} = \ell_1$ and $\ell_{v_2w_{i-1}} = \ell_2$.

**Claim 1** $v_1 \neq w_i$ and $v_2 \neq u_i$.

**Proof of Claim 1.** Since $H$ is a split graph, there exists a partition of its vertices into a clique $K$ and an independent set $I$. Then, since $H$ has an $\text{AP}_k$ on the vertices $v_1, v_2, u_{i-2}, u_{i-1}, w_{i-1}, w_i-1$, Lemma 5 implies that either $v_1, u_{i-2}, w_{i-1} \in K$ and $v_2, u_{i-1}, w_i-1 \in I$, or $v_1, u_{i-2}, w_{i-1} \in I$ and $v_2, u_{i-1}, w_i-1 \in K$. In the former case, since $w_{i-1} \in K$ and $w_{i-1}w_i$ is not an edge in $H$, it follows that $w_i \in I$. Thus $v_1 \neq w_i$, since $v_1 \in K$. Furthermore, since $u_i \in I$ and $u_iw_i$ is an edge in $H$, it follows that $u_i \in K$. Thus $v_2 \neq u_i$, since $v_2 \in K$. Similarly, in the latter case, since $u_{i-1} \in K$ and $u_{i-1}u_i$ is not an edge in $H$, it follows that $u_i \in I$. Thus $v_2 \neq u_i$, since $v_2 \in K$. Furthermore, since $u_i \in I$ and $u_iw_i$ is an edge in $H$, it follows that $w_i \in K$. Thus $v_1 \neq w_i$, since $v_1 \in I$. Summarizing, in both cases $v_1 \neq w_i$ and $v_2 \neq u_i$.

Suppose that $v_1, v_2, w_i, u_i$ is not an edge in $H$. Then $u_iw_i$ is in conflict with $v_1u_{i-1}$, since also $u_{i-1}u_i$ is not an edge in $H$. Therefore $\ell_{v_1u_{i-1}} = \ell_{u_{i-1}u_i}$. Thus, since $\ell_{u_{i-1}u_i} = \ell_2$ and $\ell_{v_1u_{i-1}} = \ell_1$, it follows that $\ell_1 = \ell_2$, which is a contradiction, since no two literals among $\{\ell_1, \ell_2, \ell_3\}$ are one the negation of the other. Therefore $v_1w_i$ is an edge in $H$. Furthermore $\ell_{v_1u_{i-1}} = \ell_3$, since $\ell_{v_2w_{i-1}} = \ell_3$ and $w_{i-1}w_i$, $v_1w_i$ are not edges in $H$. By symmetry it follows that also $v_2u_i$ is an edge in $H$ and that $\ell_{v_2u_i} = \ell_3$. Thus the vertices $v_1, v_2, u_{i-1}, w_{i-1}, w_i, u_i$ build an $\text{AP}_k$ in $H$, which has the literals $\ell_1, \ell_2, \ell_3$ on its edges (in this order). This completes the induction step whenever $i$ is odd.

Let now $i \geq 3$ be even. Then $i - 1$ is even, and thus there exists by the induction hypothesis an $\text{AP}_k$ in $H$ on the vertices $v_1, v_2, u_{i-1}, u_{i-2}, w_{i-2}, w_i-1$ which has the literals $\ell_1, \ell_2, \ell_3$ on its edges (in this order). That is, $\ell_{v_1u_{i-1}} = \ell_1$, $\ell_{u_{i-2}w_{i-2}} = \ell_2$, and $\ell_{w_{i-2}w_i-1} = \ell_3$. Thus, since the edges $u_{i-2}w_{i-2}$ and $u_{i-1}w_{i-1}$ are in conflict by assumption, it follows that $\ell_{u_{i-1}w_{i-1}} = \ell_2$. Furthermore, since the edges $u_{i-1}w_{i-1}$ and $u_iw_i$ are in conflict by assumption, it follows that $\ell_{u_iw_i} = \ell_2$. Thus the vertices $v_1, v_2, u_{i-2}, u_{i-1}, w_{i-1}, u_i, w_i$ build an $\text{AP}_k$ in $H$, which has the literals $\ell_1, \ell_2, \ell_3$ on its edges (in this order). This completes the induction step whenever $i$ is odd.

Summarizing, for $i = t$, there exists an $\text{AP}_k$ in $H$ on the vertices $v_1, v_2, u_{t-1}, u_{t-2}, w_{t-2}, w_i$ (if $t$ is odd), which has the literals $\ell_1, \ell_2, \ell_3$ on its edges (in this order). In both cases where $t$ is even or odd, this $\text{AP}_k$ has the non-edge $v_1v_2$ as it base and the edge $e = u_iw_i$ as its ceiling. This completes the proof of the lemma. □
Lemma 15. Let $\alpha = (\ell_1 \lor \ell_2 \lor \ell_3)$ and $\beta = (\ell_1 \lor \ell_2 \lor \ell_4)$ be two clauses of $\phi_1$ that share two literals $\ell_1$ and $\ell_2$. Then also $\ell_3 = \ell_4$.

Proof. By the construction of the formula $\phi_1$ (cf. Section 5), the clauses $\alpha$ and $\beta$ correspond to two $AC_6$'s in $H$. Since $H$ is a split graph, Lemma 7 implies that each of these two $AC_6$'s is an $AP_6$, i.e. an alternating path of length 6 (cf. Figure 1(b)). Let $v_1, v_2, v_3, v_4, v_5, v_6$ be the vertices of the first $AP_6$, which has the literals $\ell_1, \ell_2, \ell_3$ on its edges (in this order). Note that, by the construction of $\phi_1$, no two literals among $\{\ell_1, \ell_2, \ell_3\}$ are one the negation of the other, i.e. $\ell_1 \neq \ell_2$, $\ell_1 \neq \ell_3$, and $\ell_2 \neq \ell_3$. Furthermore let $w_1, w_2, w_3, w_4, w_5, w_6$ be the vertices of the second $AP_6$, which has the literals $\ell_1, \ell_2, \ell_4$ on its edges (in this order). Since $H$ is a split graph, there exists a partition of its vertices into a clique $K$ and an independent set $I$.

Consider now the base $v_5v_6$ and the ceiling $v_2v_3$ of the first $AP_6$ (cf. Definition 4). That is, the vertices of this $AP_6$ can be ordered as $v_5, v_6, v_1, v_2, v_3, v_4$ (where $v_5v_6$ is not an edge); then the literals on its edges are $\ell_3, \ell_1, \ell_2$ (in this order). Since $\ell_{v_5v_6} = \ell_{v_2v_3} = \ell_1$, there exists by Lemma 14 an $AP_6$ with $v_5v_6$ as its base and $v_2v_3$ as its ceiling, which has the literals $\ell_3, \ell_1, \ell_2$ on its edges (in this order). Note that the ordering of the vertices in this $AP_6$ can be either $v_5, v_6, a, w_3, w_2, b$, or $v_5, v_6, a, w_2, w_3, b$, for some vertices $a$ and $b$ of $H$. We distinguish now these two cases.

Case 1. The $AP_6$ with $v_5v_6$ as its base and $v_2v_3$ as its ceiling has vertex ordering $v_5, v_6, a, w_3, w_2, b$. Consider now the base $aw_3$ and the ceiling $bv_5$ of this $AP_6$. That is, its vertices can be ordered as $a, w_3, w_2, b, v_5, v_6$ (where $aw_3$ is not an edge); then the literals on its edges are $\ell_1, \ell_2, \ell_3$ (in this order). Since $\ell_{aw_3} = \ell_{bw_3} = \ell_2$, there exists by Lemma 14 an $AP_6$ with $aw_3$ as its base and $bw_3$ as its ceiling, which has the literals $\ell_1, \ell_2, \ell_3$ on its edges (in this order). Note that the ordering of the vertices in this $AP_6$ can be either $a, w_3, c, w_5, w_4, d$, or $a, w_3, c, w_4, w_5, d$, for some vertices $c$ and $d$ of $H$. We distinguish now these two cases.

Case 1.1. The $AP_6$ with $aw_3$ as its base and $bw_3$ as its ceiling has vertex ordering $a, w_3, c, w_5, w_4, d$. Since no two literals among $\{\ell_1, \ell_2, \ell_3\}$ are one the negation of the other, it follows that no pair among the edges of this $AP_6$ is in conflict. Thus Lemma 2 implies in particular that the edge $w_3w_4$ exists in $H$. This is a contradiction to our initial assumption that the vertices $a, w_1, w_2, w_3, w_4, w_5, w_6$ build an $AC_6$ (and thus $w_3w_4$ is not an edge).

Case 1.2. The $AP_6$ with $aw_3$ as its base and $bw_3$ as its ceiling has vertex ordering $a, w_3, c, w_5, w_4, d$. Then Lemma 6 implies that either $w_3 \in K$ and $w_5 \in I$, or $w_3 \in I$ and $w_5 \in K$. However, due to our initial assumption that the vertices $a, w_1, w_2, w_3, w_4, w_5, w_6$ build an $AC_6$, Lemma 6 implies that either $w_3, w_5 \in K$ or $w_3, w_5 \in I$, which is a contradiction.

Case 2. The $AP_6$ with $v_5v_6$ as its base and $v_2v_3$ as its ceiling has vertex ordering $v_5, v_6, a, w_2, w_3, b$. Consider now the base $aw_2$ and the ceiling $bv_5$ of this $AP_6$. That is, its vertices can be ordered as $a, w_2, w_3, b, v_5, v_6$ (where $aw_2$ is not an edge); then the literals on its edges are $\ell_1, \ell_2, \ell_3$ (in this order). Since $\ell_{aw_2} = \ell_{bw_3} = \ell_2$, there exists by Lemma 14 an $AP_6$ with $aw_2$ as its base and $bw_3$ as its ceiling, which has the literals $\ell_1, \ell_2, \ell_3$ on its edges (in this order). Note that the ordering of the vertices in this $AP_6$ can be either $a, w_2, c, w_5, w_4, d$, or $a, w_2, c, w_4, w_5, d$, for some vertices $c$ and $d$ of $H$. We distinguish now these two cases.

Case 2.1. The $AP_6$ with $aw_2$ as its base and $bw_3$ as its ceiling has vertex ordering $a, w_2, c, w_5, w_4, d$. Then Lemma 6 implies that either $w_2 \in K$ and $w_4 \in I$, or $w_2 \in I$ and $w_4 \in K$. However, due to our initial assumption that the vertices $a, w_1, w_2, w_3, w_4, w_5, w_6$ build an $AC_6$, Lemma 6 implies that either $w_2, w_4 \in K$ or $w_2, w_4 \in I$, which is a contradiction.

Case 2.2. The $AP_6$ with $aw_2$ as its base and $bw_3$ as its ceiling has vertex ordering $a, w_2, c, w_4, w_5, d$. Since no two literals among $\{\ell_1, \ell_2, \ell_3\}$ are one the negation of the other, it follows that no pair among the edges of this $AP_6$ is in conflict. Thus Lemma 2 implies in particular that the edge $w_5w_4$ exists in $H$ and that $ad | w_5w_4$. Thus, since $\ell_{aw_2} = \ell_3$, it follows that $\ell_{w_5w_4} = \ell_3$. Recall now that we initially assumed that the vertices $w_1, w_2, w_3, w_4, w_5, w_6$ build an $AP_6$ in $H$, which has the literals $\ell_1, \ell_2, \ell_4$ on its edges (in this order). Similarly, Lemma 2 implies for this $AP_6$ that $w_6w_1 | w_3w_2$. Thus, since $\ell_{w_6w_1} = \ell_4$, it follows that $\ell_{w_3w_2} = \ell_4$. That is, $\ell_{w_5w_4} = \ell_3 = \ell_4$, and thus $\ell_3 = \ell_4$. This completes the proof of the lemma.

We are now ready to prove the three main structural properties of the formula $\phi_1 \land \phi_2$ in Lemmas 16 and 17 respectively. The proof of the next lemma is a based on the results of 21.

Lemma 16. Let $\alpha$ and $\beta$ be two clauses of $\phi_1$. If $\alpha$ and $\beta$ share at least one variable, then $\{\alpha, \beta\} = \{\beta, \overline{\beta}\}$.

Proof. In Theorem 3.2 of 21, the authors consider an arbitrary graph $G$ and its conflict graph $G^*$, which is bipartite. For every edge $e$ of $G$, denote by $C^*(e)$ the connected component of $G^*$ in which the vertex
exists, since no two literals among $\{\alpha, \beta\}$. Furthermore, let without loss of generality $\ell \in \{v_2v_5, e' = v_2v_5\}$, and $e'' = v_3v_4$. This $AP_5$ has the non-edge $v_1v_3$ as its base and the edge $v_3v_6$ as its front, cf. Definition 4. Note that either $\alpha = (e_1 \lor e_2 \lor e_3)$ and $\tau = (e_2 \lor e_2 \lor e_3)$, or $\alpha = (e_2 \lor e_2 \lor e_3)$, and $\tau = (e_2 \lor e_2 \lor e_3)$. Assume without loss of generality that $\alpha = (e_2 \lor e_2 \lor e_3)$ and $\tau = (e_2 \lor e_2 \lor e_3)$. Recall by our assumption that $\alpha$ shares at least one literal with at least one of the dual clauses $\{\beta, \overline{\beta}\}$. Assume without loss of generality that $\alpha$ shares at least one literal with $\beta$ (the case where $\alpha$ shares at least one literal with $\overline{\beta}$ can be handled in exactly the same way). Furthermore, let without loss of generality $\ell_\ell'$. be the common literal of $\alpha$ and $\beta$, i.e. let $\beta = (e_2 \lor \ell \lor e_3)$. Since $\alpha$ is a clause of $\phi_1$, it follows by the construction of $\phi_1$ that no two literals among $\{\ell, \ell', \ell_\ell', \ell_{\overline{\ell}}\}$ are one the negation of the other (cf. lines 3 of Algorithm 2). Similarly no two literals among $\{\ell, \ell', \ell_{\overline{\ell}}, \ell_{\overline{\ell}}\}$ are one the negation of the other, since $\beta$ is a clause of $\phi_1$. Consider now an arbitrary truth assignment $\tau$ of the variables $x_1, x_2, \ldots, x_k$, such that $\alpha = 0$ in $\tau$, i.e. $\ell = \ell_\ell' = \ell_{\overline{\ell}} = 0$ in $\tau$. Note that such an assignment exists, since no two literals among $\{\ell, \ell', \ell_{\overline{\ell}}\}$ are one the negation of the other. Let $\chi$ be the 2-coloring of the vertices of $H^*$ (i.e. of the edges of $H$) that corresponds to the truth assignment $\tau$, cf. Observation 3. Since $\alpha = 0$ in the truth assignment $\tau$, Observation 3 implies that the $AP_5$ on the vertices $v_1, \ldots, v_6$ is monochromatic in the edge-coloring $\chi$ of $H$. Then, due to the results of 21, if we flip in $\chi$ the colors of all edges of $C^*(v_3v_6)$, in the new edge coloring $\chi'$ of $H$ no edge of $C^*(v_3v_6)$ participates in a monochromatic $AP_5$. Let $\tau'$ be the truth assignment that corresponds to this new coloring $\chi'$ (cf. Observation 4). Then $\tau$ and $\tau'$ coincide on all variables except the variable of the component $C^*(v_3v_6)$ of $H^*$. Note that the color of $e'' = v_2v_6$ has been flipped in the transition from $\chi'$ to $\chi$, since $e'' \in C^*(v_3v_6)$, and thus $\ell_\ell' = 1$ in $\chi'$. Furthermore, since no edge of $C^*(v_3v_6)$ participates in a monochromatic $AP_5$ in $\chi'$, it follows that both clauses $\beta = (e_2 \lor \ell \lor e_3)$ and $\overline{\beta} = (e_2 \lor e_2 \lor e_3)$ are satisfied in $\tau'$, i.e. $\beta = 1$ and $\overline{\beta} = 1$ in $\tau'$, since both $\beta$ and $\overline{\beta}$ include one of the literals $\{\ell, \ell_\ell\}$. We will now prove that $\{\ell_p, \ell_q\} \cap \{\ell, \ell_{\overline{\ell}}\} = \emptyset$. Assume otherwise that $\{\ell_p, \ell_q\} \cap \{\ell, \ell_{\overline{\ell}}\} = \emptyset$. We distinguish the following three cases.

**Case 1.** $\ell_p \neq \ell$ and $\ell_q \neq \ell_{\overline{\ell}}$. Then, since no two literals among $\{\ell, \ell_{\overline{\ell}}, \ell_{\overline{\ell}}\}$ are one the negation of the other, it follows that $\ell_p, \ell_q \notin \{\ell, \ell_{\overline{\ell}}\}$. Therefore the values of $\ell_p$ and $\ell_q$ remain the same in both assignments $\tau$ and $\tau'$. Since $\tau$ has been assumed to be an arbitrary assignment such that $\ell_\ell' = \ell_{\overline{\ell}} = 0$ in $\tau$, we can choose the assignment $\tau$ to be such that $\ell_p = \ell_q = 1$ in $\tau$. Since the value of $\ell_{\overline{\ell}}$ changes to 1 in $\tau'$, while the values of $\ell_p$ and $\ell_q$ are the same in both $\tau$ and $\tau'$, it follows that $\ell_{\overline{\ell}} = \ell_p = \ell_q = 1$ in $\tau'$, and thus $\overline{\beta} = 0$ in $\tau'$, which is a contradiction.

**Case 2.** Exactly one of $\ell_p, \ell_q$ is equal to $\ell_{\overline{\ell}}$. Let without loss of generality $\ell_p = \ell_{\overline{\ell}}$ and $\ell_q \neq \ell_{\overline{\ell}}$, i.e. $\ell_q \notin \{\ell, \ell_{\overline{\ell}}\}$. Therefore the value of $\ell_q$ remains the same in both assignments $\tau$ and $\tau'$. Since $\tau$ has been
assumed to be an arbitrary assignment such that $\ell_e = \ell_{e'} = \ell_{e''} = 0$ in $\tau$, we can choose the assignment $\tau$ to be such that $\ell_\tau = 1$ in $\tau$. Since the value of $\ell_p = \ell_e$ changes to 1 in $\tau'$, while the value of $\ell_q = 0$ is the same in both $\tau$ and $\tau'$, it follows that $\ell_{e'} = \ell_p = \ell_q = 1$ in $\tau'$, and thus $\beta = 0$ in $\tau'$, which is a contradiction.

**Case 3.** $\ell_p = \ell_q = \ell_e$. Then $\beta = (\ell_e \lor \ell_p \lor \ell_q) = (\ell_e)$ and $\beta = (\ell_{e'} \lor \ell_p \lor \ell_q) = (\ell_{e'})$, and thus it is not possible that both $\beta = 1$ and $\beta = 1$ in $\tau'$, which is again a contradiction.

Therefore $\{\ell_p, \ell_q\} \cap \{\ell_e, \ell_{e'}\} \neq \emptyset$. Thus, since the clauses $\alpha$ and $\beta$ share also the literal $\ell_{e'}$, it follows that $\alpha$ and $\beta$ share at least two literals. Therefore $\alpha = \beta$ by Lemma 15. This is a contradiction, since we assumed that $\beta \notin \{\alpha, \overline{\alpha}\}$. Therefore $\beta \notin \{\alpha, \overline{\alpha}\}$, and thus $\{\alpha, \overline{\alpha}\} = \{\beta, \overline{\beta}\}$. This completes the proof of the lemma. \qed

**Definition 11.** The clauses of $\phi_2$ are partitioned into the subformulas $\phi'_2, \phi''_2$, such that $\phi'_2$ contains all tautologies of $\phi_2$ and all clauses of $\phi_2$ in which at least one literal corresponds to an uncommitted edge, while $\phi''_2$ contains all the remaining clauses of $\phi_2$.

**Lemma 17.** Let $\{e_1, e_2, e_3\}$ be the three edges of an $AC_0$ in $H$, which has clauses in $\phi_1$. Let $e$ be an edge of $H$ such that $(\ell_e \lor \ell_{e_1})$ is a clause in $\phi'_2$. Then $\phi''_2$ contains also at least one of the clauses $\{\ell_e \lor \ell_{e_1}, \ell_e \lor \ell_{e_2}, \ell_e \lor \ell_{e_3}\}$.

**Proof.** Recall that $H$ is the associated split graph of $\bar{G}$, where $\bar{G}$ is the bipartite complement $\bar{C}(P)$ of the domination bipartite graph $C(P)$ of the partial order $P$, cf. Definitions 5 and 8. For the purposes of the proof denote $C(P) = (U \cup V, E)$, where $U = \{u_1, u_2, \ldots, u_n\}$, $V = \{v_1, v_2, \ldots, v_n\}$; then $u_i v_j \in E$ if and only if $u_i <_P v_j$ (cf. Definition 8). Furthermore denote $\bar{G} = (U \cup V, \bar{E})$ for the bipartite complement $\bar{G} = \bar{C}(P)$ of $C(P)$. Then $H = (U \cup V, E_H)$, where $E_H = \bar{E} \cup (V \times V)$ (cf. Definition 5). Moreover let $E_0 = \{u_i v_j \mid 1 \leq i \leq n\}$ and observe that $E_0 \subseteq \bar{E} \subseteq E_H$. Since edges of $E_0$ correspond to non-edges of $E$, it follows by the definition of $E_0$ that $u_i v_j \notin E$ if and only if $u_i <_P v_j$. That is, the non-edges of $E$ between vertices of $U$ and vertices of $V$ follow the transitivity of the partial order $P$.

Since $H$ is a split graph, Lemma 7 implies that the $AC_0$ of $H$ is an $AP_0$, i.e. an alternating path of length 6 (cf. Figure 1(b)). Furthermore, since $V$ induces a clique and $U$ induces an independent set in $H$, Lemma 6 implies that the vertices of $AP_0$ in $H$ belong alternately to $U$ and to $V$. Thus let $u_i v_j, u_p v_q, u_r v_t$ be the vertices of the $AP_0$ (where $u_i v_j \notin E_H$ according to our notation, cf. Definition 3). Without loss of generality let $e_1 = u_p v_j, e_2 = u_r v_q$, and $e_3 = u_t v_s$. Since the $AP_0$ has clauses in $\phi_1$ by assumption, note by the construction of $\phi_1$ (cf. Section 5) that no two literals among $\{\ell_{e_1}, \ell_{e_2}, \ell_{e_3}\}$ are one the negation of the other. Therefore no pair among the edges $\{e_1, e_2, e_3\}$ is in conflict, and thus Lemma 2 implies that the edges $u_p v_q, u_r v_t$ exist in $H$ and $e_2 = u_r v_q \lor u_p v_q, e_1 = u_p v_q \lor u_i v_j, e_3 = u_t v_s \lor u_r v_t$. Therefore $\ell_{u_p v_q} = \ell_{e_1}, \ell_{u_r v_t} = \ell_{e_2},$ and $\ell_{u_r v_t} = \ell_{e_3}.$

Since $e_1 = u_p v_q$, and $(\ell_e \lor \ell_{e_1})$ is a clause of $\phi'_2$ (and thus also of $\phi_2$), it follows by the construction of $\phi_2$ (cf. Section 5) that either $e = u_p v_q$ or $e = u_r v_t$ for some index $a \in \{1, 2, \ldots, n\}$.

**Case 1.** $e = u_p v_q$. Denote $E'_H = E_H \setminus E_0$. Then it follows by the construction of $\phi_2$ that $u_p v_q \notin E'_H$ and thus either $u_p v_q \notin E_H$ or $u_p v_q \in E_0$. Furthermore, since $(\ell_e \lor \ell_{e_1})$ is a clause of $\phi''_2$ by assumption, it follows by Definition 11 that $e$ is a committed edge in $H$. That is, there exists an edge $e' = u_p v_q$ such that $e' \notin E_H$. Therefore $e' \notin E_H$. Furthermore, since $u_p v_q \notin E_H$, it follows that $u_p <_P v_q$ and $u_q <_P u_p$. Therefore $u_q <_P u_p$, since $P$ is a partial order, and thus $u_q v_p \in E_H$.

Note that either $a = j$ or $a \neq j$ (cf. Figures 3(a) and 3(b)) respectively. We distinguish now these two cases, which are illustrated in Figures 4(a) and 4(b) respectively. In these figures, the edges $e_1, e_2, e_3$ of the $AP_0$, as well as the edges $e$ and $e'$, are drawn by thick lines and all other edges are drawn by thin lines, while non-edges are illustrated with dashed lines.

**Case 1.1.** $a = j$ (cf. Figure 4(a)). Suppose that $u_i v_s \in E_H$. Then $u_i v_s \lvert u_i v_j = u_p v_q$, since $u_p v_q \notin E_H$. Thus the edge $u_i v_j \in E_0$ is committed, which is a contradiction by Lemma 12. Therefore $u_i v_s \notin E_H$. Suppose now that $u_i v_s \notin E_H$. Then $u_i v_s \lvert u_p v_q$, since $u_p v_q \notin E_H$. Thus the edge $u_p v_q \in E_0$ is committed, which is a contradiction by Lemma 12. Therefore $u_i v_s \in E_H$. Furthermore $u_p v_q \lvert u_i v_s, u_p v_q \not\in E_H$, and thus $\ell_{u_i v_s} = \ell_{u_p v_q}$, Therefore, since $\ell_{u_i v_q} = \ell_{e_1}$, it follows that $\ell_{u_i v_s} = \ell_{e_1}$.

Suppose that $a = j$ and $P$ is a partial order, it follows that $u_i <_P u_q$. Then, since $u_i v_j \notin E_H$, it follows that $u_i <_P u_j$. Therefore, since $a = j$ and $P$ is a partial order, it follows that $u_i <_P u_q$, and thus $u_i v_j \notin E_H$, which is a contradiction. Therefore $u_i v_s \in E_H$. Furthermore $u_i v_s \lvert u_i v_q$, since $u_i v_q \notin E_H$, and thus $\ell_{u_i v_s} = \ell_{u_i v_q}$. Therefore, since $\ell_{u_i v_q} = \ell_{e_1}$, it follows that $\ell_{u_i v_s} = \ell_{e_1}$.

Since $u_i v_s, u_r v_t \notin E_H$, it follows that $e' = u_r v_t \lvert u_p v_q$, and thus $\ell_e = \ell_{u_r v_t}, \ell_{u_r v_t} = \ell_{u_i v_s}$, Therefore, since $\ell_{u_i v_s} = \ell_{e_1}$, it follows that $\ell_e = \ell_{e_1}$. Finally, since $e' \lvert e$, it follows that $\ell_e = \ell_{e'},$ and thus $\ell_e = \ell_{e_1}$. Therefore the clause $(\ell_e \lor \ell_{e_1})$ of $\phi''_2$ is a tautology, which is a contradiction by Definition 11.

21
Case 1.2. $a \neq j$ (cf. Figure 4(b)). Then $u_av_j \notin E_H$. Thus, since $u_av_j \notin E'_H$, it follows that $u_av_j \notin E_H$. Suppose that $u_av_s \in E_H$ (cf. Figure 4(b)). Then $u_av_s || u_av_j$, since $u_av_j, u_av_s \notin E_H$, and thus $\ell_{u_av_s} = \ell_{u_av_j}$. Therefore, since $\ell_{u_av_j} = \ell_{e_3}$, it follows that $\ell_{u_av_s} = \ell_{e_3}$. Suppose that $u_av_q \notin E_H$. Then $u_av_q || u_av_s$, since $u_av_s, u_av_q \notin E_H$. Therefore $\ell_{u_av_q} = \ell_{e_3}$, since $\ell_{u_av_q} = \ell_{e_2}$. Thus, since $\ell_{u_av_s} = \ell_{e_3}$, it follows that $\ell_{e_3} = \ell_{e_2}$. This is a contradiction, since no two literals among $\{e_1, e_2, e_3\}$ are one the negation of the other. Therefore $u_av_q \notin E_H$. Moreover, since $u_av_j, u_av_q \notin E_H$, it follows that $\ell_{u_av_q} = \ell_{e_1}$, and thus $\ell_{u_av_s} = \ell_{e_2}$. Furthermore, $u_av_q || u_av_c = e'$, since $u_av_q, u_av_c \notin E_H$. Therefore $\ell_{u_av_q} = \ell_{e'}$. Thus $\ell_{u_av_s} = \ell_{e'}$, since $\ell_{e'} = \ell_{e_2}$. Furthermore, since $\ell_{u_av_s} = \ell_{e_3}$ and $\ell_{u_av_q} = \ell_{e'}$, it follows that $\ell_{e'} = \ell_{e_3}$. Therefore the clause $(\ell_{e'} \lor \ell_{e_3})$ of $\phi_2'$ is a tautology, which is a contradiction by Definition 11.

Therefore $u_av_s \notin E_H$. Then also $u_av_q \notin E_H$, and thus $\phi_2$ has the clause $(\ell_{u_av_s} \lor \ell_{u_av_q}) = (\ell_{e'} \lor \ell_{e_3})$, since $e = u_av_p$ and $\ell_{u_av_s} = \ell_{e_2}$. Furthermore, since both $e$ and $u_pv_s$ are committed in $H$ (as $e' || e$ and $u_pv_q || u_pv_s$), the clause $(\ell_{e'} \lor \ell_{e_3})$ belongs to $\phi_2'$ by Definition 11.

Fig. 4. (a) The Case 1.1 and (b) the Case 1.2 in the proof of Lemma 17

Case 2. $e = u_jv_a$. This case is exactly symmetric to Case 1. To see this, imagine exchanging the roles of $U$ and $V$, i.e. $U$ induces now a clique (instead of an independent set) and $V$ induces an independent set (instead of a clique) in $H$. Imagine also flipping the lines $L_1$ and $L_2$ in Figure 4 (i.e. $L_2$ comes now above $L_1$), such that the vertices of $U$ and $V$ still lie on the lines $L_1$ and $L_2$, respectively. Similarly to Cases 1.1 and 1.2, we distinguish the cases $a = p$ (Case 2.1) and $a \neq p$ (Case 2.2), respectively. Then, Case 2.1 leads to a contradiction (similarly to Case 1.1), and Case 2.2 implies that the clause $(\ell_{e'} \lor \ell_{e_3})$ belongs to $\phi_2'$ (instead of the clause $(\ell_{e'} \lor \ell_{e_2})$ in Case 1.2).

Summarizing, if $e = u_av_p$ then $\phi_2'$ includes the clause $(\ell_{e'} \lor \ell_{e_2})$, while if $e = u_jv_a$ then $\phi_2'$ includes the clause $(\ell_{e'} \lor \ell_{e_3})$. This completes the proof of the lemma. □

Lemma 18. Let $\{e_1, e_2, e_3\}$ be the three edges of an $AC_6$ in $H$, which has clauses in $\phi_1$. Let $e$ be an edge of $H$ such that $(\ell_{e} \lor \ell_{e_1})$ is a clause in $\phi_2'$. Then $\phi_2'$ contains at least one of the clauses $(\ell_{e} \lor \ell_{e_2}), (\ell_{e} \lor \ell_{e_3})$.

Proof. Since $H$ is a split graph, Lemma 7 implies that the $AC_6$ of $H$ is an $AP_6$, i.e. an alternating path of length 6 (cf. Figure 1(b)). Using the notation of Lemma 17 denote by $V$ and $U$ the clique and the independent set of $H$, respectively. Then the vertices of the $AP_6$ in $H$ belong alternately to $V$ and to $V$ by Lemma 6. That is, $u_i, v_j, u_p, v_q, u_r, v_s$ are the vertices of the $AP_6$ in this order, for some vertices $u_i, u_p, u_r \in U$ and $v_j, v_q, v_s \in V$ (where $u_i, v_j, u_p, v_q, u_r, v_s \notin E_H$ according to our notation, cf. Definition 3). Without loss of generality let $e_1 = u_pv_s$, $e_2 = u_qv_p$, and $e_3 = u_rv_s$. Then, similarly to the preamble of the proof of Lemma 17, it follows that the edges $e'_1 = u_qv_q$, $e'_2 = u_pv_p$, and $e'_3 = u_rv_r$ exist in $H$ and $e_1 = u_pv_q || u_qv_q = e'_1$, $e_2 = u_pv_p || u_rv_p = e'_2$, and $e_3 = u_rv_s || u_rv_s = e'_3$. Therefore $\ell_{e'_1} = \ell_{e_1}$, $\ell_{e'_2} = \ell_{e_2}$, and $\ell_{e'_3} = \ell_{e_3}$.

Since $u_pv_q, u_rv_p, u_rv_s \notin E_H$, it follows that the vertices $u_i, v_j, u_p, v_q, u_r, v_s$ (in this order) build an $AC_6$ in $H$, where $\{e'_1, e'_2, e'_3\}$ are its three edges. Therefore, by applying Lemma 7 on this new $AC_6$, it follows that

22
Furthermore, since \( AC \) edges of such an \( AC \) among the three edges in the corresponding that in every clause of \( \phi \) corresponds to an uncommitted edge by Definition 11. Recall now by the construction of \( \phi \) to be the bipartite complement of \( C \) be the bipartite complement of \( C \), \( \phi \) of the sub-formula \( \phi \), i.e. \( \phi \) is a gradually mixed formula.

**6.2 The recognition algorithm**

In this section we use Corollary 3 to design an algorithm that decides satisfiability on \( \phi_1 \land \phi_2 \) in time linear to its size (cf. Theorem 8). This will enable us to combine the results of Sections 4 and 5 to recognize efficiently whether a given graph is a PI graph, or equivalently, due to Theorem 4, whether a given partial order \( P \) is the intersection of a linear order \( P_1 \) and an interval order \( P_2 \).

**Theorem 8.** \( \phi_1 \land \phi_2 \) is satisfiable if and only if \( \phi_1 \land \phi_2' \) is satisfiable. Given a satisfying truth assignment of \( \phi_1 \land \phi_2' \) we can compute a satisfying truth assignment of \( \phi_1 \land \phi_2 \) in linear time.

**Proof.** If \( \phi_1 \land \phi_2 \) is satisfiable then \( \phi_1 \land \phi_2' \) is also satisfiable as a sub-formula of \( \phi_1 \land \phi_2 \). Conversely, suppose that \( \phi_1 \land \phi_2' \) is satisfiable and let \( \tau \) be a satisfying assignment. Consider an arbitrary clause \( \gamma = (\ell_{e_1} \lor \ell_{e_2}) \) of the sub-formula \( \phi_2' \) of \( \phi_2 \), cf. Definition 11. If \( \gamma \) is a tautology then \( \gamma \) is satisfied by any truth assignment of \( \phi \), and thus also by \( \tau \). Assume now that \( \gamma \) is not a tautology. Then at least one of its literals \( \{\ell_{e_1}, \ell_{e_2}\} \) corresponds to an uncommitted edge by Definition 11. Recall now by the construction of \( \phi_1 \) (cf. Section 5) that in every clause of \( \phi_1 \), no literal is the negation of another literal. Thus, for every clause of \( \phi_1 \), no pair among the three edges in the corresponding \( AC_6 \) is in conflict. Therefore Lemma 2 implies that all three edges of such an \( AC_6 \) are committed. Thus, for every literal \( \ell_e \) of \( \phi_2' \), which corresponds to an uncommitted edge \( e \), neither \( \ell_e \) nor \( \ell_{\overline{e}} \) appears in \( \phi_1 \). Furthermore recall that \( \phi_2' \) does not include any literal \( \ell_e \) of any uncommitted edge \( e \) of \( H \) by Definition 11.

Summarizing, for every literal \( \ell_e \) of \( \phi_2' \), which corresponds to an uncommitted edge \( e \), neither \( \ell_e \) nor \( \ell_{\overline{e}} \) appears in \( \phi_1 \). That is, the truth assignment \( \tau \) of \( \phi_1 \land \phi_2 \) does not assign any value to the literal \( \ell_e \). Furthermore, since \( e \) is uncommitted, no edge of \( H \) is assigned the literal \( \ell_e \). Therefore we can extend (in linear time) the truth assignment \( \tau \) to a truth assignment \( \tau' \) that satisfies both \( \phi_1 \land \phi_2' \) and \( \phi_2' \), by setting \( \ell_e = 1 \) for all uncommitted edges \( e \) of \( H \). That is, \( \tau' \) satisfies the formula \( \phi_1 \land \phi_2 \). Therefore \( \phi_1 \land \phi_2 \) is satisfiable if and only if \( \phi_1 \land \phi_2' \) is satisfiable. This completes the proof of the theorem.

**Remark.** Now we are ready to present our recognition algorithm for PI graphs (Algorithm 5). Its correctness and timing analysis is established in Theorem 9.

**Theorem 9.** Let \( G = (V,E) \) be a graph and \( \overline{G} = (V,E) \) be its complement, where \( |V| = n \) and \( |E| = m \). Then Algorithm 5 constructs in \( O(n^2m) \) time a PI representation of \( G \), or it announces that \( G \) is not a PI graph.

**Proof.** If the given graph \( G \) is a trapezoid graph, then Algorithm 5 computes in line 2 a partial order \( P \) of its complement \( \overline{G} \). Otherwise, if \( G \) is not a trapezoid graph, then clearly it is also not a PI graph, and thus the algorithm correctly announces in line 3 that \( G \) is not a PI graph.

Let \( C(P) \) be the domination bipartite graph of the partial order \( P \) (cf. Definition 8), and let \( \tilde{G} = \tilde{C}(P) \) be the bipartite complement of \( C(P) \), which are computed in lines 4 and 5 of Algorithm 5 respectively. Furthermore let \( H \) be the associated split graph of \( \tilde{G} \) (cf. Definition 5) and \( H^* \) be the conflict graph of \( H \) (cf. Definition 7), which are computed in lines 6 and 7 of Algorithm 5 respectively. If \( H^* \) is not bipartite, i.e. if \( \chi(H^*) > 2 \), then \( \tilde{G} \) is not linear-interval coverable by Lemma 11 and thus \( G \) is not a PI graph by
Algorithm 5 Recognition of PI graphs

Input: A graph $G = (V, E)$
Output: A PI representation $R$ of $G$, or the announcement that $G$ is not a PI graph

1. if $G$ is a trapezoid graph then
2. Compute a partial order $P$ of the complement $\overline{G}$
3. else return “$G$ is not a PI graph”
4. Compute the domination bipartite graph $C(P)$ from $P$
5. $\overline{G} \leftarrow \overline{C(P)}$
6. Compute the associated split graph $H$ of $\overline{G}$
7. Compute the conflict graph $H^*$ of $H$
8. if $H^*$ is bipartite then
9. Compute a 2-coloring $\chi_0$ of the vertices of $H^*$
10. Compute the formulas $\phi_1$ and $\phi_2$
11. if $\phi_1 \wedge \phi_2$ is satisfiable then
12. Compute a satisfying truth assignment $\tau$ of $\phi_1 \wedge \phi_2$ by Theorem 8
13. Compute from $\tau$ a linear-order cover of $G$ by Algorithm 4
14. Compute a PI representation $R$ of $G$ by Algorithm 5
15. else
16. return “$G$ is not a PI graph”
17. else
18. return “$G$ is not a PI graph”
19. return $R$

Corollary 2 Therefore Algorithm 5 correctly announces in line 18 that $G$ is not a PI graph if $H^*$ is not bipartite.

Suppose now that $H^*$ is bipartite, i.e. $\chi(H^*) \leq 2$. Let $\chi_0$ be a 2-coloring of the vertices of $H^*$, which is computed in line 9 of Algorithm 5. Furthermore let $\phi_1$ and $\phi_2$ be the Boolean formulas that can be computed by Algorithms 2 and 3 respectively (cf. line 10 of Algorithm 5). If the formula $\phi_1 \wedge \phi_2$ is not satisfiable, then $\overline{G}$ is not linear-interval coverable by Theorem 7, and thus $G$ is not a PI graph by Corollary 2. Therefore Algorithm 5 correctly announces in line 18 that $G$ is not a PI graph if $\phi_1 \wedge \phi_2$ is not satisfiable.

Suppose now that $\phi_1 \wedge \phi_2$ is satisfiable, and let $\tau$ be a satisfying truth assignment of $\phi_1 \wedge \phi_2$, as it is computed in line 12 of Algorithm 5. Then $\overline{G}$ is linear-interval coverable by Theorem 7, and thus $G$ is a PI graph by Corollary 2. Furthermore, given $\tau$, we can compute a linear-interval cover of $G$ using Algorithm 3 (cf. line 13 of Algorithm 5). Finally, given this linear-interval cover of $G$, we can compute a PI representation $R$ of $G$ using Algorithm 5 (cf. line 14 of Algorithm 5). Thus, if $\phi_1 \wedge \phi_2$ is satisfiable, Algorithm 5 correctly returns $R$ in line 19.

Time complexity. First note that the complement $\overline{G}$ of $G$ can be computed in $O(n^2)$ time, since both $G$ and $\overline{G}$ have $n$ vertices. Furthermore, using the algorithm of 15 we can decide in $O(n^2)$ time whether $G$ is a trapezoid graph, and within the same time bound we can compute a trapezoid representation of $G$, if it exists. Suppose in the following that $G$ is a trapezoid graph. Then we can then compute in $O(n^2)$ time a partial order $P$ of the complement $\overline{G}$ of $G$ as follows: $u <_P v$ if and only if the trapezoid for vertex $u$ lies entirely to the left of the trapezoid for vertex $v$ in this trapezoid representation of $G$. Therefore, lines 13-15 of Algorithm 5 can be executed in $O(n^2)$ time in total. Note that we choose to compute the partial order $P$ using the trapezoid graph recognition algorithm of 15, in order to achieve the $O(n^2)$ time bound. Alternatively we could solve the transitive orientation problem on $\overline{G}$ using the standard forcing algorithm with $O(nm)$ running time (note that $m$ is the number of edges of $\overline{G}$).

Denote $\hat{G} = (U, V, \hat{E})$, where $U = \{u_1, u_2, \ldots, u_n\}$ and $V = \{v_1, v_2, \ldots, v_n\}$. Furthermore denote $E_0 = \{u_i v_i \mid 1 \leq i \leq n\}$. Then $H = (U, V, E_H)$, where $E_H = \hat{E} \cup (V \times V)$ by Definition 9. Since $C(P)$ and $H$ have $2n$ vertices each, each of the lines 13-15 of Algorithm 5 can be computed by a straightforward implementation in $O(n^2)$ time. Note that the partial order $P$ has $m$ pairs of comparable elements, since the complement $\overline{G}$ of $G$ has $m$ edges. Therefore the domination bipartite graph $C(P)$ of $P$ has $m$ edges (cf. Definition 8), and thus its bipartite complement $\overline{G} = \overline{C(P)}$ has $|\overline{E}| = n^2 - m$ edges.

Consider a pair $\{e, e'\}$ of edges of $H$ that are in conflict, i.e. $e \parallel e'$ in $H$. Then $e, e' \notin V \times V$ by Observation 3, since $H$ is a split graph and $V$ induces a clique in $H$. Therefore both $e$ and $e'$ are edges of $\hat{G}$, i.e. $e, e' \in \hat{E}$, and thus $e = u_i v_j$ and $e' = u_p v_q$ for some indices $i, j, p, q \in \{1, 2, \ldots, n\}$. Furthermore, since $e$ and $e'$ are in
conflict, it follows that \(u, v_j, u_p, v_j \notin E\). That is, every pair of conflicting edges in \(H\) corresponds to exactly one pair \(\{u, v_q, u_p, v_j\}\) of non-edges of \(\hat{G} = \hat{C}(P)\). Equivalently, every edge in the conflict graph \(H^*\) of \(H\) corresponds to exactly one pair of edges of \(C(P)\). Since \(C(P)\) has \(m\) edges, it follows that the conflict graph \(H^*\) has at most \(O(n^2)\) edges. Furthermore note that the conflict graph \(H^*\) has \((\frac{n}{2}) + |\hat{E}| = O(n^2)\) vertices, since \(H\) has \((\frac{n}{2}) + |\hat{E}|\) edges. Therefore the conflict graph \(H^*\) can be computed in \(O(n^2 + m^2)\) time (cf. line 7 of Algorithm 5).

Note now that the size of the size of \(H^*\), we can check whether \(H^*\) is bipartite, and we can compute a 2-coloring \(\chi_0\) of the vertices of \(H^*\), if one exists. Therefore lines 30 of Algorithm 5 can be executed in \(O(n^2 + m^2)\) time. Furthermore, in time linear to the size of \(H^*\), i.e. in \(O(n^2 + m^2)\) time, we can compute the connected components \(C_1, C_2, \ldots, C_k\) of \(H^*\). Then, having already computed the 2-coloring \(\chi_0\) and the connected components \(C_1, C_2, \ldots, C_k\) of \(H^*\), we can assign to every edge \(e\) of \(H\) the literal \(\ell_e \in \{x_i, \overline{x_i} \mid 1 \leq i \leq k\}\) (cf. Section 5). This can be done in \(O(n^2)\) time, since \(H\) has \((\frac{n}{2}) + |\hat{E}| = O(n^2)\) edges.

Now we bound the size of the formulas \(\phi_1\) and \(\phi_2\) that are computed by Algorithms 2 and 3, respectively. Regarding the size of \(\phi_2\), note that, by the construction of \(\phi_2\), if \((\ell_e \lor \ell_{e'}\) is a clause of \(\phi_2\), then \(e = u, v_j, e' = u, v_j,\) and \(u, v_j \notin E_H \setminus E_0\) for some indices \(i, j, t \in \{1, 2, \ldots, n\}\). That is, for every index \(t \in \{1, 2, \ldots, n\}\) and every pair \((i, j)\) of indices in the set \(\{(i, j) \mid i = j \lor u, v_j \notin E_H\}\), the formula \(\phi_2\) has at most one clause. Note that every pair \((i, j)\) of the set \(\{(i, j) \mid u, v_j \notin E_H\}\) corresponds to exactly one edge \(u, v_j\) of the bipartite graph \(C(P)\). Thus, since \(C(P)\) has \(m\) edges, it follows that \(|\{(i, j) \mid i = j \lor u, v_j \notin E_H\}| \leq n + m\).

Regarding the size of \(\phi_1\), recall first that every connected component \(C_i\) of the conflict graph \(H^*\) has been assigned exactly one Boolean variable \(x_i\), where \(i \in \{1, 2, \ldots, k\}\). Furthermore recall that every edge \(e\) of \(H\) has been assigned a literal \(\ell_e \in \{x_i, \overline{x_i} \mid 1 \leq i \leq k\}\). Therefore, since every clause of \(\phi_1\) appears only once in \(\phi_1\) (cf. lines 5 of Algorithm 5), it follows by the construction of \(\phi_1\) and by Lemma 16 that \(\phi_1\) has at most \(2\frac{n}{k}\) clauses. Furthermore note that \(k = O(n^2)\), since \(H^*\) has \(O(n^2)\) vertices. Thus \(\phi_1\) has at most \(O(n^2)\) clauses.

Claim 2 The following two statements are equivalent:

(a) the formula \(\phi_1\) contains the clauses \(\alpha = (\ell_e \lor \ell_{e'} \lor \ell_{e''})\) and \(\alpha' = (\overline{\ell_e} \lor \overline{\ell_{e'}} \lor \overline{\ell_{e''}})\),

(b) there exist four distinct vertices \(a, b, c, d\) in \(H\), such that:

- \(ab \notin E_H\) and \(bc, cd, da \in E_H\),
- either \(a, c \in U\) and \(b, d \in V\), or \(a, c \in V\) and \(b, d \in U\),
- the edges \(bc, cd, da\) are committed in \(H\),
- \(\ell_{bc} = \ell_e, \ell_{cd} = \ell_{e'}, \ell_{da} = \ell_{e''}\), and
- \(\ell_e \neq \overline{\ell_{e'}}, \ell_{e'} \neq \overline{\ell_{e''}}, \ell_{e''} \neq \overline{\ell_e}\).

Proof of Claim 2. ((a) \(\Rightarrow\) (b)) Consider first a pair of clauses \(\alpha = (\ell_e \lor \ell_{e'} \lor \ell_{e''})\) and \(\alpha' = (\overline{\ell_e} \lor \overline{\ell_{e'}} \lor \overline{\ell_{e''}})\) in \(\phi_1\). These clauses correspond to an \(AC_6\) on the edges \(\{e, e', e''\}\) of \(H\) by the construction of \(\phi_1\). Furthermore, since \(H\) is a split graph, Lemma 7 implies that this \(AC_6\) of \(H\) is an \(AP_6\), an alternating path of length 6 (cf. Figure 1(b)). Let \(w_1, w_2, w_3, w_4, w_5, w_6\) be the vertices of this \(AP_6\), such that \(c = w_2w_3, e' = w_4w_5,\) and \(e'' = w_6w_1\) (note that there always exists an enumeration of the vertices of the \(AP_6\) such that the edges \(e, e', e''\) are met in this order on the \(AP_6\)). Then, since \(V\) induces a clique in \(H\) and \(U\) induces an independent set in \(H\), Lemma 3 implies that either \(w_1, w_3, w_5 \in U\) and \(w_2, w_4, w_6 \in V\), or \(w_1, w_3, w_5 \in V\) and \(w_2, w_4, w_6 \in U\). Since \(\ell_e \neq \ell_{e'}, \ell_{e'} \neq \ell_{e''}\), and \(\ell_{e''} \neq \overline{\ell_e}\) (cf. line 3 of Algorithm 3), it follows that no pair among the edges \(\{e, e', e''\}\) is in conflict in \(H\). Therefore Lemma 2 implies that the edges \(w_3w_6, w_4w_1, w_5w_2\) exist in \(H\) and \(e'\mid w_3w_6, e\mid w_4w_1, e''\mid w_5w_2\). Thus all six edges \(\{e, e', e''\}\) are committed. Furthermore \(\ell_{w_3w_6} = \ell_e, \ell_{w_4w_1} = \ell_{e'}, \ell_{w_5w_2} = \ell_{e''}\). Thus the vertices \(a = w_1, b = w_2, c = w_3,\) and \(d = w_6\) of \(H\) satisfy the conditions of the part (b) of the claim.

((b) \(\Rightarrow\) (a)) Conversely, consider four vertices \(a, b, c, d\) in \(H\), as specified in the part (b) of the claim. Then, since the edge \(cd\) is committed, there exists an edge \(pq \in E_H\) such that \(pc, qd \notin E_H\), and thus \(cd\mid pq\). Then \(\ell_{pq} = \ell_e\). Therefore, since \(\ell_{cd} = \ell_{e''}\), it follows that \(\ell_{pq} = \ell_{e'}\). Thus there exists an \(AC_6\) in \(H\) on the vertices \(a, b, c, p, q, d\), where \(\ell_{bc} = \ell_e, \ell_{pq} = \ell_{e'},\) and \(\ell_{da} = \ell_{e''}\). Furthermore, since \(\ell_e \neq \ell_{e'}, \ell_{e'} \neq \ell_{e''}\), and \(\ell_{e''} \neq \overline{\ell_e}\) by assumption, it follows by the construction of \(\phi_1\) (cf. Algorithm 2) that \(\phi_1\) contains the clauses \(\alpha = (\ell_e \lor \ell_{e'} \lor \ell_{e''})\) and \(\alpha' = (\overline{\ell_e} \lor \overline{\ell_{e'}} \lor \overline{\ell_{e''}})\).

Now, due to Claim 2, we can implement Algorithm 2 for the computation of \(\phi_1\) in time \(O(n^2 + m^2)\) as follows. Recall first that \(C(P)\) has \(m\) edges. We iterate for every edge \(u, v_j\) of \(C(P)\), i.e. for every non-edge \(u, v_j \notin E_H\) of \(H\). For every such \(u, v_j\), we mark all vertices in the sets \(A\) and \(B\), where \(A = \{e \in V \mid u, v \in E_H\}\) and \(u, v\) is committed in \(H\) and
Let $B = \{ u \in U \mid uv \in E_H \text{ and } uv \text{ is committed in } H \}$. Then we scan through the adjacency lists of all vertices in $A$ to discover a pair of vertices $v \in A$ and $u \in B$ such that $uv$ is a committed edge of $H$, and $\ell_{v,u} \neq \ell_{uv}, \ell_{u,v} \neq \ell_{vu}$, and $\ell_{v,u} \neq \ell_{vu}$. Since $H$ has $O(n^2)$ edges, this scan through the adjacency lists of the vertices of $A$ can be done in $O(n^2)$ time. If we discover such an edge $uv$, then we add to $\phi_1$ the clauses $\alpha = (\ell_{v,u} \lor \ell_{u,v} \lor \ell_{vu})$ and $\alpha' = (\ell_{v,u} \lor \ell_{u,v} \lor \ell_{vu})$. Due to Claim 2, Algorithm 2 would add the same two clauses to $\phi_2$.

Due to Lemma 16, no other clause of $\phi_1$ has one of the literals $\ell_{v,u}, \ell_{u,v} = \ell_{vu}$. After we add the two clauses $\alpha$ and $\alpha'$ to $\phi_1$, we visit all edges $e$ of $H$ which correspond to the same connected component in $H^*$ with one of the edges $\{v,u,uv,vu\}$. Note that exactly these edges $e$ of $H$ have a literal $\ell_e \in \{\ell_{v,u}, \ell_{u,v} = \ell_{vu}, \ell_{u,v} = \ell_{vu}, \ell_{v,u} = \ell_{vu}\}$. We then mark all these edges $e$ such that we avoid visiting them again in any subsequent iteration during the construction of $\phi_1$. Thus we ensure that each clause appears at most once $\phi_1$ (cf. lines 12-13 of Algorithm 2). Note that we can perform all such markings of edges $e$ (or all iterations during the construction of $\phi_1$) in time linear to the size of $H^*$, i.e. in $O(n^2 + n^2)$ time. Summarizing, we need in total $O(n^2 m + m^2)$ time to compute the formula $\phi_1$. Thus, since the formula $\phi_2$ can be computed in $O(n(n + m))$ time, it follows that line 10 of Algorithm 5 can be executed in $O(n^2 m + m^2)$ time.

Now, we can test whether the formula $\phi_1 \land \phi_2$ is satisfiable in time linear to its size by Theorem 5, moreover, within the same time bound we can compute a satisfying truth assignment $\tau$ of $\phi_1 \land \phi_2$, if one exists. Thus, since $\phi_1$ has $O(n^2)$ clauses and $\phi_2$ has $O(n(n + m))$ clauses, lines 11-12 of Algorithm 5 can be executed in $O(n(n + m))$ time. Furthermore, line 13 of Algorithm 5 can be executed in $O(n^2)$ time by Theorem 5, calling Algorithm 4 as a subroutine. Finally, line 14 of Algorithm 5 can be executed in $O(n^2)$ time by Theorem 5, calling Algorithm 3 as a subroutine. Summarizing, since $m = O(n^2)$, the total running time of Algorithm 5 is $O(n^2 m)$. This completes the proof of the theorem.

Due to characterization of PI graphs in Theorem 4 using partial orders, the next theorem follows now by Theorem 5.

**Theorem 10.** Let $P = (U, R)$ be a partial order, where $|U| = n$ and $|R| = m$. Then we can decide in $O(n^2 m)$ time whether $P$ is a linear-interval order, and in this case we can compute a linear order $P_1$ and an interval order $P_2$ such that $P = P_1 \cap P_2$.

### 7 Concluding remarks

In this article we provided the first polynomial algorithm for the recognition of simple-triangle graphs, or equivalently for the recognition of linear-interval orders, solving thus a longstanding open problem. For a graph $G$ with $n$ vertices, where its complement $\overline{G}$ has $m$ edges, our $O(n^2 m)$-time algorithm either computes a simple-triangle representation of $G$, or it announces that such one does not exist. The main tool for our recognition algorithm was a new hybrid tractable subclass of $3$SAT, called the class of *gradually mixed* formulas. In addition, we introduced the notion of a *linear-interval cover* of bipartite graphs, which naturally extends the well-known notion of the chain-cover of bipartite graphs. There are two main lines for further research. The first one is to identify more “islands of tractability” for hybrid classes of SAT (and more generally of CSP), while the ultimate goal is to find a complete characterization of the hybrid classes of CSP that are tractable. The second line for further research is to resolve the complexity of the recognition for the related classes with simple-triangle graphs, such as the classes of *unit* and *proper tolerance* graphs [11] (these are subclasses of parallelogram graphs, and thus also subclasses of trapezoid graphs), *proper bitolerance* graphs [11] (they coincide with *unit bitolerance* graphs [2]), and *multitolerance* graphs [17] (they naturally generalize trapezoid graphs [17]). On the contrary, the recognition problems for the related classes of *triangle* graphs [18], *tolerance* and *bounded tolerance* (i.e. *parallelogram*) graphs [19], and *max-tolerance* graphs [14] have been already proved to be NP-complete.

### References

1. S. Benzer. On the topology of the genetic fine structure. *Proc. of the National Academy of Sciences (PNAS)*, 45:1607–1620, 1959.
2. K. P. Bogart and G. Isaak. Proper and unit bitolerance orders and graphs. *Discrete Mathematics*, 181(1-3):37–51, 1998.
3. A. Brandstädt, V. B. Le, and J. P. Spinrad. *Graph classes: a survey*. Society for Industrial and Applied Mathematics (SIAM), 1999.
4. A. V. Carrano. Establishing the order to human chromosome-specific DNA fragments. Biotechnology and the Human Genome, Plenum Press:37–50, 1988.

5. M. R. Cerioli, F. de S. Oliveira, and J. L. Szwarcfiter. Linear-interval dimension and PI orders. Electronic Notes in Discrete Mathematics, 30:111–116, 2008.

6. D. G. Corneil and P. A. Kamula. Extensions of permutation and interval graphs. In Proceedings of the 18th Southeastern Conference on Combinatorics, Graph Theory and Computing, pages 267–275, 1987.

7. I. Dagan, M. C. Golumbic, and R. Y. Pinter. Trapezoid graphs and their coloring. Discrete Applied Mathematics, 21(1):35–46, 1988.

8. S. Even, A. Itai, and A. Shamir. On the complexity of timetable and multicommodity flow problems. SIAM Journal on Computing, 5(4):691–703, 1976.

9. P. W. Goldberg, M. C. Golumbic, H. Kaplan, and R. Shamir. Four strikes against physical mapping of DNA. Journal of Computational Biology, 2(1):139–152, 1995.

10. M. C. Golumbic. Algorithmic graph theory and perfect graphs (Annals of Discrete Mathematics, Vol. 57). North-Holland Publishing Co., 2 edition, 2004.

11. M. C. Golumbic and A. N. Trenk. Tolerance Graphs. Cambridge Studies in Advanced Mathematics, 2004.

12. P. L. Hammer, T. Ibaraki, and U. N. Peled. Threshold numbers and threshold completions. Annals of Discrete Mathematics, 11:125–145, 1981.

13. T. Ibaraki and U. N. Peled. Sufficient conditions for graphs to have threshold number 2. Annals of Discrete Mathematics, 11:241–268, 1981.

14. M. Kaufmann, J. Kratochvıl, K. A. Lehmann, and A. R. Subramanian. Max-tolerance graphs as intersection graphs: cliques, cycles, and recognition. In Proceedings of the 17th annual ACM-SIAM symposium on Discrete Algorithms (SODA), pages 832–841, 2006.

15. T.-H. Ma and J. P. Spinrad. On the 2-chain subgraph cover and related problems. Journal of Algorithms, 17:251–268, 1994.

16. N. Mahadev and U. N. Peled. Threshold Graphs and Related Topics (Annals of Discrete Mathematics, Vol. 56). North-Holland Publishing Co., 1995.

17. G. B. Mertzios. An intersection model for multitolerance graphs: Efficient algorithms and hierarchy. Algorithmica. To appear; a preliminary version appeared in Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA), pages 1306–1317, 2011.

18. G. B. Mertzios. The recognition of triangle graphs. Theoretical Computer Science, 438:34–47, 2012.

19. G. B. Mertzios, I. Sau, and S. Zaks. The recognition of tolerance and bounded tolerance graphs. SIAM Journal on Computing, 40(5):1234–1257, 2011.

20. A. Parra. Triangulating multitolerance graphs. Discrete Applied Mathematics, 84(1-3):183–197, 1998.

21. T. Raschle and K. Simon. Recognition of graphs with threshold dimension two. In Proceedings of the 27th ACM symposium on Theory of computing (STOC), pages 650–661, 1995.

22. J. P. Spinrad. Efficient graph representations, volume 19 of Fields Institute Monographs. American Mathematical Society, 2003.

23. M. Yannakakis. The complexity of the partial order dimension problem. SIAM Journal on Algebraic and Discrete Methods, 3:351–358, 1982.