Extracting Constrained 2-Interval Subsets in 2-Interval Sets ⋆

Guillaume Blin

Université Paris-Est, LABINFO-IGM
UMR CNRS 8049, 5 bd Descartes,
77 454 Marne-la-Vallée Cedex 2, FRANCE

Guillaume Fertin

LINA, FRE CNRS 2729
Université de Nantes, 2 rue de la Houssinière
BP 92208 44322 Nantes Cedex 3 - FRANCE

Stéphane Vialette

Laboratoire de Recherche en Informatique (LRI), UMR CNRS 8623
Université Paris-Sud, 91405 Orsay Cedex - FRANCE

Abstract

2-interval sets were used in [28,29] for establishing a general representation for macroscopic describers of RNA secondary structures. In this context, we have a 2-interval for each legal local fold in a given RNA sequence, and a constrained pattern made of disjoint 2-intervals represents a putative RNA secondary structure. We focus here on the problem of extracting a constrained pattern in a set of 2-intervals. More precisely, given a set of 2-intervals $\mathcal{D}$ and a model R describing if two disjoint 2-intervals in a solution can be in precedence order ($<$), be allowed to nest ($\sqsubset$) and/or be allowed to cross ($\between$), we consider the problem of finding a maximum cardinality subset $\mathcal{D}' \subseteq \mathcal{D}$ of disjoint 2-intervals such that any two 2-intervals in $\mathcal{D}'$ agree with R. The different combinations of restrictions on model R alter the computational complexity of the problem, and need to be examined separately.

In this paper, we improve the time complexity of [29] for model $R = \{\sqsubset\}$ by giving an optimal $O(n \log n)$ time algorithm, where n is the cardinality of the 2-interval set $\mathcal{D}$. We also give a graph-like relaxation for model $R = \{\sqsubset, \between\}$ that is solvable in $O(n^2\sqrt{n})$ time. Finally, we prove that the considered problem is **NP**-complete for model $R = \{<, \between\}$ even for same-length intervals, and give a fixed-parameter tractability result based on the crossing structure of $\mathcal{D}$.

Key words: 2-intervals, Pattern Matching, Computational complexity

1 Introduction

The problem of establishing a general representation of structured patterns, *i.e.*, *macroscopic describers* of RNA secondary structures, was considered in [28,29]. The approach is to set up a *geometric description* of helices by means of a natural generalization of intervals, namely a 2-*interval*. A 2-interval is the disjoint union of two intervals on the line. The geometric properties of 2-intervals provide a possible guide for understanding the computational complexity of finding structured patterns in RNA sequences. Using a model to represent non sequential information allows us for varying restrictions on the complexity of the pattern structure. Indeed, two disjoint 2-intervals, *i.e.*, two 2-intervals that do not intersect in any point, can be in precedence order ($<$), be allowed to nest ($\sqsubset$) or be allowed to cross ($\between$). Furthermore, the set of 2-intervals and the pattern can have different restrictions, *e.g.*, all intervals have the same length or all the intervals are disjoint. These different combinations of restrictions alter the computational complexity of the problems, and need to be examined separately. This examination produces efficient algorithms for more restrictive structured patterns, and hardness results for those less restrictive.

In this paper, we consider the problem of finding a constrained pattern in a set of 2-intervals. More precisely, given a set of 2-intervals $\mathcal{D}$ and a model R describing if two disjoint 2-intervals in a solution can be in precedence order ($<$), be allowed to nest ($\sqsubset$) and/or be allowed to cross ($\between$), we consider the problem of finding a maximum cardinality subset $\mathcal{D}' \subseteq \mathcal{D}$ of disjoint 2-intervals such that any two 2-intervals in $\mathcal{D}'$ agree with R. The problem of finding the largest 2-interval pattern in a set of 2-intervals $\mathcal{D}$ with respect to a given abstract model, referred hereafter as the 2-Interval Pattern problem, has been introduced by Vialette [28,29]. Vialette divided the problem in different classes based on the structure of the model and gave for most of them either **NP**-completeness results or polynomial-time algorithms. Dividing the problem in several classes was later proved to be extremely useful for approximating the 2-Interval Pattern problem [8].

⋆ An extended abstract of this work appeared in Proceedings of the 15^{th} Annual Symposium on Combinatorial Pattern Matching (CPM 2004) [5]. This work was partially supported by the CNRS *ACI Masse de Données* "NavGraphe" project.

Email addresses: `gblin@univ-mlv.fr` (Guillaume Blin), `fertin@lina.univ-nantes.fr` (Guillaume Fertin), `vialette@lri.fr` (Stéphane Vialette).

In the present paper, we focus on three special cases of the 2-Interval Pattern problem:

(1) The 2-intervals of the solution subset need to be pairwise nested,
(2) Two 2-intervals in a solution can only be nested or crossing, and all the intervals involved in the 2-interval set $\mathcal{D}$ are disjoint, and
(3) Two 2-intervals in a solution can only be nested or in precedence, and all the intervals involved in the 2-interval set $\mathcal{D}$ have the same length.

We give precise results for these three problems. Those three problems are of importance since each one is a straightforward extension of the problem of finding a given 2-interval set in another 2-interval set introduced in [29] and further studied in [18] and [23], and hence is strongly related, in the context of molecular biology, to pattern matching over RNA secondary structures. More precisely, in this paper, we improve the time complexity of the best known algorithm for $R = \{\sqsubset\}$ by giving an optimal $O(n \log n)$ time algorithm. Also, we give a graph-like relaxation for $R = \{\sqsubset, \between\}$ that is solvable in $O(n^2\sqrt{n})$ time. Finally, we prove that the problem is **NP**-complete for $R = \{<, \between\}$, and, we give a fixed-parameter tractability result based on the crossing structure of $\mathcal{D}$. Those results almost complete the table proposed by Vialette [29] (see Table 1) and provide an important step towards a better understanding of the precise complexity of 2-interval pattern matching problems.

There are basically two main lines of research our results refer to: (i) *arc-annotated sequences* and *protein topologies*, and (ii) *t-intervals combinatorics*.

- For a sequence S, an *arc-annotation* of S is a set of unordered pairs of positions in S. In this context, given two arc-annotated sequences S_1 and S_2, the Arc-Preserving Subsequence (APS) problem asks to find an occurrence of S_1 in S_2, and the Longest Arc-Preserving Common Subsequence (LAPCS) problems asks to find the longest common arc-annotated sequence that occurs both in S_1 and S_2. The APS and LAPCS problems are useful in representing the structural information of RNA and protein sequences [11,21,19,1]. The basic idea is to provide a measure for similarity, not only on the sequence level, but also on the structural level (an arc-annotated sequence is viewed as a RNA sequence together with phosphodiester bonds). Furthermore, a similar problem to compare the three-dimensional structure of proteins is the Contact Map Overlap problem described in [16]. Viksna and Gilbert described algorithms for pattern matching and pattern learning in TOPS diagram (formal description of protein topologies) [30].
- Our results are also related to the independent set problem in different extensions of 2-interval graphs. A graph G is a t-interval graph if there is an intersection model whose objects consist of collections of t intervals, $t \geq 1$, such that G is the intersection graph of this model [26,20]. From this definition, it is clear that every interval graph is a 1-interval graph.

Of particular interest is the class of 2-interval graphs. For example, line graphs, trees and circular-arc graphs are 2-interval graphs. However, West and Shmoys [31] have shown that the recognition problem for t-interval graphs is **NP**-complete for every $t \geq 2$ (this has to be compared with linear time recognition of 1-interval graphs). In the context of sequence similarity, [22] contains an application of graphs having interval number at most two. In [3], the authors considered the problem of scheduling jobs that are given as groups of non-intersecting segments on the real line. Of particular importance, they showed that the maximum weighted independent set for t-interval graphs ($t \geq 2$) is **APX**-hard even for highly restricted instances Also, they gave a $2t$-approximation algorithm for general instances based on a fractional version of the Local Ratio Technique [2]. Finally, some complexity issues of standard optimization problems for t-interval sets are given in [6].

The remainder of the paper is organized as follows. In Section 2 we briefly review the terminology introduced in [29]. In Section 3, we improve the time complexity of the best known algorithm for model $R = \{\sqsubset\}$. In Section 4, we give a graph-like relaxation for model $\{\sqsubset, \between\}$ that is solvable in polynomial-time. In Section 5, we prove that the 2-interval pattern problem for model $R = \{<, \between\}$ is **NP**-complete even when all intervals involved in the input 2-interval set have the same length. Finally, we give in Section 6 a fixed-parameter tractability result based on the crossing structure of $\mathcal{D}$.

2 Preliminaries

An interval and a 2-interval represent respectively a sequence of contiguous bases and pairings between two intervals, *i.e.*, *stems*, in RNA secondary structures. Thus, 2-intervals can be seen as *macroscopic describers* of RNA structures.

Formally, a 2-*interval* is the disjoint union of two intervals on a line. We denote it by $D = (I_1, J_1)$ where I_1 and J_1 are intervals such that $I_1 < J_1$ (here $<$ is the strict precedence order between intervals) ; in that case we also write $\mathsf{Left}(D) = I_1$ and $\mathsf{Right}(D) = J_1$. If $[x : y]$ and $[x' : y']$ are two intervals such that $[x : y] < [x' : y']$, we will sometimes write $D = ([x : y], [x' : y'])$ to emphasize on the precise definition of the 2-interval D. Let $D_1 = (I_1, J_1)$ and $D_2 = (I_2, J_2)$ be two 2-intervals. They are called *disjoint* if $(I_1 \cup J_1) \cap (I_2 \cup J_2) = \emptyset$ (*i.e.*, involved intervals do not intersect). The *covering interval* of a 2-interval D, written $\mathsf{Cover}(D)$, is the least interval covering both $\mathsf{Left}(D)$ and $\mathsf{Right}(D)$.

Of particular interest is the relation between two disjoint 2-intervals $D_1 =$

(I_1, J_1) and $D_2 = (I_2, J_2)$. We will write $D_1 < D_2$ if $I_1 < J_1 < I_2 < J_2$, $D_1 \sqsubset D_2$ if $I_2 < I_1 < J_1 < J_2$ and $D_1 \between D_2$ if $I_1 < I_2 < J_1 < J_2$. Two 2-intervals D_1 and D_2 are τ-comparable for some $\tau \in \{<, \sqsubset, \between\}$ if $D_1 \tau D_2$ or $D_2 \tau D_1$. Let $\mathcal{D}$ be a set of 2-intervals and $R \subseteq \{<, \sqsubset, \between\}$ be non-empty. The set $\mathcal{D}$ is *R-comparable* if any two distinct 2-intervals of $\mathcal{D}$ are τ-comparable for some $\tau \in R$. Throughout the paper, the non-empty subset R is called a *model*. Clearly, if a set of 2-intervals $\mathcal{D}$ is R-comparable then $\mathcal{D}$ is a set of disjoint 2-intervals. The *ground set* of a set of 2-intervals $\mathcal{D}$, written $\mathsf{GS}(\mathcal{D})$, is the set of all *simple* intervals involved in $\mathcal{D}$, *i.e.*, $\mathsf{GS}(\mathcal{D}) = \bigcup_{D \in \mathcal{D}} (\mathsf{Left}(D) \cup \mathsf{Right}(D))$. The *leftmost* (resp. *rightmost*) element of a set of disjoint 2-intervals $\mathcal{D}$ is the 2-interval $D_i \in \mathcal{D}$ such that $\mathsf{Left}(D_i) < \mathsf{Left}(D_j)$ (resp. $\mathsf{Right}(D_j) < \mathsf{Right}(D_i)$) for all $D_j \in \mathcal{D} - D_i$. Observe that it could be the case that D_i is both the leftmost and rightmost element of $\mathcal{D}$ (this is indeed the case if $|\mathcal{D}| = 1$ or if $D_j \sqsubset D_i$ for all $D_j \in \mathcal{D} - D_i$).

We define hereafter two additional parameters on $\mathcal{D}$. The *depth* of $\mathcal{D}$, written $\mathsf{Depth}(\mathcal{D})$, is the size of a maximum cardinality $\{\between\}$-comparable subset of $\mathcal{D}$ (according to [29], this parameter is polynomial-time computable). The *forward crossing number* of $\mathcal{D}$, written $\mathsf{FCrossing}(\mathcal{D})$, is defined by $\mathsf{FCrossing}(\mathcal{D}) = \max_{D_i \in \mathcal{D}} |\{D_j : D_i \between D_j\}|$. Clearly, $\mathsf{FCrossing}(\mathcal{D}) \geq \mathsf{Depth}(\mathcal{D}) - 1$ for any set $\mathcal{D}$ of 2-intervals.

Following [11], Vialette proposed in [29], two natural restrictions on the ground set of $\mathcal{D}$ (a third restriction, *i.e.*, *balanced 2-intervals*, well-suited for investigating RNA secondary structures space was introduced in [8]):

(1) all the intervals of the ground set $\mathsf{GS}(\mathcal{D})$ are of the same length,
(2) all the intervals of the ground set $\mathsf{GS}(\mathcal{D})$ are disjoint, *i.e.*, if two intervals $I, I' \in \mathsf{GS}(\mathcal{D})$ overlap, then $I = I'$.

Using restrictions on the ground set allows us for varying restrictions on the complexity of the 2-interval set structure, and hence on the complexity of the problems. These two restrictions involve three levels of complexity:

- UNLIMITED: no restrictions
- UNIT: restriction 1
- DISJOINT: restrictions 1 and 2

Given a set of 2-intervals $\mathcal{D}$, a model $R \subseteq \{<, \sqsubset, \between\}$ and a positive integer k, the 2-Interval Pattern problem consists in finding a subset $\mathcal{D}' \subseteq \mathcal{D}$ of cardinality at least k such that $\mathcal{D}'$ is R-comparable. For the sake of brevity, the 2-Interval Pattern problem with respect to a model R over an unlimited (resp. unit and disjoint) ground set is abbreviated in 2-IP-Unl-R (resp. 2-IP-Unit-R and 2-IP-Dis-R).

Vialette proved in [29] that 2-IP-Unit-$\{<, \sqsubset, \between\}$ and 2-IP-Unit-$\{\sqsubset, \between\}$ are

NP-complete. Moreover, he gave polynomial-time algorithms for the problem with respect to the models $\{<\}$, $\{\sqsubset\}$, $\{\between\}$ and $\{<, \sqsubset\}$ (cf. Table 1).

In this article, we answer three open problems and we improve the complexity of another one, as shown in Table 1. Moreover, we show that 2-IP-Unit-$\{<, \between\}$ is fixed parameter tractable when parameterized by the forward crossing number of $\mathcal{D}$.

2-Interval Pattern Problem			
	Ground Set		
Model	Unlimited	Unit	Disjoint
$\{<, \sqsubset, \between\}$	**NP**-complete		$O(n\sqrt{n})$[24]
$\{\sqsubset, \between\}$	**NP**-complete		$O(n^2\sqrt{n})$ $\star$
$\{<, \sqsubset\}$	$O(n^2)$		
$\{<, \between\}$	**NP**-complete $\star$		?
$\{<\}$	$O(n \log n)$		
$\{\sqsubset\}$	$O(n \log n)$ $\star$ $\bullet$		
$\{\between\}$	$O(n^2 \log n)$		

Table 1
2-interval pattern problem complexity where $n = |\mathcal{D}|$. When not specified, the complexity comes from [29]. $\star$ contributions of the present paper. $\bullet$ improvement of the existing complexity (which was $O(n^2)$ in [29]).

3 Improving the complexity of 2-IP-Unl-$\{\sqsubset\}$

The problem of finding the largest $\{\sqsubset\}$-comparable subset in a set of 2-intervals was considered in [29]. Observing that this problem is equivalent to finding a largest clique in a comparability graph (a linear time solvable problem [17]), an $O(n^2)$ time algorithm was thus proposed. We improve that result by giving an optimal $O(n \log n)$ time algorithm.

The inefficiency of the algorithm proposed in [29] lies in the effective construction of a comparability graph. We show that this construction can be avoided by considering trapezoids instead of 2-intervals. Recall that a *trapezoid graph* is the intersection graph of a finite set of trapezoids between two parallel lines [9] (it is easily seen that trapezoid graphs generalize both interval graphs and permutation graphs). Analogously to 2-intervals, we will denote by $T = ([x : y], [x' : y'])$ the trapezoid with top interval $[x : y]$ and bottom interval $[x' : y']$.

Proposition 1 2-IP-Unl-$\{\sqsubset\}$ *is solvable in* $O(n \log n)$ *time.*

PROOF. Let $\mathcal{D} = \{D_1, D_2, \ldots, D_n\}$ be a collection of 2-intervals of the real line. Construct a collection of trapezoids $\mathcal{T} = \{T_1, T_2, \ldots, T_n\}$ between two parallel lines as follows. For each 2-interval $D_i = ([x : y], [x' : y']) \in \mathcal{D}$, we add the trapezoid $T_i = ([x : y], [-y' : -x'])$ to $\mathcal{T}$.

Claim 2 *For all $1 \leq i \leq j \leq n$, the 2-intervals D_i and D_j are $\{\sqsubset\}$-comparable if and only if the trapezoids T_i and T_j are non-intersecting.*

PROOF. [of Claim] Let $D_i = ([x_i : y_i], [x'_i : y'_i])$ and $D_j = ([x_j : y_j], [x'_j : y'_j])$ be two 2-intervals of $\mathcal{D}$, and $T_i = ([x_i : y_i], [-y'_i : -x'_i])$ and $T_j = ([x_j : y_j], [-y'_j : -x'_j])$ be the two corresponding trapezoids in $\mathcal{T}$. Suppose that D_i and D_j are $\{\sqsubset\}$-comparable. Without loss of generality, we may assume $D_j \sqsubset D_i$. Thus, we have $y_i < x_j$ and $y'_j < x'_i$. It follows immediately that $-x'_i < -y'_j$, and hence the two trapezoids T_i and T_j are non-intersecting. The proof of the converse is identical. □

Clearly, the collection $\mathcal{T}$ can be constructed in $O(n)$ time. Based on a geometric representation of trapezoid graphs by boxes in the plane, Felsner *et al.* [12] have designed a $O(n \log n)$ algorithm for finding a maximum cardinality subcollection of non-intersecting trapezoids in a collection of trapezoids, and the proposition follows. □

Based on Fredman's bound for the number of comparisons needed to compute maximum increasing subsequences in permutation [13], Felsner *et al.* [12] argued that their $O(n \log n)$ time algorithm for finding a maximum cardinality subcollection of non-intersecting trapezoids in a collection of trapezoids is optimal. Then it follows from Proposition 1 that our $O(n \log n)$ time algorithm for finding a maximum cardinality $\{\sqsubset\}$-comparable subset in a set of 2-intervals is optimal as well.

4 A polynomial-time algorithm for 2-IP-DIS-$\{\sqsubset, \between\}$

In this section, we give an $O(n^2\sqrt{n})$ time algorithm for the 2-IP-DIS-$\{\sqsubset, \between\}$ problem, where n is the cardinality of the set of 2-intervals $\mathcal{D}$. Recall that given a set of 2-intervals $\mathcal{D}$ over a disjoint ground set, the problem asks to find the size of a maximum cardinality $\{\sqsubset, \between\}$-comparable subset $\mathcal{D}' \subseteq \mathcal{D}$. Observe that the 2-IP-DIS-$\{\sqsubset, \between\}$ problem has an interesting formulation in terms of constrained matchings in general graphs: Given a graph G together with a linear ordering π of its vertices, the 2-IP-DIS-$\{\sqsubset, \between\}$ problem is equivalent to finding a maximum cardinality matching $\mathcal{M}$ in G with the property that for any two distinct edges $\{u, v\}$ and $\{u', v'\}$ of $\mathcal{M}$, neither $\max\{\pi(u), \pi(v)\} < \min\{\pi(u'), \pi(v')\}$ nor $\max\{\pi(u'), \pi(v')\} < \min\{\pi(u), \pi(v)\}$ occur.

Roughly speaking, our algorithm is a three-step procedure. First, the interval graph of all the covering intervals of the 2-intervals in $\mathcal{D}$ is constructed. Next, all the maximal cliques of that graph are efficiently computed. Finally, for each maximal clique we construct a new graph and find a solution using a maximum cardinality matching algorithm. The size of a best solution found in the third step is thus returned. Clearly, the efficiency of our algorithm relies upon an efficient algorithm for finding all the maximal cliques in the intersection of the covering intervals. We now proceed with the details of our algorithm.

Let $\mathcal{D} = \{D_i : 1 \leq i \leq n\}$ be a set of 2-intervals. Consider the set $\mathcal{C}_\mathcal{D}$ composed of all the covering intervals of the 2-intervals in $\mathcal{D}$, *i.e.*, $\mathcal{C}_\mathcal{D} = \{\mathsf{Cover}(D) : D \in \mathcal{D}\}$. Now, let $\Omega(\mathcal{C}_\mathcal{D})$ be the interval graph associated with $\mathcal{C}_\mathcal{D}$. The graph $\Omega(\mathcal{C}_\mathcal{D})$ has a vertex v_i for each interval $\mathsf{Cover}(D_i)$ in $\mathcal{C}_\mathcal{D}$ and two vertices v_i and v_j of $\Omega(\mathcal{C}_\mathcal{D})$ are joined by an edge if the two associated intervals $\mathsf{Cover}(D_i)$ and $\mathsf{Cover}(D_j)$ intersect. An illustration of $\mathcal{C}_\mathcal{D}$ and $\Omega(\mathcal{C}_\mathcal{D})$ for a given set of 2-intervals $\mathcal{D}$ is given in Figure 1. Most in the interest in the interval graph $\Omega(\mathcal{C}_\mathcal{D})$ stems from the following lemma.

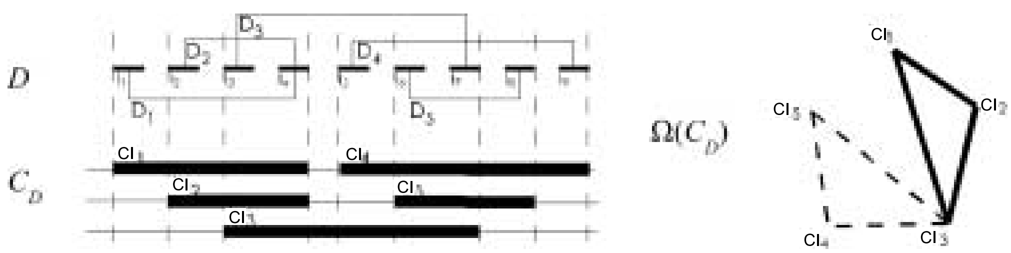

Fig. 1. Illustration of $\mathcal{C}_\mathcal{D}$ and $\Omega(\mathcal{C}_\mathcal{D})$ for a given set of 2-intervals $\mathcal{D}$ on a disjoint ground set.

Lemma 3 *Let $\mathcal{D}$ be a set of 2-intervals and $\mathcal{D}'$ be a $\{\sqsubset, \between\}$-comparable subset of $\mathcal{D}$. Then, $\{v_i : D_i \in \mathcal{D}'\}$ induces a complete graph in $\Omega(\mathcal{C}_\mathcal{D})$.*

PROOF. Let D_i and D_j be two distinct 2-intervals of $\mathcal{D}'$. Since D_i and D_j are $\{\sqsubset, \between\}$-comparable then it follows that either intervals $\mathsf{Cover}(D_i)$ and $\mathsf{Cover}(D_j)$ overlap or one interval is completely contained in the other. In both cases, intervals $\mathsf{Cover}(D_i)$ and $\mathsf{Cover}(D_j)$ intersect, and hence vertices v_i and v_j are joined by an edge in $\Omega(\mathcal{C}_\mathcal{D})$. Therefore $\{v_i : D_i \in \mathcal{D}'\}$ induces a complete graph in $\Omega(\mathcal{C}_\mathcal{D})$. $\square$

Observe that the converse is false since the intersection of two 2-intervals in $\mathcal{D}$ results in an edge in $\Omega(\mathcal{C}_\mathcal{D})$, and hence two 2-intervals associated to two distinct vertices in a clique may not be $\{\sqsubset, \between\}$-comparable. However, thanks to Lemma

3 we now only need to focus on maximal cliques of $\Omega(\mathcal{C}_{\mathcal{D}})$. Several problems that are **NP**-complete on general graphs have polynomial-time algorithms for interval graphs. The problem of finding all the maximal cliques of a graph is one such example. Indeed, an interval graph $G = (V, E)$ is a chordal graph and as such has at most $|V|$ maximal cliques [14]. Furthermore, all the maximal cliques of a chordal graph can be found in $O(n+m)$ time, where $n = |V|$ and $m = |E|$, by a modification of Maximum Cardinality Search (MCS) [25,4].

Let C be a maximal clique of $\Omega(\mathcal{C}_{\mathcal{D}})$. As observed above, any two 2-intervals associated to two distinct vertices in the maximal clique C may not be $\{\sqsubset, \between\}$-comparable. Let $\mathcal{D}' \subseteq \mathcal{D}$ be the set of all 2-intervals associated to vertices in the maximal clique C. Based on C, consider the graph $G_C = (V_C, E_C)$ defined by $V_C = \mathsf{GS}(\mathcal{D}')$ and $E_C = \{\{I, J\} : D = (I, J) \in \mathcal{D}'\}$. In other words, the set of vertices of G_C is the ground set of $\mathcal{D}'$ and the edges of G_C is the 2-interval subset $\mathcal{D}'$ itself viewed as a set of subsets of size 2. Note that the construction of G_C is possible only because $\mathcal{D}'$ has disjoint ground set. The following lemma is an immediate consequence of the definition of G_C and Lemma 3.

Lemma 4 *Let C be a clique in $\Omega(\mathcal{C}_{\mathcal{D}})$ and $G_C = (V_C, E_C)$ be the graph constructed as detailed above. Then, $\{(I_{i_1}, J_{i_1}), (I_{i_2}, J_{i_2}), \ldots, (I_{i_k}, J_{i_k})\}$ is a $\{\sqsubset, \between\}$-comparable subset if and only if $\{\{I_{i_1}, J_{i_1}\}, \{I_{i_2}, J_{i_2}\}, \ldots, \{I_{i_k}, J_{i_k}\}\}$ is a matching in G_C.*

Proposition 5 *The* 2-IP-DIS-$\{\sqsubset, \between\}$ *problem is solvable in $O(n^2\sqrt{n})$ time, where n is the number of 2-intervals in $\mathcal{D}$.*

PROOF. Consider the algorithm given in Figure 2. Correctness of this algorithm follows from Lemmas 3 and 4. What is left is to prove the time complexity. Clearly, the interval graph $\Omega(\mathcal{C}_{\mathcal{D}})$ can be constructed in $O(n^2)$ time. All the maximal cliques of $\Omega(\mathcal{C}_{\mathcal{D}})$ can be found in $O(n+m)$ time, where m is the number of edges in $\Omega(\mathcal{C}_{\mathcal{D}})$ [25,4]. Summing up, the first two steps can be done in $O(n^2)$ time since $m < n^2$. We now turn to the time complexity of the loop (in fact the dominant term of our analysis). For each maximal clique C of $\Omega(\mathcal{C}_{\mathcal{D}})$, the graph G_C can be constructed in $O(n)$ time since $|C| \leq n$. We now consider the computation of a maximal matching in G_C. Micali and Vazirani [24] (see also [27]) gave an $O(\sqrt{|V|}|E|)$ time algorithm for finding a maximal matching in a graph $G = (V, E)$. But G_C has at most n edges (as each edge corresponds to a 2-interval) and hence has at most $2n$ vertices. Then it follows that a maximum matching $\mathcal{M}$ in G_C can be found in $O(n\sqrt{n})$ time. Since $\Omega(\mathcal{C}_{\mathcal{D}})$ is an interval graph with n vertices, it has at most n maximal cliques [14], we conclude that the algorithm as a whole runs in $O(n^2\sqrt{n})$ time. $\square$

Max $\{\sqsubset, \between\}$-Comparable 2-Interval Pattern

Input: A set of 2-intervals $\mathcal{D}$ with disjoint ground set
Output: The size of a maximum cardinality $\{\sqsubset, \between\}$-comparable subset of $\mathcal{D}$

1. Construct the interval graph $\Omega(\mathcal{C}_{\mathcal{D}})$
2. Compute all maximal cliques in $\Omega(\mathcal{C}_{\mathcal{D}})$
3. For each maximal clique C in $\Omega(\mathcal{C}_{\mathcal{D}})$

3.1. Construct the graph G_C
3.2. Compute a maximal matching $\mathcal{M}$ in G_C
3.3. Store the cardinality of $\mathcal{M}$ in $m(C)$

4. Return $\max\{m(C) : C \text{ is a maximal clique of } \Omega(\mathcal{C}_{\mathcal{D}})\}$

Fig. 2. Algorithm Max $\{\sqsubset, \between\}$-Comparable 2-Interval Pattern.

5 2-IP-Unit-$\{<, \between\}$ is NP-complete

Theorem 6 below completes the analysis of 2-IP-Unit-R and 2-IP-Unl-R for any model $R \subseteq \{<, \sqsubset, \between\}$ (see Table 1).

Theorem 6 *The* 2-IP-Unit-$\{<, \between\}$ *problem is* **NP**-*complete.*

PROOF. First, we will present the two decision problems we will deal with (Exact 3-CNF-Sat and 2-IP-Unit-$\{<, \between\}$). Then, we will give several intermediate lemmas that will finally be used in Proposition 14 to validate the proof of the **NP**-completeness of the 2-IP-Unit-$\{<, \between\}$ problem.

We provide a polynomial-time reduction from the Exact 3-CNF-Sat problem: Given a set $\mathcal{V}_n$ of n variables and a set $\mathcal{C}_q$ of q clauses (each composed of three literals) over $\mathcal{V}_n$, the problem asks to find a truth assignment for $\mathcal{V}_n$ that satisfies all clauses of $\mathcal{C}_q$. It is well-known that the Exact 3-CNF-Sat problem is **NP**-complete [15]. For the sake of clarity, we now state formally the 2-IP-Unit-$\{<, \between\}$ problem: Given a set of 2-intervals $\mathcal{D}$, and a positive integer k, the problem asks to find a subset $\mathcal{D}' \subseteq \mathcal{D}$ of cardinality greater than or equal to k, such that $\mathcal{D}'$ is $\{<, \between\}$-comparable.

Clearly, 2-IP-Unit-$\{<, \between\}$ problem is in NP. We show that given any instance of Exact 3-CNF-Sat with q clauses on a set of n variables, we can construct in polynomial-time an instance of the 2-IP-Unit-$\{<, \between\}$ problem with $k = (7n-2)q$ such that there exists a satisfying truth assignment for the boolean formula iff there exists a $\{<, \between\}$-comparable subset $\mathcal{D}' \in \mathcal{D}$ of size at least k. We detail this construction hereafter.

Let $\mathcal{V}_n = \{x_1, x_2, ...x_n\}$ be a set of n variables and $\mathcal{C}_q = \{c_1, c_2, \ldots, c_q\}$ be a collection of q clauses. For the sake of clarity, let us define $\mathcal{D}$ on the integral line such that any interval of the ground set is of size four. Let us start with the precise definition of the representation of a single clause c_i of $\mathcal{C}_q$ as illustrated in Figure 4. The dotted rectangle on the left (resp. right) is part of the representation of clause c_{i-1} (resp. c_{i+1}). The precise adjustment of the representation of two consecutive clauses is illustrated in Figure 3 and formally defined afterwards. For convenience, we will split the representation of c_i into seven groups (represented in gray): A^i, B^i, C^i_L, C^i_R, D^i, E^i and F^i. Each group in turn is divided into blocks (represented in white). There are $11 + 2n$ blocks for each clause: n blocks for A^i; 3 blocks for B^i; 1 block for C^i_L; n blocks for C^i_R; 2 blocks for D^i; 3 blocks for E^i; 2 blocks for F^i.

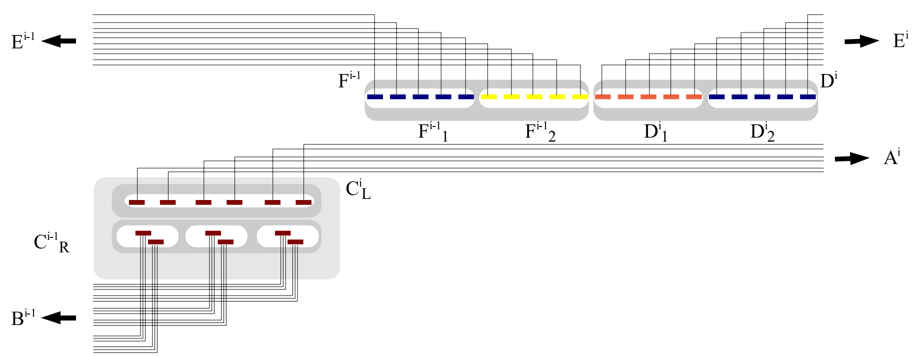

Fig. 3. Junction between the representation of clauses c_{i-1} and c_i

For example, in Figure 4 we use three boolean variables and hence we have seventeen blocks. For the sake of clarity, in the figures of this section, the intervals of the ground set might be drawn on different levels.

We now turn to give a precise definition of each group in the representation of a given clause c_i. In the following, we will refer to an interval of the ground set as a *simple* interval. Let $FP(c_i)$ denote the smallest starting position of any simple interval of the representation of clause c_i. We set, for convenience, $FP(c_1) = 0$. For any $1 < i \leq q$, we have $FP(c_i) = FP(c_{i-1}) + 104n - 21$. Moreover, let $FP(\alpha)$ denote the smallest starting position of any simple interval of group $\alpha \in \{C^i_L, A^i, B^i, C^i_R, D^i, E^i, F^i | 1 \leq i \leq q\}$.

Group C^i_L is composed of one block containing $2n$ simple intervals (as illustrated in Figure 5): $\{[FP(C^i_L) + 7k, FP(C^i_L) + 7k + 4] | 0 \leq k \leq 2n - 1\}$, where $FP(C^i_L) = FP(c_i)$. The $2n$ simple intervals of the block of group C^i_L represent in the left to right order $(x_1, \overline{x_1}, x_2, \overline{x_2} \ldots x_n, \overline{x_n})$. By definition, the simple interval representing x_m in C^i_L is defined by $[FP(C^i_L) + 14(m-1), FP(C^i_L) + 14(m-1) + 4]$. And consequently, the simple interval representing $\overline{x_m}$ in C^i_L is defined by $[FP(C^i_L) + 14(m-1) + 7, FP(C^i_L) + 14(m-1) + 11]$.

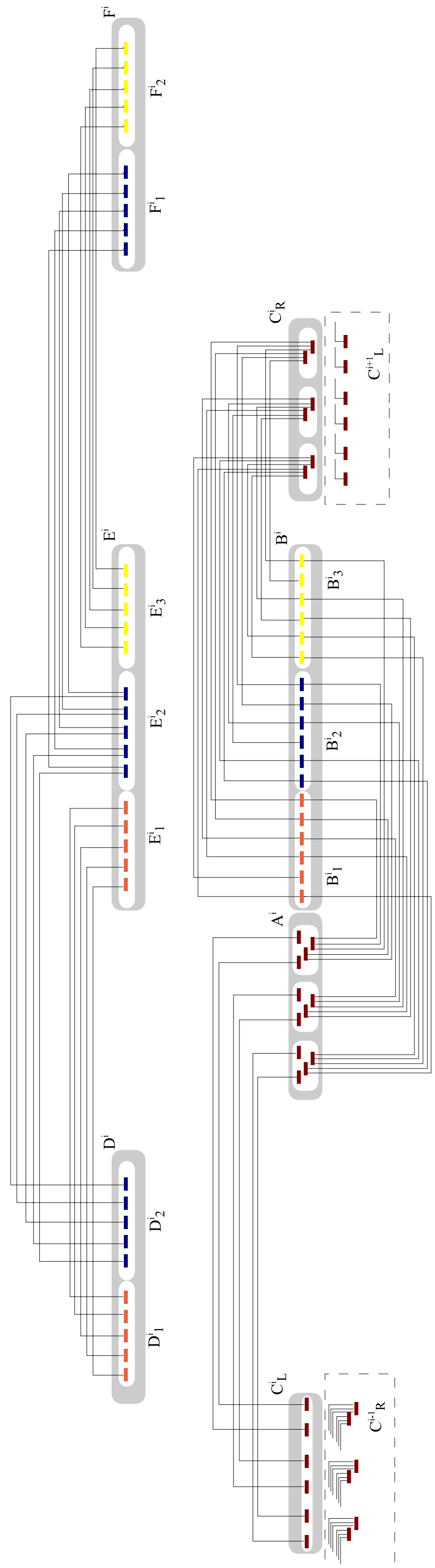

Fig. 4. Representation of clause $c_i = (\overline{x_1} \vee x_2 \vee x_3)$ where $n = 3$.

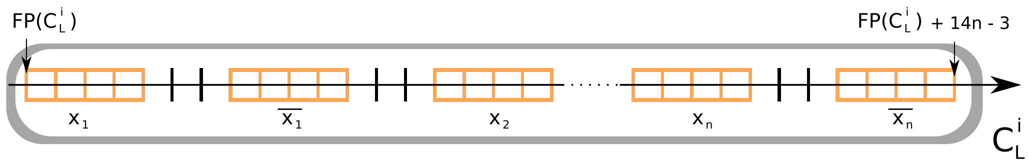

Fig. 5. Description of the simple intervals (represented as blocks of four consecutive squares) of group C_L^i.

Group D^i is composed of two blocks (D_1^i and D_2^i), each containing $2n-1$ simple intervals (as illustrated in Figure 6): $\{[FP(D^i) + 5k, FP(D^i) + 5k + 4] | 0 \leq k \leq 4n-3\}$ where $FP(D^i) = FP(c_i) + 34n - 10$. By construction, block D_1^i is composed of the following simple intervals: $\{[FP(D^i)+5k, FP(D^i)+5k+4] | 0 \leq k \leq 2n-2\}$ and block D_2^i is composed of the following simple intervals: $\{[FP(D^i) + 5k, FP(D^i) + 5k + 4] | 2n-1 \leq k \leq 4n-3\}$.

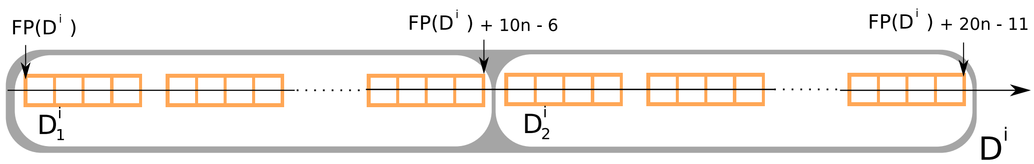

Fig. 6. Description of the simple intervals of group D^i.

Group A^i is composed of n blocks (one block for each boolean variable), each containing four simple intervals (as illustrated in Figure 7): $\{[FP(A^i) + 7k, FP(A^i) + 7k + 4], [FP(A^i) + 2 + 14l, FP(A^i) + 6 + 14l], [FP(A^i) + 5 + 14l, FP(A^i) + 9 + 14l] | 0 \leq k \leq 2n-1, 0 \leq l \leq n-1\}$ where $FP(A^i) = FP(c_i) + 54n - 20$. The $4n$ simple intervals of group A^i represent in the left to right order $(\overline{x_1}, x_1, \overline{x_1}, x_1, \overline{x_2}, x_2, \overline{x_2}, x_2, \ldots \overline{x_n}, x_n, \overline{x_n}, x_n)$. By construction, in any block of group A^i the second (resp. third) simple interval overlaps both the first and the third (resp. the second and the fourth) simple intervals. By definition, the two simple intervals representing x_m in A^i are defined by $[FP(A^i)+14(m-1)+7, FP(A^i)+14(m-1)+11]$ and $[FP(A^i)+14(m-1)+2, FP(A^i)+14(m-1)+6]$. Consequently, the two simple intervals representing $\overline{x_m}$ in A^i are defined by $[FP(A^i) + 14(m-1), FP(A^i) + 14(m-1) + 4]$ and $[FP(A^i) + 14(m-1) + 5, FP(A^i) + 14(m-1) + 9]$.

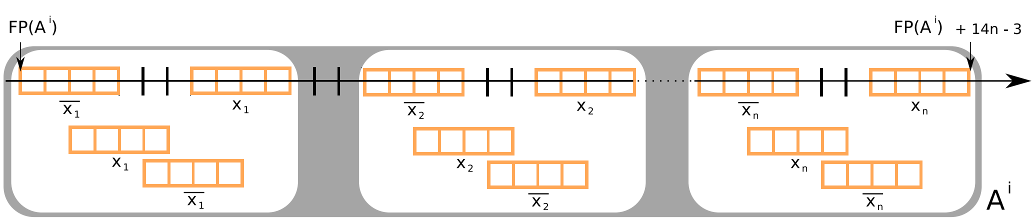

Fig. 7. Description of the simple intervals of group A^i.

Group B^i is composed of three blocks (one for each literal in a clause), each containing $2n$ simple intervals (as illustrated in Figure 8): $\{[FP(B_1^i) +$

$6k, FP(B_1^i)+6k+4], [FP(B_2^i)+6k, FP(B_2^i)+6k+4], [FP(B_3^i)+6k, FP(B_3^i)+6k+4] | 0 \leq k \leq 2n-1\}$ where $FP(B_1^i) = FP(c_i)+68n-20$, $FP(B_2^i) = FP(c_i)+80n-20$, $FP(B_3^i) = FP(c_i)+92n-20$. The $2n$ simple intervals of each block of group B^i represent in the left to right order $(x_1, \overline{x_1}, x_2, \overline{x_2} \ldots x_n, \overline{x_n})$. By definition, the simple interval representing x_m in B_j^i, with $j \in \{1,2,3\}$, is defined by $[FP(B_j^i)+12(m-1), FP(B_j^i)+12(m-1)+4]$. And consequently, the simple interval representing $\overline{x_m}$ in B_j^i, with $j \in \{1,2,3\}$, is defined by $[FP(B_j^i)+12(m-1)+6, FP(B_j^i)+12(m-1)+10]$.

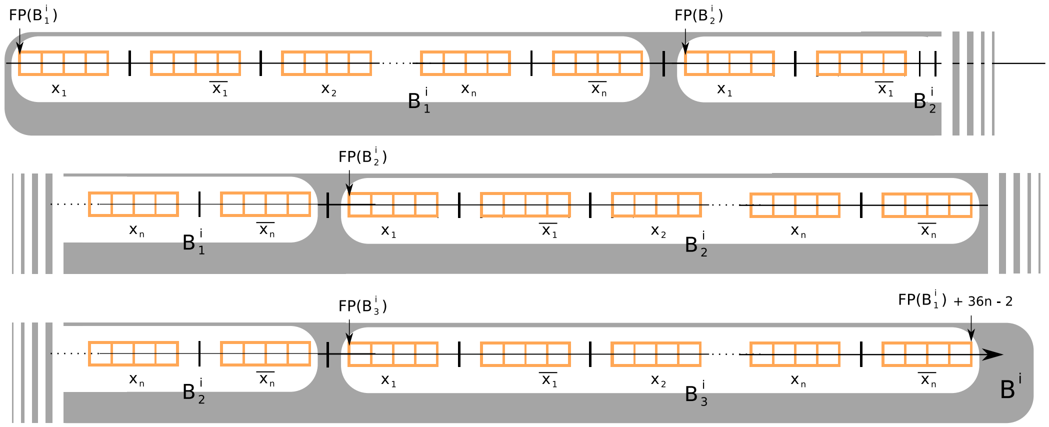

Fig. 8. Description of the simple intervals of group B^i. Due to space considerations, the description is divided in three lines. Each line starts with the end part of the previous line in order to indicate the configuration of the whole description.

Group E^i is composed of three blocks, each containing $2n-1$ simple intervals (as illustrated in Figure 9): $\{[FP(E_1^i)+6k, FP(E_1^i)+6k+4], [FP(E_2^i)+6k, FP(E_2^i)+6k+4], [FP(E_3^i)+6k, FP(E_3^i)+6k+4] | 0 \leq k \leq 2n-2\}$ where $FP(E_1^i) = FP(c_i)+68n-17$, $FP(E_2^i) = FP(c_i)+80n-17$, $FP(E_3^i) = FP(c_i)+92n-17$. Therefore, each simple interval of block E_j^i intersects exactly two simple intervals of block B_j^i, for $1 \leq j \leq 3$.

Group C_R^i is composed of n blocks (one block for each boolean variable), each containing two simple intervals (as illustrated in Figure 10): $\{[FP(C_R^i)+14k, FP(C_R^i)+14k+4], [FP(C_R^i)+14k+3, FP(C_R^i)+14k+7] | 0 \leq k \leq n-1\}$ where $FP(C_R^i) = FP(c_i)+104n-19$. The $2n$ simple intervals of group C_R^i represent in the left to right order $(x_1, \overline{x_1}, x_2, \overline{x_2} \ldots x_n, \overline{x_n})$. By definition, the simple interval representing x_m in C_R^i is defined by $[FP(C_R^i)+14(m-1), FP(C_R^i)+14(m-1)+4]$. And consequently, the simple interval representing $\overline{x_m}$ in C_R^i is defined by $[FP(C_R^i)+14(m-1)+3, FP(C_R^i)+14(m-1)+7]$. Therefore, by construction, in any block of group C_R^i the two simple intervals composing this block are overlapping.

Finally, group F^i is composed of two blocks, each containing $2n-1$ simple intervals (as illustrated in Figure 11): $\{[FP(F^i)+5k, FP(F^i)+5k+4] | 0 \leq k \leq 4n-3\}$ where $FP(F^i) = FP(c_i)+118n-21$. By construction, block F_1^i is

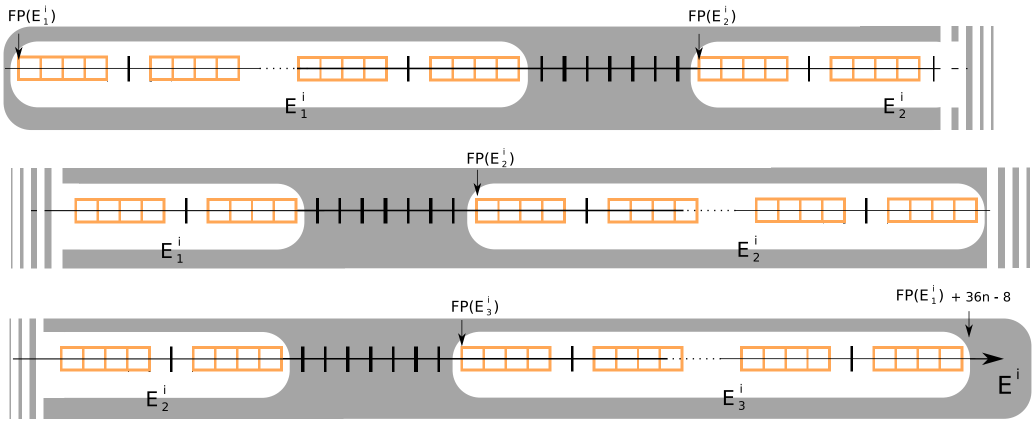

Fig. 9. Description of the simple intervals of group E^i. As in Figure 8, due to space considerations, the description is divided in three lines.

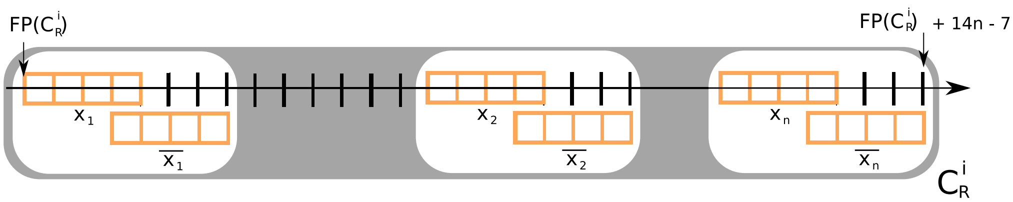

Fig. 10. Description of the simple intervals of group C_R^i.

composed of the following simple intervals: $\{[FP(F^i)+5k, FP(F^i)+5k+4] | 0 \leq k \leq 2n-2\}$ and block F_2^i is composed of the following simple intervals: $\{[FP(F^i)+5k, FP(F^i)+5k+4] | 2n-1 \leq k \leq 4n-3\}$.

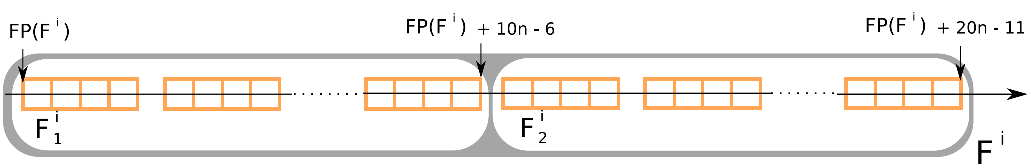

Fig. 11. Description of the simple intervals of group F^i.

The set of simple intervals of the instance of 2-IP-Unit-$\{<, \between\}$ is obtained by assembling together in order the representation of the clauses c_1 to c_q. It is easy to check the following properties (which are represented in Figure 12):

- for any $1 < i \leq q$, the smallest position of any simple interval of group C_L^i is greater than the biggest position of any simple interval of groups E^{i-1} and B^{i-1};
- for any $1 < i \leq q$, the smallest position of any simple interval of group F^{i-1} is greater than the biggest position of any simple interval of group C_L^i;
- for any $1 < i \leq q$, the biggest position of any simple interval of group F^{i-1} is less than the smallest position of any simple interval of group D^i;

- for any $1 \leq i \leq q$, the smallest position of any simple interval of group A^i is greater than the biggest position of any simple interval of group D^i;
- for any $1 \leq i \leq q$, the biggest position of any simple interval of group A^i is less than the smallest position of any simple interval of groups B^i and E^i;
- for any $1 \leq i \leq q$, the smallest position of any simple interval of group C_R^i is greater than the biggest position of any simple interval of groups B^i and E^i;
- for any $1 \leq i \leq q$, the biggest position of any simple interval of group C_R^i is less than the smallest position of any simple interval of group F^i.

Now that we have defined the ground set of $\mathcal{D}$, let us define formally the 2-intervals of $\mathcal{D}$ (partially illustrated in Figure 4).

For each clause c_i, $\mathcal{D}$ is composed of $2n$ 2-intervals built with a simple interval of group C_L^i and a simple interval of group A^i:

- $\{([FP(C_L^i)+r, FP(C_L^i)+r+4], [FP(A^i)+s, FP(A^i)+s+4]),$
- $([FP(C_L^i)+s, FP(C_L^i)+s+4], [FP(A^i)+r, FP(A^i)+r+4])\}$
  with $r = 14(k-1), s = r+7, 1 \leq k \leq n$

For each clause c_i, $\mathcal{D}$ is composed of $4n-2$ 2-intervals built with a simple interval of group D^i and a simple interval of group E^i:

- $\{([FP(D^i)+5k, FP(D^i)+5k+4], [FP(E_1^i)+6k'', FP(E_1^i)+6k''+4]),$
- $([FP(D^i)+5k', FP(D^i)+5k'+4], [FP(E_2^i)+6k'', FP(E_2^i)+6k''+4])\}$
  with $0 \leq k \leq 2n-2, 2n-1 \leq k' \leq 4n-3, 0 \leq k'' \leq 2n-2$.

For each clause c_i, $\mathcal{D}$ is composed of $6n$ 2-intervals built with a simple interval of group B^i and a simple interval of group C_R^i:

- $\{([FP(B_j^i)+r, FP(B_j^i)+r+4], [FP(C_R^i)+s, FP(C_R^i)+s+4]),$
- $([FP(B_j^i)+r+6, FP(B_j^i)+r+10], [FP(C_R^i)+s+3, FP(C_R^i)+s+7])\}$
  with $r = 12(k-1), s = 14(k-1), j \in \{1,2,3\}, 1 \leq k \leq n$.

For each clause c_i, $\mathcal{D}$ is composed of $4n-2$ 2-intervals built with a simple interval of group E^i and a simple interval of group F^i:

- $\{([FP(E_2^i)+6k', FP(E_2^i)+6k'+4], [FP(F^i)+5k, FP(F^i)+5k+4]),$
- $([FP(E_3^i)+6k', FP(E_3^i)+6k'+4], [FP(F^i)+5k'', FP(F^i)+5k''+4])\}$
  with $2n-2 \leq k \leq 4n-3, 0 \leq k' \leq 2n-2, 4n-2 \leq k'' \leq 6n-4\}$.

For each clause c_i, $\mathcal{D}$ is composed of $6n$ 2-intervals built with a simple interval of group A^i and a simple interval of group B^i:

- $\{([FP(A^i)+r+2, FP(A^i)+r+6], [FP(B_j^i)+s, FP(B_j^i)+s+4]),$
- $([FP(A^i)+r+5, FP(A^i)+r+9], [FP(B_j^i)+s+6, FP(B_j^i)+s+10])\}$

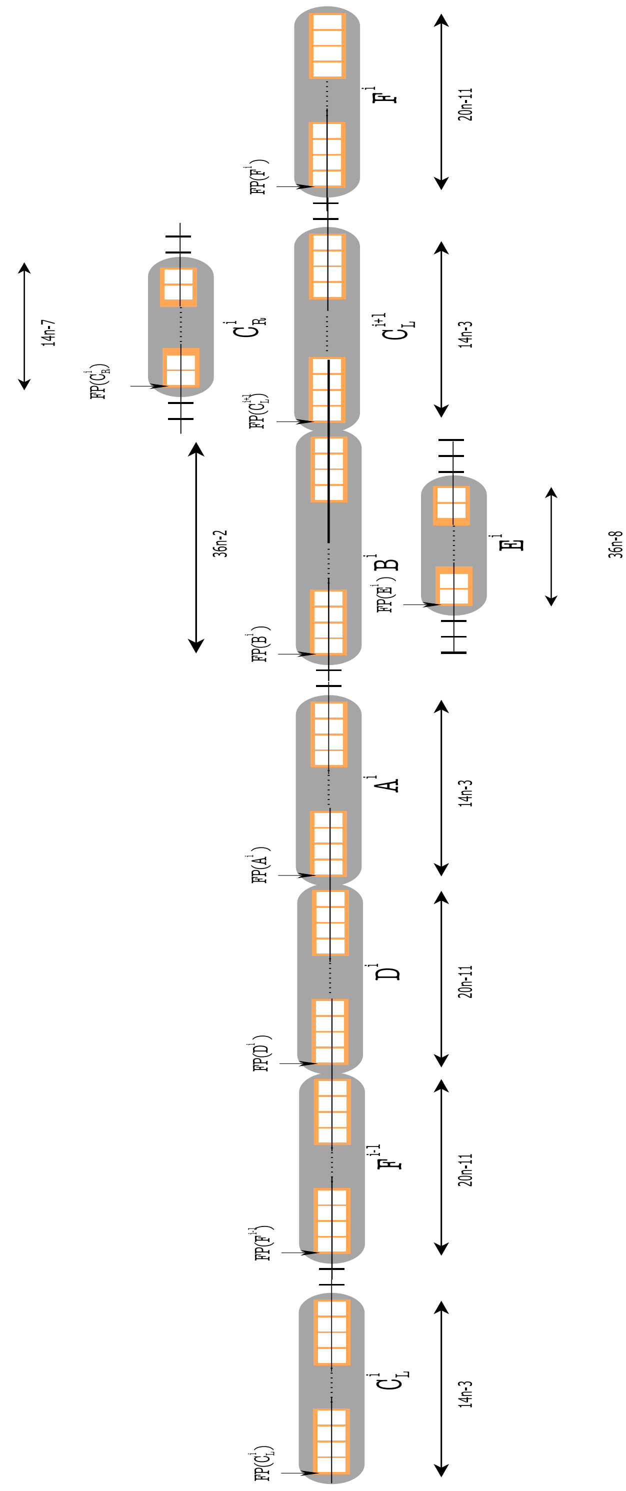

Fig. 12. Schematic representation of the distances between groups of a clause c_i

with $r = 14(k-1), s = 12(k-1), j \in \{1, 2, 3\}, 1 \le k \le n$.

For each clause c_i, in order to represent the clause c_i, we delete from $\mathcal{D}$ the 2-interval $([FP(A^i) + r + 2, FP(A^i) + r + 6], [FP(B^i_j) + s, FP(B^i_j) + s + 4])$ with $r = 14(m-1)$, $s = 12(m-1)$ if x_m is the value of the j^{th} literal of c_i. In a similar way, if $\overline{x_m}$ is the value of the j^{th} literal of c_i, we delete from $\mathcal{D}$ the 2-interval $([FP(A^i)+r+5, FP(A^i)+r+9], [FP(B^i_j)+s+6, FP(B^i_j)+s+10])$ with $r = 14(m-1)$, $s = 12(m-1)$.

Clearly, this construction can be carried out in polynomial-time. We now give an intuitive description of the different elements of the set of 2-intervals that we have built. Block B^i_1 (resp. B^i_2 and B^i_3) represents the value of the first (resp. second and third) literal, say x_m (or $\overline{x_m}$), of the clause c_i; for this, the 2-interval between the simple interval of the m^{th} block of group A^i and the simple interval of B^i_1 (resp. B^i_2 and B^i_3) corresponding to x_m (or $\overline{x_m}$) is not in $\mathcal{D}$ (still the simple intervals are in $\mathsf{GS}(\mathcal{D})$). For instance, in Figure 13, the fact that there is no 2-interval between the simple interval corresponding to $\overline{x_1}$ in B^i_1 and a simple interval of group A^i indicates that the first literal of clause c_i is $\overline{x_1}$. Similarly, the fact that there is no 2-interval between the simple interval corresponding to x_2 (resp. x_3) in B^i_2 (resp. B^i_3) and a simple interval of group A^i indicates that the second (resp. third) literal of clause c_i is x_2 (resp. x_3).

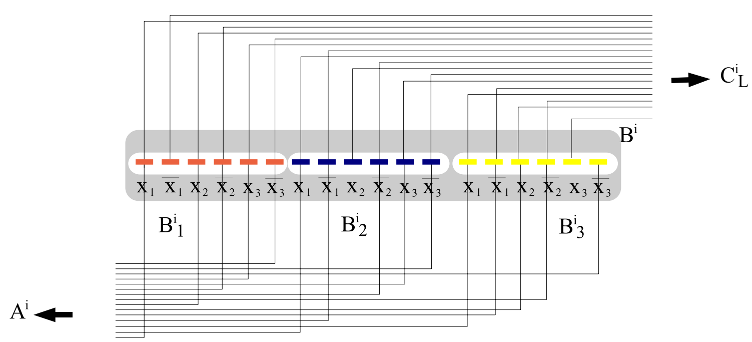

Fig. 13. Zoom on group B^i of the representation of a clause $c_i = (\overline{x_1} \vee x_2 \vee x_3)$

The sequence of blocks (C^{i-1}_R, C^i_L, A^i, B^i, C^i_R) corresponds to a mechanism which propagates the value of each variable of $\mathcal{V}_n$. Blocks (D^i, E^i, F^i) correspond to a literal selecting mechanism that indicates, for each clause c_i, the literal (*i.e.*, the first, second or third) which satisfies c_i. Notice that the two previous intuitive notions will be detailed and clarified afterwards.

We start the proof by giving some properties (Lemmas 8 to 13) about the maximal cardinality of a set of $\{<, \between\}$-comparable 2-intervals in $\mathcal{D}$ in our construction. Then, these results will be used in Proposition 14 to prove the validity of the reduction. In the rest of this paper, we will use the following notations:

- *a 2-interval between blocks X and Y* represents a 2-interval $D = (I, J)$

where I is a simple interval belonging to block X and J is a simple interval belonging to block Y;

- for any $1 \leq i \leq q$ and any set of groups $\alpha \subseteq \{C_L^i, A^i, B^i, C_R^i, D^i, E^i, F^i\}$, $\mathcal{D}(\alpha)$ denotes a set of $\{<, \between\}$-comparable 2-intervals between blocks of groups belonging to α (for example, $\mathcal{D}(D^i, E^i, F^i)$ denotes a set of $\{<, \between\}$-comparable 2-intervals between blocks D_1^i, D_2^i, E_1^i, E_2^i, E_3^i, F_1^i and F_2^i);
- for any $1 \leq i \leq q$, $\mathcal{D}(c_i)$ denotes a set of $\{<, \between\}$-comparable 2-intervals in the representation of clause c_i.

Lemma 7 *For any set of groups α and β, $|\mathcal{D}(\alpha)| + |\mathcal{D}(\beta)| \geq |\mathcal{D}(\alpha \bigcup \beta)|$.*

PROOF. The union of the sets α and β could result in one of the following cases:

(a) $\mathcal{D}(\alpha)$ and $\mathcal{D}(\beta)$ have at least a 2-interval in common;
(b) at least a 2-interval of $\mathcal{D}(\alpha)$ and a 2-interval of $\mathcal{D}(\beta)$ are not disjoint;
(c) at least a 2-interval of $\mathcal{D}(\alpha)$ and a 2-interval of $\mathcal{D}(\beta)$ are not $\{<, \between\}$-comparable.

In case (a) it is clear that the duplicated 2-interval will not be counted more than once in $|\mathcal{D}(\alpha \bigcup \beta)|$. In case (b), only one of the two 2-intervals which are not disjoint can be in $\mathcal{D}(\alpha \bigcup \beta)$. In case (c), only one of the two 2-intervals which are not $\{<, \between\}$-comparable can be in $\mathcal{D}(\alpha \bigcup \beta)$. If none of those three cases occur then, $\mathcal{D}(\alpha) \bigcup \mathcal{D}(\beta)$ is $\{<, \between\}$-comparable, and thus, $|\mathcal{D}(\alpha)| + |\mathcal{D}(\beta)| = |\mathcal{D}(\alpha \bigcup \beta)|$. Therefore, $|\mathcal{D}(\alpha)| + |\mathcal{D}(\beta)| \geq |\mathcal{D}(\alpha \bigcup \beta)|$. $\square$

By construction, a 2-interval can only exist between two blocks that correspond to a single clause (cf. Figure 4). Thus, the maximum cardinality of a set of $\{<, \between\}$-comparable 2-intervals of $\mathcal{D}$ (*i.e.*, the full representation of the boolean formula) can be deduced from the maximum cardinality of $\mathcal{D}(c_i)$ where c_i is a clause of $\mathcal{C}_q$, for any $1 \leq i \leq q$. Precisely, the maximum cardinality of a set of $\{<, \between\}$-comparable 2-intervals in the representation of all the clauses is less than or equal to $q \cdot max_{i \in [1,q]} \ |\mathcal{D}(c_i)|$.

We first compute the maximum cardinality of a set $\mathcal{D}(c_i)$ of $\{<, \between\}$-comparable 2-intervals between blocks corresponding to a single clause c_i.

Lemma 8 *$|\mathcal{D}(\alpha)| \leq 3n$ for $\alpha = \{C_L^i, A^i, B^i, C_R^i\}$.*

PROOF. By the disjunction constraint, at most one simple interval per block of A^i can be involved in a 2-interval between blocks of A^i and B^i. As there are n blocks in A^i, we have $|\mathcal{D}(A^i, B^i)| \leq n$. Similarly, by the disjunction constraint, at most one simple interval per block of C_R^i can be involved in a 2-interval

between blocks of B^i and C^i_R. As there are n blocks in C^i_R, $|\mathcal{D}(B^i, C^i_R)| \leq n$. Thus, according to Lemma 7, $|\mathcal{D}(A^i, B^i, C^i_R)| \leq |\mathcal{D}(A^i, B^i)| + |\mathcal{D}(B^i, C^i_R)| \leq 2n$.
Moreover, at most one simple interval per block of A^i can be involved in a 2-interval between blocks of A^i and C^i_L since the two 2-intervals between a given block of A^i and C^i_L are $\{\sqsubset\}$-comparable. As there are n blocks in A^i, $|\mathcal{D}(C^i_L, A^i)| \leq n$. Thus, by Lemma 7, $|\mathcal{D}(C^i_L, A^i, B^i, C^i_R)| \leq |\mathcal{D}(A^i, B^i, C^i_R)| + |\mathcal{D}(C^i_L, A^i)| \leq 3n$. $\square$

In the following, $\theta(i,j)$ will denote the set of all the simple intervals in B^i_j and E^i_j, with $1 \leq j \leq 3$. The set $\delta(i,j) \subseteq \theta(i,j)$ will denote a set of disjoint simple intervals and $k(E,i,j)$ (resp. $k(B,i,j)$) will be the number of simple intervals of block E^i_j (resp. B^i_j) which are in $\delta(i,j)$. By construction, each simple interval in block E^i_j intersects two simple intervals of block B^i_j (cf. Figure 14 and page 14).

Observation 1 *(a) If $k(E,i,j) > 0$ then at least $k(E,i,j)+1$ simple intervals of block B^i_j cannot belong to $\delta(i,j)$. Thus, $k(B,i,j) \leq 2n - (k(E,i,j)+1)$. Hence, $|\delta(i,j)| \leq k(B,i,j) + k(E,i,j) \leq 2n - (k(E,i,j)+1) + k(E,i,j) \leq 2n - 1$.*

*(b) If $k(E,i,j) = 0$ then all the simple intervals (*i.e.*, $2n$) of block B^i_j can belong to $\delta(i,j)$. Thus, $k(B,i,j) \leq 2n$. Hence, $|\delta(i,j)| \leq k(B,i,j) + k(E,i,j) \leq 2n$.*

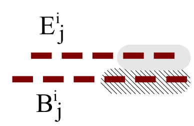

Fig. 14. If two simple intervals of block E^i_j are part of $\delta(i,j)$ then at least three simple intervals of block B^i_j cannot belong to $\delta(i,j)$, and thus $|\delta(i,j)| \leq 2n - 1$.

Lemma 9 *If $|\mathcal{D}(D^i, E^i, F^i)| > 4n - 2$ then $|\mathcal{D}(c_i)| < 7n - 2$.*

PROOF. Assume that $|\mathcal{D}(D^i, E^i, F^i)| = 4n - 2 + \gamma$ with $\gamma > 0$. As each block of group E^i (*i.e.*, E^i_1, E^i_2, E^i_3) is composed of $2n - 1$ simple intervals, there is at least one simple interval in each block of group E^i involved in a 2-interval of $\mathcal{D}(D^i, E^i, F^i)$.
Thus, considering only the simple intervals in groups B^i and E^i, there are at most $6n - 3$ (*i.e.*, $3 \cdot (2n - 1)$ by Observation 1 (a)) disjoint simple intervals. By construction, any 2-interval of $\mathcal{D}(A^i, B^i, C^i_R, D^i, E^i, F^i)$ is composed of a simple interval of either group B^i or E^i. Thus, as there are at most $6n - 3$ disjoint simple intervals in groups B^i and E^i, there are at most $6n - 3$ 2-intervals in $\mathcal{D}(A^i, B^i, C^i_R, D^i, E^i, F^i)$. As $|\mathcal{D}(C^i_L, A^i)| \leq n$ (cf. proof of Lemma 8), by Lemma 7, we can conclude that $|\mathcal{D}(C^i_L, A^i, B^i, C^i_R, D^i, E^i, F^i)| \leq 7n -$

$3 < 7n-2$. Thus, since $|\mathcal{D}(c_i)|$ cannot exceed $|\mathcal{D}(C_L^i, A^i, B^i, C_R^i, D^i, E^i, F^i)|$, if $|\mathcal{D}(D^i, E^i, F^i)| > 4n-2$ then $|\mathcal{D}(c_i)| < 7n-2$. $\square$

Lemma 10 *$|\mathcal{D}(c_i)| \leq 7n-2$. Moreover, if $|\mathcal{D}(c_i)| = 7n-2$ then $|\mathcal{D}(\alpha')| = 4n-2$ for $\alpha' = \{D^i, E^i, F^i\}$ and $|\mathcal{D}(\alpha)| = 3n$ for $\alpha = \{C_L^i, A^i, B^i, C_R^i\}$.*

PROOF. Suppose, aiming to a contradiction, that $|\mathcal{D}(c_i)| > 7n-2$. By Lemma 7, $|\mathcal{D}(c_i)| \leq |\mathcal{D}(D^i, E^i, F^i)| + |\mathcal{D}(C_L^i, A^i, B^i, C_R^i)|$. Thus, $|\mathcal{D}(D^i, E^i, F^i)| + |\mathcal{D}(C_L^i, A^i, B^i, C_R^i)| > 7n-2$. As, by Lemma 8, $|\mathcal{D}(C_L^i, A^i, B^i, C_R^i)| \leq 3n$, we have $|\mathcal{D}(D^i, E^i, F^i)| > 4n-2$. But, by Lemma 9, if $|\mathcal{D}(D^i, E^i, F^i)| > 4n-2$ then $|\mathcal{D}(c_i)| < 7n-2$, a contradiction. Therefore, we have $|\mathcal{D}(c_i)| \leq 7n-2$.

Now, if $|\mathcal{D}(c_i)| = 7n-2$ then, by Lemma 9, $|\mathcal{D}(D^i, E^i, F^i)| \leq 4n-2$. Thus, $|\mathcal{D}(C_L^i, A^i, B^i, C_R^i)| \geq 3n$. But, by Lemma 8, $|\mathcal{D}(C_L^i, A^i, B^i, C_R^i)| \leq 3n$. Therefore, $|\mathcal{D}(C_L^i, A^i, B^i, C_R^i)| = 3n$ and thus $|\mathcal{D}(D^i, E^i, F^i)| = 4n-2$. $\square$

Lemma 11 *If $|\mathcal{D}(c_i)| = 7n-2$ then the set $\mathcal{D}(D^i, E^i, F^i)$ contains 2-intervals built with all the simple intervals from exactly two blocks of group E^i* (i.e., *(E_1^i, E_2^i), (E_1^i, E_3^i) or (E_2^i, E_3^i)).*

PROOF. Since $|\mathcal{D}(c_i)| = 7n-2$, by Lemma 10, we know that $|\mathcal{D}(C_L^i, A^i, B^i, C_R^i)| = 3n$. Moreover, $|\mathcal{D}(C_L^i, A^i)| \leq n$ (cf. proof of Lemma 8). Thus, by Lemma 7, we must have $|\mathcal{D}(A^i, B^i, C_R^i)| \geq 2n$. As $|\mathcal{D}(A^i, B^i, C_R^i)| \leq 2n$ (cf. proof of Lemma 8), $|\mathcal{D}(A^i, B^i, C_R^i)| = 2n$.

Since $|\mathcal{D}(c_i)| = 7n-2$, by Lemma 10, we have $|\mathcal{D}(D^i, E^i, F^i)| = 4n-2$. Moreover, by construction, each 2-interval of $\mathcal{D}(D^i, E^i, F^i)$ is built with a simple interval of E^i. Thus, $\sum_{j=1}^{3}(k(E,i,j)) = 4n-2$.

Suppose, for the sake of contradiction, that $k(E,i,j) > 0$ for all $1 \leq j \leq 3$. By Observation 1, we then have $k(B,i,j) \leq 2n - (k(E,i,j)+1)$ for all $1 \leq j \leq 3$. Thus, $\sum_{j=1}^{3} k(B,i,j) \leq \sum_{j=1}^{3} 2n - (k(E,i,j)+1) \leq 6n-3-\sum_{j=1}^{3} k(E,i,j)$. As $\sum_{j=1}^{3} k(E,i,j) = 4n-2$, we conclude that $\sum_{j=1}^{3} k(B,i,j) \leq 2n-1$. Moreover, by construction, each 2-interval of $\mathcal{D}(A^i, B^i, C_R^i)$ is built with a simple interval of B^i. Therefore, $|\mathcal{D}(A^i, B^i, C_R^i)| \leq 2n-1$, a contradiction.

Therefore at least one of $k(E,i,1)$, $k(E,i,2)$ or $k(E,i,3)$ is equal to 0. Hence, $\mathcal{D}(D^i, E^i, F^i)$ contains 2-intervals built with all the simple intervals from exactly two blocks of the group E^i (*i.e.*, $(E_1^i, E_2^i), (E_1^i, E_3^i)$ or (E_2^i, E_3^i)). $\square$

Corollary 12 *If $|\mathcal{D}(c_i)| = 7n-2$ then the set $\mathcal{D}(A^i, B^i, C_R^i)$ contains all the simple intervals of a unique block of group B^i* (i.e., *B_1^i, B_2^i or B_3^i).*

PROOF. By Lemma 10, if $|\mathcal{D}(c_i)| = 7n - 2$ then $|\mathcal{D}(C_L^i, A^i, B^i, C_R^i)| = 3n$. Moreover, by construction, each 2-interval of $\mathcal{D}(A^i, B^i, C_R^i)$ is built with a simple interval of B^i. As $|\mathcal{D}(A^i, B^i, C_R^i)| = 2n$ (cf. proof of Lemma 11), $\sum_{j=1}^{3}(k(B, i, j)) = 2n$. By Lemma 11, if $|\mathcal{D}(c_i)| = 7n - 2$ then $\mathcal{D}(D^i, E^i, F^i)$ contains 2-intervals built with all the simple intervals from exactly two blocks E_s^i and E_t^i of group E^i, for $1 \leq s, t \leq 3$. By Observation 1, $\mathcal{D}(A^i, B^i, C_R^i)$ contains 2-intervals built with all the simple intervals from exactly one block B_u^i of group B^i with $1 \leq u \leq 3$, $u \neq s$ and $u \neq t$. $\square$

Lemma 13 *If $|\mathcal{D}(c_i)| = 7n - 2$ then:*
(a) if $j = 1$ then $\mathcal{D}(D^i, E^i, F^i)$ is the set of all the 2-intervals between blocks E_2^i, E_3^i, F_1^i and F_2^i.
(b) if $j = 2$ then $\mathcal{D}(D^i, E^i, F^i)$ is the set of all the 2-intervals between blocks E_1^i, E_3^i, D_1^i and F_2^i.
(c) if $j = 3$ then $\mathcal{D}(D^i, E^i, F^i)$ is the set of all the 2-intervals between blocks E_1^i, E_2^i, D_1^i and D_2^i.

PROOF. (a) By Lemma 10, if $|\mathcal{D}(c_i)| = 7n-2$ then $|\mathcal{D}(D^i, E^i, F^i)| = 4n-2$. By Corollary 12, Lemma 11 and the disjunction constraint, if the $2n$ 2-intervals of $\mathcal{D}(A^i, B^i, C_R^i)$ contain 2-intervals built with all the simple intervals from B_1^i, then $\mathcal{D}(D^i, E^i, F^i)$ contains 2-intervals built with all the simple intervals from E_2^i and E_3^i. Thus, $\mathcal{D}(D^i, E^i, F^i)$ is composed of the $2n-1$ 2-intervals between blocks E_3^i and F_2^i. Moreover, any 2-interval between blocks E_2^i and D_2^i is $\{\sqsubset\}$-comparable to any 2-interval between blocks A^i and B_1^i. Therefore, the set $\mathcal{D}(D^i, E^i, F^i)$ of $4n-2$ 2-intervals is also composed of the $2n-1$ 2-intervals between blocks E_2^i and F_1^i.

(b) Similarly to (a), if the $2n$ 2-intervals of $\mathcal{D}(A^i, B^i, C_R^i)$ contain 2-intervals built with all the simple intervals from B_2^i, then $\mathcal{D}(D^i, E^i, F^i)$ contains 2-intervals built with all the simple intervals from E_1^i and E_3^i. Thus, $\mathcal{D}(D^i, E^i, F^i)$ is composed of the $2n-1$ 2-intervals between blocks E_1^i and D_1^i and the $2n-1$ 2-intervals between blocks E_3^i and F_2^i.

(c) Similarly to (a) and (b), if the $2n$ 2-intervals of $\mathcal{D}(A^i, B^i, C_R^i)$ contain 2-intervals built with all the simple intervals from B_3^i, then $\mathcal{D}(D^i, E^i, F^i)$ contains 2-intervals built with all the simple intervals from E_1^i and E_2^i. Thus, $\mathcal{D}(D^i, E^i, F^i)$ is composed of the $2n-1$ 2-intervals between blocks E_1^i and D_1^i. Moreover, any 2-interval between blocks E_2^i and F_1^i is $\{\sqsubset\}$-comparable to any 2-interval between blocks B_3^i and C_R^i. Therefore, $\mathcal{D}(D^i, E^i, F^i)$ is also composed of the $2n-1$ 2-intervals between blocks E_2^i and D_2^i. $\square$

In the following, we denote by $x_m(U, V)$ (resp. $\overline{x_m}(U, V)$), for $1 \leq m \leq n$, the 2-interval composed of the two simple intervals representing x_m (resp. $\overline{x_m}$) in blocks U and V.

Observation 2 *Suppose* $|\mathcal{D}(c_i)| = 7n - 2$.

- *If, for a given* $1 \leq j \leq 3$, $x_m(C^i_L, A^i) \in \mathcal{D}(c_i)$ *then* $x_m(A^i, B^i_j) \in \mathcal{D}(c_i)$.
- *If, for a given* $1 \leq j \leq 3$, $\overline{x_m}(C^i_L, A^i) \in \mathcal{D}(c_i)$ *then* $\overline{x_m}(A^i, B^i_j) \in \mathcal{D}(c_i)$.

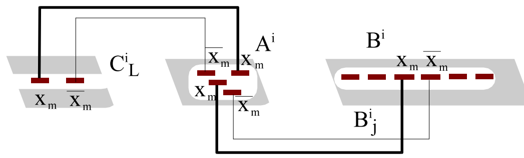

Fig. 15. $x_m(C^i_L, A^i) \in \mathcal{D}(c_i)$ implies $x_m(A^i, B^i_j) \in \mathcal{D}(c_i)$.

PROOF. An illustration of Observation 2 is given in Figure 15. Indeed, $|\mathcal{D}(c_i)| = 7n - 2$, thus by Lemma 10 $|\mathcal{D}(C^i_L, A^i, B^i, C^i_R)| = 3n$. We have proved (cf. proof of Lemma 8) that $|\mathcal{D}(A^i, B^i)| \leq n$, $|\mathcal{D}(B^i, C^i_R)| \leq n$, and $|\mathcal{D}(C^i_L, A^i)| \leq n$. By Lemma 7, $|\mathcal{D}(A^i, B^i)| + |\mathcal{D}(B^i, C^i_R)| + |\mathcal{D}(C^i_L, A^i)| \geq |\mathcal{D}(C^i_L, A^i, B^i, C^i_R)|$. Thus, $|\mathcal{D}(A^i, B^i)| = |\mathcal{D}(B^i, C^i_R)| = |\mathcal{D}(C^i_L, A^i)| = n$.

Moreover, we proved that $|\mathcal{D}(C^i_L, A^i)| = n$ implies that one simple interval per block of A^i is involved in a 2-interval between C^i_L and A^i (cf . proof of Lemma 8). Consider the m^{th} block of A^i. Therefore, by the $\{<, \between\}$-comparability constraint, either $x_m(C^i_L, A^i) \in \mathcal{D}(c_i)$ or $\overline{x_m}(C^i_L, A^i) \in \mathcal{D}(c_i)$.

Similarly, we proved that $|\mathcal{D}(A^i, B^i)| = n$ implies that one simple interval per block of A^i is involved in a 2-interval between A^i and B^i (cf . proof of Lemma 8). Consider the m^{th} block of A^i. We mentioned that, by construction, the simple intervals of this block represent in order $(\overline{x_m}, x_m, \overline{x_m}, x_m)$.Therefore, either $x_m(A^i, B^i_j) \in \mathcal{D}(c_i)$ or $\overline{x_m}(A^i, B^i_j) \in \mathcal{D}(c_i)$.

Moreover, by the disjunction constraint and the adjustment of the simple intervals of each block of A^i, if $x_m(C^i_L, A^i) \in \mathcal{D}(c_i)$ then $x_m(A^i, B^i_j) \in \mathcal{D}(c_i)$. Similarly, if $\overline{x_m}(C^i_L, A^i) \in \mathcal{D}(c_i)$ then $\overline{x_m}(A^i, B^i_j) \in \mathcal{D}(c_i)$. $\square$

Observation 3 *Suppose* $|\mathcal{D}(c_i)| = 7n - 2$.

- *If, for a given* $1 \leq j \leq 3$, $x_m(A^i, B^i_j) \in \mathcal{D}(c_i)$ *then* $\overline{x_m}(B^i_j, C^i_R) \in \mathcal{D}(c_i)$.
- *If, for a given* $1 \leq j \leq 3$, $\overline{x_m}(A^i, B^i_j) \in \mathcal{D}(c_i)$ *then* $x_m(B^i_j, C^i_R) \in \mathcal{D}(c_i)$.

PROOF. An illustration of Observation 3 is given in Figure 16. Suppose $x_m(A^i, B^i_{j_0}) \in \mathcal{D}(c_i)$ for a given $1 \leq j_0 \leq 3$. By Corollary 12, as $|\mathcal{D}(c_i)| = 7n-2$, the set $\mathcal{D}(A^i, B^i, C^i_R)$ contains all the simple intervals of a unique block B^i_j of group B^i. Thus, by the supposition we made, the set $\mathcal{D}(A^i, B^i, C^i_R)$ contains all the simple intervals of block $B^i_{j_0}$. We proved (cf. proof of Observation 2) that

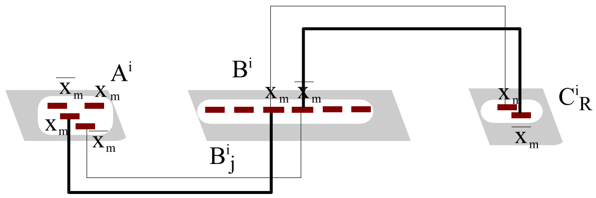

Fig. 16. $x_m(A^i, B^i_j) \in \mathcal{D}(c_i)$ implies $\overline{x_m}(B^i_j, C^i_R) \in \mathcal{D}(c_i)$.

either $x_m(A^i, B^i_{j_0}) \in \mathcal{D}(c_i)$ or $\overline{x_m}(A^i, B^i_{j_0}) \in \mathcal{D}(c_i)$ for some $1 \le j_0 \le 3$. By the disjunction constraint, as $x_m(A^i, B^i_{j_0}) \in \mathcal{D}(c_i)$ we have $x_m(B^i_{j_0}, C^i_R) \notin \mathcal{D}(c_i)$. Moreover, as the set $\mathcal{D}(A^i, B^i, C^i_R)$ contains all the simple intervals of block $B^i_{j_0}$, $\overline{x_m}(B^i_{j_0}, C^i_R) \in \mathcal{D}(c_i)$. Similarly, if $\overline{x_m}(A^i, B^i_{j_0}) \in \mathcal{D}(c_i)$ then $x_m(B^i_{j_0}, C^i_R) \in \mathcal{D}(c_i)$ for any $1 \le j_0 \le 3$. $\square$

Observation 4 *Suppose* $|\mathcal{D}(c_i)| = |\mathcal{D}(c_{i+1})| = 7n - 2$.

- *If, for a given* $1 \le j \le 3$, $x_m(B^i_j, C^i_R) \in \mathcal{D}(c_i)$ *then* $\overline{x_m}(C^{i+1}_L, A^{i+1}) \in \mathcal{D}(c_{i+1})$.
- *If, for a given* $1 \le j \le 3$, $\overline{x_m}(B^i_j, C^i_R) \in \mathcal{D}(c_i)$ *then* $x_m(C^{i+1}_L, A^{i+1}) \in \mathcal{D}(c_{i+1})$.

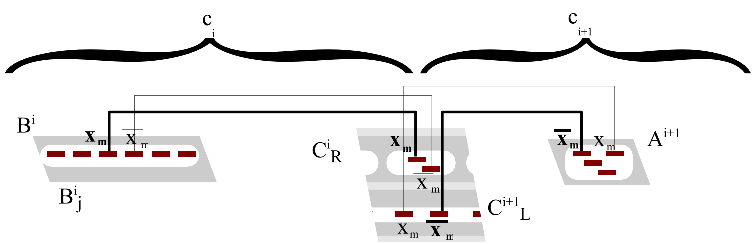

Fig. 17. $x_m(B^i_j, C^i_R) \in \mathcal{D}(c_i)$ implies $\overline{x_m}(C^{i+1}_L, A^{i+1}) \in \mathcal{D}(c_{i+1})$

PROOF. An illustration of Observation 4 is given in Figure 17. If $|\mathcal{D}(c_{i+1})| = 7n - 2$, then $|\mathcal{D}(C^{i+1}_L, A^{i+1})| = n$ (cf. proof of Observation 2). By the $\{<, \between\}$-comparability constraint, either $x_m(C^{i+1}_L, A^{i+1}) \in \mathcal{D}(c_{i+1})$ or $\overline{x_m}(C^{i+1}_L, A^{i+1}) \in \mathcal{D}(c_{i+1})$ (cf. proof of Observation 2). By the adjustment of blocks C^i_R and C^{i+1}_L, if $|\mathcal{D}(c_i)| = |\mathcal{D}(c_{i+1})| = 7n-2$ and $x_m(B^i_j, C^i_R) \in \mathcal{D}(c_i)$, then $\overline{x_m}(C^{i+1}_L, A^{i+1}) \in \mathcal{D}(c_{i+1})$. Similarly, if $|\mathcal{D}(c_i)| = |\mathcal{D}(c_{i+1})| = 7n - 2$ and $\overline{x_m}(B^i_j, C^i_R) \in \mathcal{D}(c_i)$ then $x_m(C^{i+1}_L, A^{i+1}) \in \mathcal{D}(c_{i+1})$. $\square$

Lemmas 8 to 13 together with Observations 2 to 4 provide us all the necessary intermediate results to show that the reduction of Exact 3-CNF-Sat to the 2-IP-Unit-$\{<, \between\}$ problem is valid.

Proposition 14 *Given an instance of the problem* Exact 3-CNF-Sat *with* n *variables and* q *clauses, there exists a satisfying true assignment iff there is* *a subset $\mathcal{D}' \subseteq \mathcal{D}$ such that $|\mathcal{D}'| \geq (7n-2)q$ and $\mathcal{D}'$ is $\{<, \between\}$-comparable.*

PROOF. $(\Rightarrow)$
Suppose we have an assignment AS of the n variables that satisfies the boolean formula. By definition, for each clause there is at least one literal that satisfies it. We look for a set of $\{<, \between\}$-comparable 2-intervals $\mathcal{D}'$ in the representation of the boolean formula such that the cardinality of $\mathcal{D}'$ is greater than or equal to $(7n-2)q$. By Lemma 10, for any clause c_i, $|\mathcal{D}(c_i)| \leq 7n-2$. Thus, $|\mathcal{D}'| \leq (7n-2)q$. Therefore, the only solution to our problem is a set $\mathcal{D}'$ such that $|\mathcal{D}'| = (7n-2)q$. As the boolean formula is composed of q clauses, each subset $\mathcal{D}'(c_i)$ of $\mathcal{D}'$ for each clause c_i, $1 \leq i \leq q$, must satisfy $|\mathcal{D}'(c_i)| = 7n-2$.

Hereafter, j_i will define the smallest index of the literal of c_i (*i.e.*, 1, 2 or 3) which, by its assignment, satisfies c_i. For any $1 \leq i \leq q$, we define $\mathcal{D}'(c_i)$ as follows. For each variable x_m with $1 \leq m \leq n$:

(a) If $x_m = True$ then $\overline{x_m}(C_L^i, A^i)$, $\overline{x_m}(A^i, B_{j_i}^i)$ and $x_m(B_{j_i}^i, C_R^i)$ are in $\mathcal{D}'(c_i)$;
(b) If $x_m = False$ then $x_m(C_L^i, A^i)$, $x_m(A^i, B_{j_i}^i)$ and $\overline{x_m}(B_{j_i}^i, C_R^i)$ are in $\mathcal{D}'(c_i)$.

Moreover, for any given $1 \leq j_i \leq 3$:

(c) If $j_i = 1$ then $\mathcal{D}'(c_i)$ is also composed of the set of all the 2-intervals between blocks E_2^i, E_3^i, F_1^i and F_2^i;
(d) If $j_i = 2$ then $\mathcal{D}'(c_i)$ is also composed of the set of all the 2-intervals between blocks E_1^i, E_3^i, D_1^i and F_2^i;
(e) If $j_i = 3$ then $\mathcal{D}'(c_i)$ is also composed of the set of all the 2-intervals between blocks E_1^i, E_2^i, D_1^i and D_2^i.

An example of subset $\mathcal{D}'(c_i)$ where $c_i = (\overline{x_1} \vee x_2 \vee x_3)$ and such that $x_1 = x_2 = x_3 = True$ is illustrated in Figure 18.

In the following, we will first prove that, for any $1 \leq i \leq q$, $\mathcal{D}'(c_i)$ is a set of $\{<, \between\}$-comparable 2-intervals. Then we will prove that $\mathcal{D}' = \bigcup_1^q \mathcal{D}'(c_i)$ is a set of $\{<, \between\}$-comparable 2-intervals such that $|\mathcal{D}'| = (7n-2)q$.

By the way we defined $\mathcal{D}'(c_i)$, it is easy to see that $|\mathcal{D}'(c_i)| = 7n-2$. Indeed, by **(a)** or **(b)**, three 2-intervals have been added to $\mathcal{D}'(c_i)$ for each variable x_m with $1 \leq m \leq n$. Moreover, by **(c)**, **(d)** or **(e)**, for any given $1 \leq j_i \leq 3$, a set of $4n-2$ 2-intervals has been added to $\mathcal{D}'(c_i)$.

For any $1 \leq i \leq q$, $\mathcal{D}'(c_i)$ is a set of $\{<, \between\}$-comparable 2-intervals iff there is no inclusion or disjunction in $\mathcal{D}'(c_i)$. First, we will prove that given a $1 \leq j_i \leq 3$, $\mathcal{D}'(C_L^i, A^i, B_{j_i}^i, C_R^i)$ is a set of $\{<, \between\}$-comparable 2-intervals. Then, we will prove that given a $1 \leq j_i \leq 3$, $\mathcal{D}'(D^i, E^i, F^i)$ is a set of $\{<, \between\}$-comparable 2-intervals. Finally, we will prove that $\mathcal{D}'(c_i)$, which is the union of those two

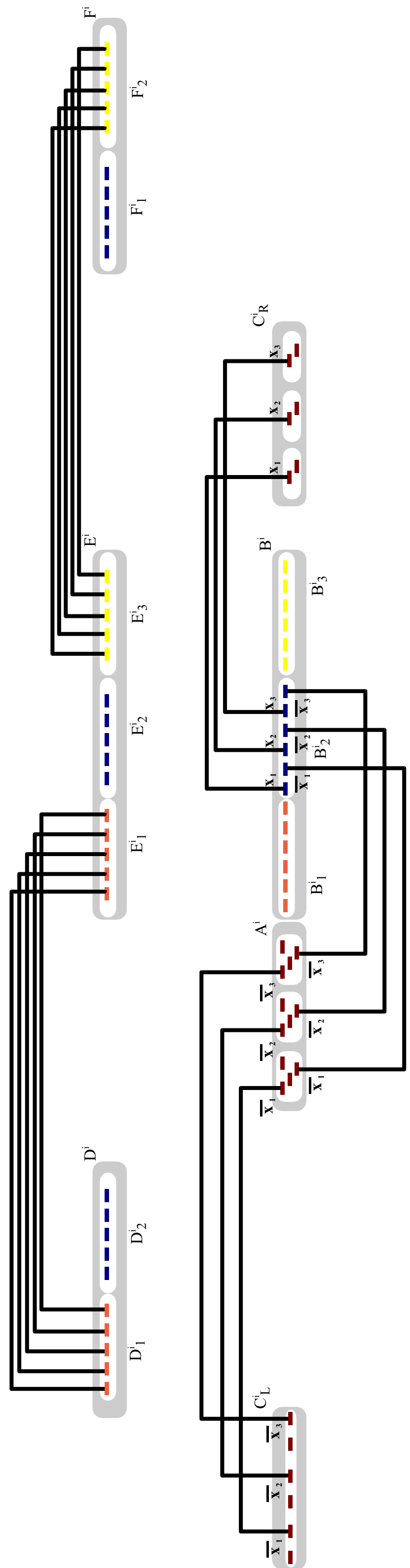

Fig. 18. $\mathcal{D}'(c_i)$ where $c_i = (\overline{x_1} \vee x_2 \vee x_3)$ and $x_1 = x_2 = x_3 = True$

sets, is a set of $\{<, \between\}$-comparable 2-intervals.

Considering only the 2-intervals of $\mathcal{D}'(C_L^i, A^i, B_{j_i}^i, C_R^i)$, by construction an inclusion can only occur between two 2-intervals built with simple intervals of exactly two groups. For any $1 \leq j_i \leq 3$, by construction, any pair of 2-intervals between A^i and $B_{j_i}^i$ (resp. $B_{j_i}^i$ and C_R^i) are crossing. Thus, an inclusion can only occur when two simple intervals of the same block of A^i are both involved in a 2-interval between C_L^i and A^i in $\mathcal{D}'(C_L^i, A^i, B_{j_i}^i, C_R^i)$.

Clearly, either $\overline{x_m}(C_L^i, A^i) \in \mathcal{D}'(c_i)$ or $x_m(C_L^i, A^i) \in \mathcal{D}'(c_i)$ for each variable x_m. Thus, only one simple interval per block of A^i is involved in a 2-interval between C_L^i and A^i. Therefore, there cannot be an inclusion in $\mathcal{D}'(C_L^i, A^i, B_{j_i}^i, C_R^i)$.

By the way we defined $\mathcal{D}'(c_i)$ and the construction of the representation of a clause, it is easy to see that there cannot be non disjoint 2-intervals in $\mathcal{D}'(C_L^i, A^i, B_{j_i}^i, C_R^i)$ (see for instance Figure 18). Thus, $\mathcal{D}'(C_L^i, A^i, B_{j_i}^i, C_R^i)$ is a set of $3n$ $\{<, \between\}$-comparable 2-intervals.

Considering only the 2-intervals of $\mathcal{D}'(D^i, E^i, F^i)$, by construction, there cannot be a problem of inclusion in $\mathcal{D}'(D^i, E^i, F^i)$. Moreover, a problem of disjunction can only occur when a simple interval of block E_2^i is involved in two 2-intervals in $\mathcal{D}'(D^i, E^i, F^i)$. By the way we defined $\mathcal{D}'(c_i)$, this situation never appears. Thus, $\mathcal{D}'(D^i, E^i, F^i)$ is a set of $4n-2$ $\{<, \between\}$-comparable 2-intervals.

Now we consider the 2-intervals of $\mathcal{D}'(c_i)$. We proved upwards that for any $1 \leq j_i \leq 3$, both $\mathcal{D}'(C_L^i, A^i, B_{j_i}^i, C_R^i)$ and $\mathcal{D}'(D^i, E^i, F^i)$ are sets of $\{<, \between\}$-comparable 2-intervals. Thus, we have to check that assembling those two sets does not create inclusion or disjunction problems. To prove that $\mathcal{D}'(c_i)$ is a set of $\{<, \between\}$-comparable 2-intervals, we will examine the three following cases:

(1) $j_i = 1$. $\mathcal{D}'(c_i)$ contains n 2-intervals between C_L^i and A^i, n 2-intervals between A^i and B_1^i, n 2-intervals between B_1^i and C_R^i, $2n-1$ 2-intervals between E_2^i and F_1^i and $2n-1$ 2-intervals between E_3^i and F_2^i.
   By construction, all the 2-intervals are disjoint. Moreover, any 2-interval between E_2^i and F_1^i (resp. E_3^i and F_2^i) is crossing any 2-interval between B_1^i and C_R^i (see Figure 19). Thus, there is no inclusion problem in $\mathcal{D}'(c_i)$. Thus, $\mathcal{D}'(c_i)$ is a set of $7n-2$ $\{<, \between\}$-comparable 2-intervals in this case.

(2) $j_i = 2$. $\mathcal{D}'(c_i)$ contains n 2-intervals between C_L^i and A^i, n 2-intervals between A^i and B_2^i, n 2-intervals between B_2^i and C_R^i, $2n-1$ 2-intervals between D_1^i and E_1^i and $2n-1$ 2-intervals between E_3^i and F_2^i.
   By construction, all the 2-intervals are disjoint. Moreover, any 2-interval between D_1^i and E_1^i is crossing any 2-interval between C_L^i and A^i (resp. A^i and B_2^i). Moreover, any 2-interval between E_3^i and F_2^i is crossing any 2-interval between B_2^i and C_R^i (see Figure 20). Thus, $\mathcal{D}'(c_i)$ is a set of $7n-2$ $\{<, \between\}$-comparable 2-intervals in this case.

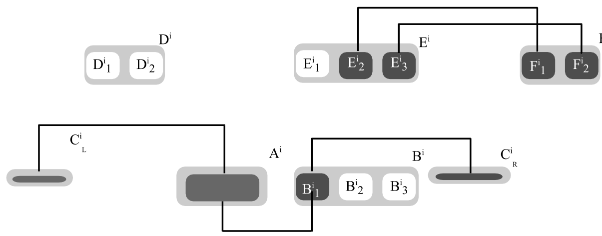

Fig. 19. Illustration of case (1). Bold lines represents sets of 2-intervals between groups.

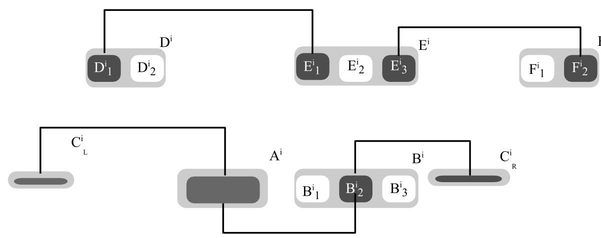

Fig. 20. Illustration of case (2). Bold lines represents sets of 2-intervals between groups.

(3) $j_i = 3$. $\mathcal{D}'(c_i)$ contains n 2-intervals between C_L^i and A^i, n 2-intervals between A^i and B_3^i, n 2-intervals between B_3^i and C_R^i, $2n-1$ 2-intervals between D_1^i and E_1^i and $2n-1$ 2-intervals between D_2^i and E_2^i.

By construction, all the 2-intervals are disjoint. Moreover, any 2-interval between D_1^i and E_1^i (resp. D_2^i and E_2^i) is crossing any 2-interval between C_L^i and A^i. Similarly, any 2-interval between D_1^i and E_1^i (resp. D_2^i and E_2^i) is crossing any 2-interval between A^i and B_3^i (see Figure 21). Thus, $\mathcal{D}'(c_i)$ is a set of $7n-2$ $\{<, \between\}$-comparable 2-intervals in this case.

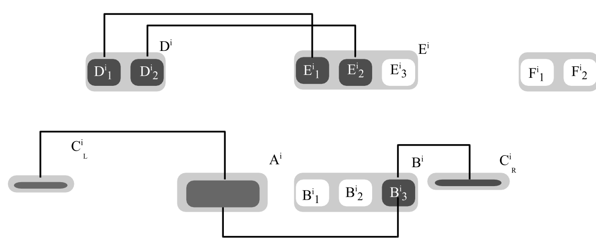

Fig. 21. Illustration of case (3). Bold lines represents sets of 2-intervals between groups.

We just proved that we can find a $\{<, \between\}$-comparable subset $\mathcal{D}(c_i)$ of $\mathcal{D}'$ for each clause c_i such that $|\mathcal{D}(c_i)| = 7n-2$. Finally, we have to verify that $\mathcal{D}' = \bigcup_1^q \mathcal{D}'(c_i)$ is $\{<, \between\}$-comparable. By construction, there cannot be inclusion problems between two 2-intervals of different clauses. What is left is to prove that the adjustment of blocks C_R^i and C_L^{i+1} for a any $1 \leq i < q$ does not imply non disjoint 2-intervals (see Figure 3).

By the adjustment of blocks C_L^{i+1} and C_R^i, a disjunction problem can only occur between the simple interval representing x_m (resp. $\overline{x_m}$) in C_R^i and the simple interval representing x_m (resp. $\overline{x_m}$) in C_L^{i+1} for some $1 \leq m \leq n$.

By the way we defined $\mathcal{D}'(c_i)$, if $x_m = True$ then for any $1 \leq i \leq q$, $\overline{x_m}(C_L^i, A^i)$ and $x_m(B_{j_i}^i, C_R^i)$ are in $\mathcal{D}'(c_i)$. Thus, if $x_m = True$ then $x_m(B_{j_i}^i, C_R^i) \in \mathcal{D}'(c_i)$ and $\overline{x_m}(C_L^{i+1}, A^{i+1}) \in \mathcal{D}'(c_{i+1})$. However, we know that, for any $1 \leq j_i \leq 3$, $x_m(B_{j_i}^i, C_R^i)$ and $\overline{x_m}(C_L^{i+1}, A^{i+1})$ are disjoint (see Figure 3).

By the way we defined $\mathcal{D}'(c_i)$, if $x_m = False$ then for any $1 \leq i \leq q$, $x_m(C_L^i, A^i)$ and $\overline{x_m}(B_{j_i}^i, C_R^i)$ are in $\mathcal{D}'(c_i)$. Thus, if $x_m = False$ then $\overline{x_m}(B_{j_i}^i, C_R^i) \in \mathcal{D}'(c_i)$ and $x_m(C_L^{i+1}, A^{i+1}) \in \mathcal{D}'(c_{i+1})$. However, we know that, for any $1 \leq j_i \leq 3$, $\overline{x_m}(B_{j_i}^i, C_R^i)$ and $x_m(C_L^{i+1}, A^{i+1})$ are disjoint (see Figure 3).

Thus, a disjunction problem due to the adjustment of blocks C_L^{i+1} and C_R^i for a given $1 \leq i < q$ in $\mathcal{D}'$ cannot exist. Therefore, there is a set of $\{<, \between\}$-comparable 2-intervals in the representation of the boolean formula of cardinality $(7n-2)q$.

($\Leftarrow$)
Suppose we have a $\{<, \between\}$-comparable subset $\mathcal{D}' \subseteq \mathcal{D}$ of cardinality $(7n-2)q$. By Lemma 10, $\mathcal{D}'$ is composed of a subset $\mathcal{D}'(c_i)$ of at most $7n-2$ $\{<, \between\}$-comparable 2-intervals for each clause c_i with $1 \leq i \leq q$. Thus, for each $1 \leq i \leq q$, $|\mathcal{D}'(c_i)| = 7n-2$. We define the assignment AS of the n variables as follows. For any $1 \leq m \leq n$:

- If $\overline{x_m}(C_L^1, A^1) \in \mathcal{D}'$ then the value of variable x_m is $True$;
- If $x_m(C_L^1, A^1) \in \mathcal{D}'$ then the value of variable x_m is $False$.

We proved (cf. proof of Observation 2) that for any $1 \leq i \leq q$ if $|\mathcal{D}(c_i)| = 7n-2$ then $|\mathcal{D}(C_L^i, A^i)| = n$. Thus, as $|\mathcal{D}'(c_1)| = 7n-2$, $\mathcal{D}'(c_1)$ is composed of n 2-intervals between blocks of C_L^1 and A^1. Moreover, we proved (cf. proof of Observation 2) that, for any $1 \leq i \leq q$, if $|\mathcal{D}(c_i)| = 7n-2$ then either $x_m(C_L^i, A^i) \in \mathcal{D}(c_i)$ or $\overline{x_m}(C_L^i, A^i) \in \mathcal{D}(c_i)$. Thus, either $x_m(C_L^1, A^1) \in \mathcal{D}'(c_1)$ or $\overline{x_m}(C_L^1, A^1) \in \mathcal{D}'(c_1)$. Therefore, AS is an assignment of n variables such that each variable have a unique value.

Now, we have to verify that AS satisfies the boolean formula corresponding to $\mathcal{D}$ (*i.e.*, each clause c_i with $1 \leq i \leq q$ must be satisfied). First, note that a direct consequence of Observations 2 to 4 is that, for any $1 \leq m \leq n$, if $x_m(C_L^i, A^i) \in \mathcal{D}(c_i)$, then $x_m(C_L^{i+1}, A^{i+1}) \in \mathcal{D}(c_{i+1})$ for any $1 \leq i < q$. Similarly, for any $1 \leq m \leq n$, if $\overline{x_m}(C_L^i, A^i) \in \mathcal{D}(c_i)$, then $\overline{x_m}(C_L^{i+1}, A^{i+1}) \in \mathcal{D}(c_{i+1})$ for any $1 \leq i < q$.

Thus, for any $1 \leq m \leq n$ if $x_m(C_L^1, A^1) \in \mathcal{D}'(c_1)$ then $x_m(C_L^i, A^i) \in \mathcal{D}'(c_i)$ for any $2 \leq i \leq q$. Similarly, for any $1 \leq m \leq n$ if $\overline{x_m}(C_L^1, A^1) \in \mathcal{D}'(c_1)$ then $\overline{x_m}(C_L^i, A^i) \in \mathcal{D}'(c_i)$ for any $2 \leq i \leq q$.

By Corollary 12, as $|\mathcal{D}'(c_i)| = 7n-2$, the set $\mathcal{D}'(c_i)$ contains all the simple intervals of a unique block $B^i_{j_i}$ of group B^i, for a given $1 \leq j_i \leq 3$. Moreover, as $|\mathcal{D}'(c_i)| = 7n-2$, $\mathcal{D}'(c_i)$ is composed of n 2-intervals between blocks A^i and $B^i_{j_i}$ (cf. proof of Observation 2). More precisely, for any $1 \leq m \leq n$, either $x_m(A^i, B^i_{j_i})$ or $\overline{x_m}(A^i, B^i_{j_i})$ is in $\mathcal{D}'(c_i)$.

Suppose x_p is the literal of clause c_i at position j_i, with $1 \leq j_i \leq 3$. Then by construction, $x_p(A^i, B^i_{j_i})$ does not exist. This implies that $\overline{x_p}(A^i, B^i_{j_i}) \in \mathcal{D}'(c_i)$.

Moreover, by Observations 2 and 3, if $\overline{x_p}(A^i, B^i_{j_i}) \in \mathcal{D}'(c_i)$ then $x_p(B^i_{j_i}, C^i_R) \in \mathcal{D}'(c_i)$ and $\overline{x_p}(C^{i+1}_L, A^{i+1}) \in \mathcal{D}'(c_{i+1})$. Therefore, according to AS, if $\overline{x_p}(C^{i+1}_L, A^{i+1}) \in \mathcal{D}'(c_{i+1})$ then the value of variable x_p is $True$. Thus, as x_p is the literal of clause c_i at position j_i, we conclude that c_i is satisfied.

Suppose $\overline{x_p}$ is the literal of clause c_i at position j_i, with $1 \leq j_i \leq 3$. By a similar reasoning, we can verify that clause c_i is satisfied due to the literal $\overline{x_p}$ at position j_i.

This reasoning can be applied to any clause c_i of the boolean formula. Thus, AS satisfies each clause c_i, $1 \leq i \leq q$. Thus, from the $\{<, \between\}$-comparable subset $\mathcal{D}' \subseteq \mathcal{D}$ of cardinality equal to $(7n-2)q$, we can find a satisfying true assignment AS. $\square$

6 A fixed-parameter algorithm for 2-IP-Unit-$\{<, \between\}$

According to Theorem 6, finding the largest $\{<, \between\}$-comparable subset in a set of 2-intervals on a unit ground set is an **NP**-complete problem. In this section, we give an exact algorithm for that problem with strong emphasis on the crossing structure of the set of 2-intervals. More precisely, we consider the time complexity of the problem with respect to the *forward crossing number* of the input. Indeed, in the context of 2-intervals, one may reasonably expect the forward crossing number to be small compared to the number of 2-intervals, and hence, a natural direction seems to be the question for the fixed-parameter tractability with respect to parameter $\mathsf{FCrossing}(\mathcal{D})$. In response to that question, we show that the problem can be solved for any ground set by means of dynamic programming in $O(n^2 \cdot \mathsf{FCrossing}(\mathcal{D}) \cdot 2^{\mathsf{FCrossing}(\mathcal{D})}(\log(n) + \mathsf{FCrossing}(\mathcal{D})))$ time where n is the number of 2-intervals in $\mathcal{D}$, and hence is fixed-parameter tractable with respect to parameter $\mathsf{FCrossing}(\mathcal{D})$.

For any $D_i \in \mathcal{D}$, let $T(D_i)$ denote the size of the largest $\{<, \between\}$-comparable subset $\mathcal{D}' \subseteq \mathcal{D}$ of which the 2-interval D_i is the rightmost element. Furthermore, for any $D_i, D_j \in \mathcal{D}$ such that $D_j \between D_i$, let $T(D_j \mid D_i)$ denotes the size of the largest $\{<, \between\}$-comparable subset $\mathcal{D}' \subseteq \mathcal{D}$ such that (1) the 2-interval

D_j is the rightmost element of $\mathcal{D}'$ and (2) the 2-interval D_i is not part of the subset $\mathcal{D}'$ but can safely be added to $\mathcal{D}'$ to obtain a new $\{<, \between\}$-comparable subset of size $|\mathcal{D}'| + 1$.

Clearly, a maximum cardinality $\{<, \between\}$-comparable subset $\mathcal{D}' \subseteq \mathcal{D}$ of which the 2-interval D_i is the rightmost element can be obtained either (1) by adding D_i to a maximum cardinality $\{<, \between\}$-comparable subset $\mathcal{D}'' \subseteq \mathcal{D}$ whose rightmost 2-interval D_j precedes the 2-interval D_i, *i.e.*, $D_j < D_i$, or (2) by adding D_i to a maximum cardinality $\{<, \between\}$-comparable subset $\mathcal{D}'' \subseteq \mathcal{D}$ whose rightmost 2-interval D_j crosses the 2-interval D_i, *i.e.*, $D_j \between D_i$, and such that D_i crosses or precedes any 2-interval of $\mathcal{D}''$. Here is another way of stating these observations:

$$\forall D_i \in \mathcal{D}, \qquad T(D_i) = 1 + \max \begin{cases} \max\{T(D_j) : D_j < D_i\} \\ \max\{T(D_j \mid D_i) : D_j \between D_i\} \end{cases} \tag{1}$$

What is left is thus to compute $T(D_j \mid D_i)$. To this aim, we extend the notation $T(D_j \mid D_i)$ as follows: for any $\{\between\}$-comparable subset $\{D_{i_1}, D_{i_2}, \ldots, D_{i_k}\} \subseteq \mathcal{D}$, $k \geq 1$, satisfying $\mathsf{Right}(D_{i_1}) < \mathsf{Right}(D_{i_2}) < \ldots < \mathsf{Right}(D_{i_k})$, we let $T(D_{i_1} \mid D_{i_2}, \ldots, D_{i_k})$ stand for the size of a largest $\{<, \between\}$-comparable subset $\mathcal{D}' \subseteq \mathcal{D}$ such that (1) the 2-interval D_{i_1} is the rightmost element of $\mathcal{D}'$ and (2) the 2-intervals $\{D_{i_2}, D_{i_3}, \ldots, D_{i_k}\}$ are not part of the subset $\mathcal{D}'$ but can safely be added to $\mathcal{D}'$ to obtain a new $\{<, \between\}$-comparable subset of size $T(D_{i_1} \mid D_{i_2}, \ldots, D_{i_k}) + k - 1$. A straightforward extension of the calculation (1) yields the following recurrence relation for computing the entry $T(D_{i_1} \mid D_{i_2}, \ldots, D_{i_k})$ of the dynamic programming table:

$$\begin{aligned} T(D_{i_1} \mid D_{i_2}, \ldots, D_{i_k}) = 1 + & \\ \max & \begin{cases} \max\,\{T(D_j) \mid D_j \text{ satisfies condition (1)}\} \\ \max\,\{T(D_j \mid D_{i_1}) \mid D_j \text{ satisfies condition (2)}\} \\ \max\,\{T(D_j \mid D_{i_1}, D_{i_2}) \mid D_j \text{ satisfies condition (3)}\} \\ \quad\vdots \\ \max\,\{T(D_j \mid D_{i_1}, D_{i_2}, \ldots, D_{i_k}) \mid D_j \text{ satisfies condition } (k+1)\} \end{cases} \end{aligned} \tag{2}$$

where condition (i), $1 \leq i \leq k+1$, is defined as follows:

$$\text{condition } (i) \qquad \begin{cases} D_j \between D_{i_r} & \text{for all } 0 < r < i & \text{(crossing conditions)} \\ D_j < D_{i_s} & \text{for all } i \leq s < k+1 & \text{(precedence conditions)} \end{cases}$$

An illustration of the different conditions of this recurrence relation is given in Figure 22. It follows from the above recurrence relation that entries of the form $T(D_i \mid *)$ depend only on entries of the form $T(D_j \mid *)$ where $D_j < D_i$ or $D_j \between D_i$. From a computational point of view, this implies that the calculation of entries of the form $T(D_i \mid *)$ depends only on the calculation of entries of the form $T(D_j \mid *)$ where $\mathsf{Right}(D_j) < \mathsf{Right}(D_i)$. The following easy lemma gives an upper-bound on the size of the dynamic programming table T with respect to the forward crossing number of $\mathcal{D}$.

Lemma 15 *The number of distinct entries of the dynamic programming table T is upper-bounded by $|\mathcal{D}| \cdot 2^{\mathsf{FCrossing}(\mathcal{D})}$.*

PROOF. For any 2-interval $D_i \in \mathcal{D}$, the number of distinct $\{\between\}$-comparable subsets of which D_i is the leftmost element is upper-bounded by $2^{\mathsf{FCrossing}(\mathcal{D})}$, and hence there exist at most $2^{\mathsf{FCrossing}(\mathcal{D})}$ distinct entries of the form $T(D_i \mid *)$ in the dynamic programming table T. $\square$

The overall algorithm for finding the size of the largest $\{<, \between\}$-comparable subset in a set of 2-intervals is given in Figure 23. Using a suitable data structure for efficiently searching 2-intervals, we have the following result.

Proposition 16 *Algorithm* Max $\{<, \between\}$-Comparable 2-Interval Pattern *returns the size of a maximum cardinality $\{<, \between\}$-comparable subset of a set of 2-intervals $\mathcal{D}$ in $O(n^2 \cdot \mathsf{FCrossing}(\mathcal{D}) \cdot 2^{\mathsf{FCrossing}(\mathcal{D})}(\log(n) + \mathsf{FCrossing}(\mathcal{D})))$ time, where n is the number of 2-intervals in $\mathcal{D}$.*

Our approach is based on the following theorem.

Theorem 17 ([10]) *Let $\mathcal{I}$ be a finite collection of n intervals on the real line. A data structure storing $\mathcal{I}$ using $O(n \log n)$ space can be constructed in $O(n \log n)$ time. By querying this data structure one can report those intervals in $\mathcal{I}$ that are completely contained in a given interval in $O(n \log n + k)$ time where k is the number of reported 2-intervals.*

Lemma 18 *Let $\mathcal{D}$ be a finite collection of n 2-intervals. After a preprocessing stage which takes $O(n \log n)$ time and uses $O(n \log n)$ space, one can report*

(1) those 2-intervals in $\mathcal{D}$ that lie entirely to the left of a given 2-interval, or
(2) those 2-intervals in $\mathcal{D}$ whose left and right intervals are completely contained in two given intervals

in $O(n \log n + k)$ time where k is the number of reported 2-intervals.

PROOF. We use a data structure composed of two separate data structures as defined in Theorem 17.

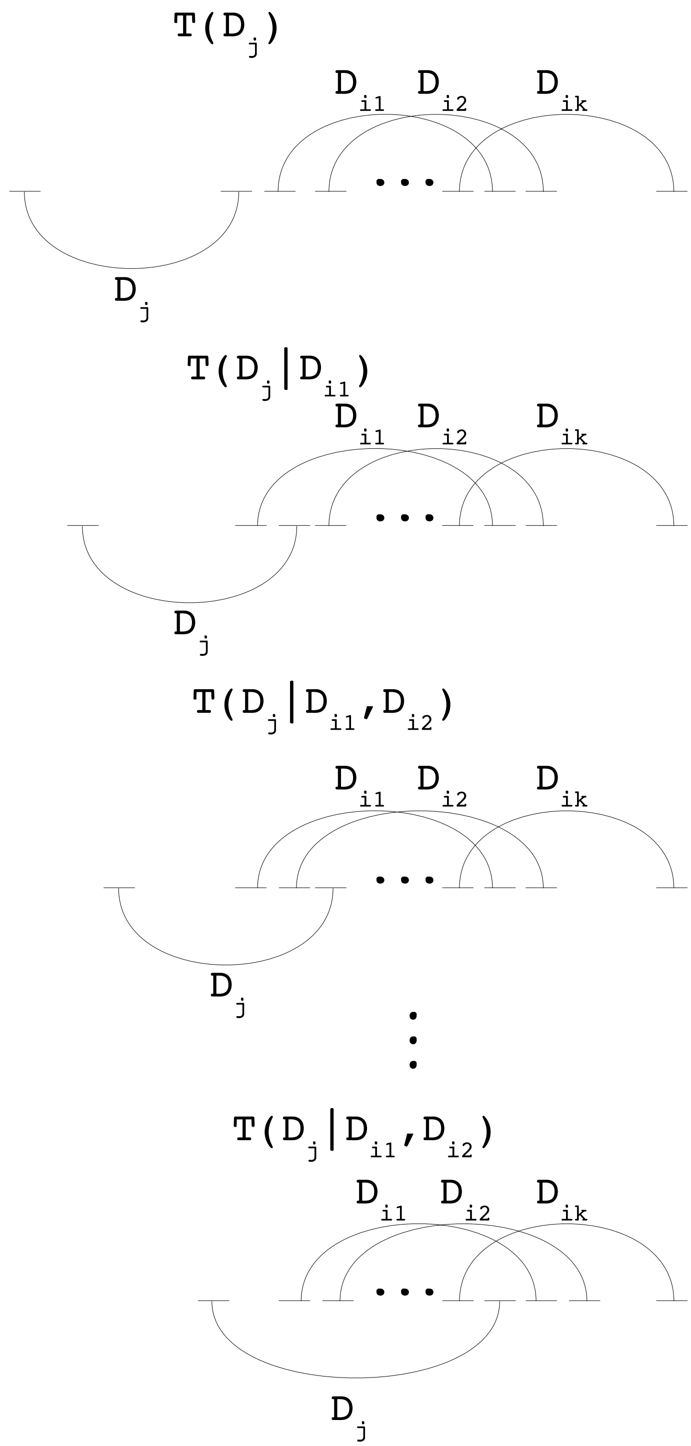

Fig. 22. Illustration of the different conditions of recurrence relation (2).

(1) We associate to each 2-interval $D \in \mathcal{D}$ its least covering interval $\mathsf{Cover}(D)$ and store all these covering intervals in the data structure of Theorem 17. Reporting those 2-intervals in $\mathcal{D}$ that lie entirely to the left of a given 2-interval D is equivalent to reporting those covering intervals that are completely contained in the left preceding interval of D. The time com-

Max $\{<, \between\}$-Comparable 2-Interval Pattern

Input: A finite set $\mathcal{D}$ of n 2-intervals.
Output: The maximum size of a $\{<, \between\}$-comparable pattern in $\mathcal{D}$.

1. Sort the set $\mathcal{D}$ according to their right interval. For the sake of clarity, let us assume that the ordered 2-intervals set is now given by $\mathcal{D} = \{D_1, D_2, \ldots, D_n\}$, *i.e.*, $\mathsf{Right}(D_i) < \mathsf{Right}(D_j)$ implies $i < j$. All ordered subsets considered in the following of the algorithm are to be understood as ordered with respect to that order.
2. For i from 1 to n

2.1. Fill the entry $T(D_i)$.

2.2. For any ordered non-empty set $\{D_{i_1}, D_{i_2}, \ldots, D_{i_q}\} \subseteq \mathcal{D}$ such that $\{D_i\} \cup \{D_{i_1}, D_{i_2}, \ldots, D_{i_q}\}$ is an ordered subset of $\{\between\}$-comparable 2-intervals with $\mathsf{Right}(D_i) < \mathsf{Right}(D_{i_1}) < \ldots < \mathsf{Right}(D_{i_q})$, fill the entry $T(D_i \mid D_{i_1}, D_{i_2}, \ldots, D_{i_q})$ according to the recurrence relation (2).

3. Return the largest entry $T(D_i)$

Fig. 23. Algorithm Max $\{<, \between\}$-Comparable 2-Interval Pattern.

plexity follows from Theorem 17.

(2) We store the left interval of each 2-interval in the data structure of Theorem 17. Reporting is now a two step procedure. First, we find those 2-intervals whose left interval is completely contained in the first query interval. Second, we report those 2-intervals of step one whose right interval is completely contained in the second query interval. Clearly, the first step takes $O(n \log n + k)$ time and the second step runs in $O(k)$ time.

□

Lemma 19 *Let $D_j \in \mathcal{D}$ be such that all entries of the dynamic programming table of the form $T(D_k|*)$ with $\mathsf{Right}(D_k) \leq \mathsf{Right}(D_j)$ have already been computed in a previous run. Then, for any $\{\between\}$-comparable subset $\{D_{i_1}, D_{i_2}, \ldots, D_{i_k}\} \subseteq \mathcal{D}$, $k \geq 1$, satisfying $\mathsf{Right}(D_j) < \mathsf{Right}(D_{i_1}) < \mathsf{Right}(D_{i_2}) < \ldots < \mathsf{Right}(D_{i_k})$, one can compute the entry of the dynamic programming table $T(D_{i_1} \mid D_{i_2}, \ldots D_{i_k})$ according to recurrence relation (2) in $O(n \cdot \mathsf{FCrossing}(\mathcal{D})\,(\log(n) + \mathsf{FCrossing}(\mathcal{D})))$ time.*

PROOF. We first need an injective mapping that associates to any $\{\between\}$-comparable subset $\{D_{i_1}, D_{i_2}, \ldots, D_{i_k}\} \subseteq \mathcal{D}$, $k \geq 1$, satisfying $\mathsf{Right}(D_{i_1}) < \mathsf{Right}(D_{i_2}) < \ldots < \mathsf{Right}(D_{i_k})$, its index in the dynamic programming table T. Let π be a numbering of $\mathcal{D}$ such that the 2-intervals are numbered according to their right interval, *i.e.*, $\mathsf{Right}(D_i) < \mathsf{Right}(D_j)$ implies $\pi(D_i) < \pi(D_j)$ for all $D_i, D_j \in \mathcal{D}$. Let $\mathcal{D}^{\between}$ be the set of ordered subsequences of $\{1, 2, \ldots, n\}$ defined as follows: for any $\{\between\}$-comparable subset $\{D_{i_1}, D_{i_2}, \ldots, D_{i_k}\} \subseteq \mathcal{D}$, $k \geq 1$, satisfying $\mathsf{Right}(D_{i_1}) < \mathsf{Right}(D_{i_2}) < \ldots < \mathsf{Right}(D_{i_k})$, the set $\mathcal{D}^{\between}$

contains the ordered sequence $(\pi(D_{i_1}), \pi(D_{i_2}), \ldots, \pi(D_{i_k}))$. Clearly, one can compare two sequences of $\mathcal{D}^{\between}$, for example according to lexicographic order, in $O(\mathsf{FCrossing}(\mathcal{D}))$ time ; this follows from the fact that sequences of $\mathcal{D}^{\between}$ are of length at most $\mathsf{Depth}(\mathcal{D}) \leq \mathsf{FCrossing}(\mathcal{D}) + 1$. Therefore, using any classical data structure for searching and inserting that guarantees logarithmic time [7], one can insert or search for a given sequence of $\mathcal{D}^{\between}$ in $O(\mathsf{FCrossing}(\mathcal{D})(\log(n) + \mathsf{FCrossing}(\mathcal{D})))$ time. We now turn to the computation of $T(D_{i_1} \mid D_{i_2}, \ldots D_{i_k})$. For each condition (i) of the recurrence relation (2), one has to find those 2-intervals D_j satisfying $D_j \between \{D_{i_r} : 0 \leq r < i\}$ and $D_j < \{D_{i_s} : i \leq s < k+1\}$. According to Lemma 18, this can be done in $O(\log n + p_i)$ where p_i is the number of 2-intervals satisfying condition (i). Then it follows that one can find the maximum value of condition (i) in $O(p_i \cdot \mathsf{FCrossing}(\mathcal{D})(\log(n) + \mathsf{FCrossing}(\mathcal{D})))$ time. Summing up over all conditions (i) and observing that $\sum_{1 \leq i \leq k+1} p_i \leq n$, we obtain an $O(n \cdot \mathsf{FCrossing}(\mathcal{D})(\log(n) + \mathsf{FCrossing}(\mathcal{D}))$ time algorithm for computing the entry of the dynamic programming table $T(D_{i_1} \mid D_{i_2}, \ldots D_{i_k})$. It remains to insert the ordered sequence $(\pi(D_{i_1}), \pi(D_{i_2}), \ldots, \pi(D_{i_k}))$ into the data structure for upcoming queries. According to the above, this can be done in $O(\mathsf{FCrossing}(\mathcal{D})(\log(n) + \mathsf{FCrossing}(\mathcal{D})))$ time. $\square$

PROOF. [of Proposition 16] Correctness of the algorithm follows from recurrence relation (2). What is left is to prove the time complexity. Sorting the set of 2-intervals $\mathcal{D}$ according to their right interval can be done in $O(n \log n)$ time. According to Lemma 19, each entry of the form $T(D_i \mid *)$ can be computed in $O(n \cdot \mathsf{FCrossing}(\mathcal{D})(\log(n) + \mathsf{FCrossing}(\mathcal{D})))$ time. Since the number of distinct entries of the dynamic programming table T is upper-bounded by $n \cdot 2^{\mathsf{FCrossing}(\mathcal{D})}$ (Lemma 15), it follows that the algorithm as a whole runs in $O(n^2 \cdot \mathsf{FCrossing}(\mathcal{D}) \cdot 2^{\mathsf{FCrossing}(\mathcal{D})}(\log(n) + \mathsf{FCrossing}(\mathcal{D})))$ time. $\square$

Corollary 20 *The* 2-IP-Unit-$\{\sqsubset, \between\}$ *problem is fixed-parameter tractable with respect to parameter* $\mathsf{FCrossing}(\mathcal{D})$.

It remains open, however, whether the 2-IP-Unit-$\{\sqsubset, \between\}$ problem is fixed-parameter tractable with respect to parameter $\mathsf{Depth}(\mathcal{D})$ (recall indeed that $\mathsf{FCrossing}(\mathcal{D}) \geq \mathsf{Depth}(\mathcal{D})$).

7 Conclusion

In the context of structured pattern matching, we considered the problem of finding an occurrence of a given structured pattern in a set of 2-intervals and solved three open problems of [29]. We gave an optimal $O(n \log n)$ algorithm for model $R = \{\sqsubset\}$ thereby improving the complexity of the best known algorithm. Also, we described a $O(n^2\sqrt{n})$ time algorithm for model $R = \{\sqsubset, \between\}$ over a disjoint ground set. Finally, we proved that the problem is **NP**-complete for model $R = \{<, \between\}$ over a unit ground set, and in addition to that, we gave a fixed parameter-tractability result based on the crossing structure of the set of 2-intervals. These results almost complete the table of complexity classes for the 2-interval pattern problem proposed by Vialette [29] (see Table 1).

An interesting question would be to answer the last remaining open problem in that area, that is to determine whether there exists a polynomial time algorithm for 2-IP-Dis-$\{<, \between\}$, *i.e.*, finding the largest $\{<, \between\}$-comparable subset of a set of 2-intervals over a disjoint ground set. Note that the 2-IP-Dis-$\{<, \between\}$ problem has an immediate formulation in terms of constrained matchings in general graphs: Given a graph G together with a linear ordering π of the vertices of G, the 2-IP-Dis-$\{<, \between\}$ problem is equivalent to finding a maximum cardinality matching $\mathcal{M}$ in G with the property that for any two distinct edges $\{u, v\}$ and $\{u', v'\}$ of $\mathcal{M}$ neither $\max\{\pi(u), \pi(v)\} < \min\{\pi(u'), \pi(v')\}$ nor $\max\{\pi(u'), \pi(v')\} < \min\{\pi(u), \pi(v)\}$ occur. We note that a related result, determining whether a given $\{<, \between\}$-structured pattern occurs in a general linear graph, has been studied in [18,23]. Gramm [18] gave a polynomial-time algorithm for this problem. Recently, Li and Li [23] proved that this algorithm was incorrect and showed the problem was in fact **NP**-complete. In the light of Table 1, we however conjecture the 2-IP-Dis-$\{<, \between\}$ problem to be polynomial-time solvable.

References

[1] J. Alber, J. Gramm, J. Guo, and R. Niedermeier. Towards optimally solving the longest common subsequence problem for sequences with nested arc annotations in linear time. In *Proceedings of the 13th Annual Symposium on Combinatorial Pattern Matching (CPM 2002)*, volume 2373 of *Lecture Notes in Computer Science*, pages 99–114. Springer-Verlag, 2002.

[2] R. Bar-Yehuda, K. Bendel, A. Freund, and D. Rawitz. Local ratio: A unified framework for approxmation algorithms. *J ACM Comput. Surv.*, 36(4):422–463, 2004.

[3] R. Bar-Yehuda, M.M. Halldorsson, J. Naor, H. Shachnai, and I. Shapira. Scheduling split intervals. In *Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 732–741, 2002.

[4] J.R.S. Blair and B. Peyton. An introduction to chordal graphs and clique trees. *Graph Theory and Sparse Matrix Computation*, 56:1–29, 1993.

[5] G. Blin, G. Fertin, and S. Vialette. New results for the 2-interval pattern problem. In *In Proc. 15th Annual Symposium on Combinatorial Pattern*

Matching (CPM 2004), volume 3109 of *Lecture Notes in Computer Science*, pages 311–322. Springer-Verlag, 2004.

[6] A. Butman, D. Hermelin, M. Lewenstein, and D. Rawitz. Optimization problems in multiple-interval graphs. In *Proc. ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 2007. To appear.

[7] T.H. Cormen, C.E. Leiserson, and R.L. Rivest. *Introduction to algorithms*. MIT Press and McGraw Hill, Cambridge, 1992.

[8] M. Crochemore, D. Hermelin, G. Landau, D. Rawitz, and S. Vialette. Approximating the 2-interval pattern problem. *Theoretical Computer Science (special issue for Alberto Apostolico)*, 2006. To appear.

[9] I. Dagan, M.C. Golumbic, and R.Y. Pinter. Trapezoid graphs and their coloring. *Discrete Applied Mathematics*, 21:35–46, 1988.

[10] M. de Berg, M.van Kreveld, M. Overmars, and O. Schwarzkopf. *Computational Geometry: Algorithms and Applications*. Springer-Verlag, 2nd edition, 2000.

[11] P. Evans. Finding common subsequences with arcs and pseudoknots. In *Proceedings of the 10th Annual Symposium Combinatorial Pattern Matching (CPM 1999)*, volume 1645 of *Lecture Notes in Computer Science*, pages 270–280. Springer-Verlag, 1999.

[12] S. Felsner, R. Müller, and L. Wernisch. Trapezoid graphs and generalizations: Geometry and algorithms. *Discrete Applied Math.*, 74:13–32, 1997.

[13] M.L. Fredman. On computing the length of longest increasing subsequences. *Disrete Mathematics*, 11:29–35, 1975.

[14] D.R. Fulkerson and O.A. Gross. Incidence matrices and interval graphs. *Pacific Journal of Math.*, 15:835–855, 1965.

[15] M.R. Garey and D.S. Johnson. *Computers and Intractability: a guide to the theory of NP-completeness*. W.H. Freeman, San Franciso, 1979.

[16] D. Goldman, S. Istrail, and C.H. Papadimitriou. Algorithmic aspects of protein structure similarity. In *Proceedings of the 40th Annual Symposium of Foundations of Computer Science (FOCS99)*, pages 512–522, 1999.

[17] M.C. Golumbic. *Algorithmic Graph Theory and Perfect Graphs*. Academic Press, New York, 1980.

[18] Jens Gramm. A polynomial-time algorithm for the matching of crossing contact-map patterns. In *Proceedings of the 4th Workshop on Algorithms in Bioinformatics (WABI 2004)*, pages 38–49, 2004.

[19] J. Gramm, J. Guo, and R. Niedermeier. Pattern matching for arc-annotated sequences. In *Proceedings of the the 22nd Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2002)*, volume 2556 of *Lecture Notes in Computer Science*, pages 182–193, 2002.

[20] J.R. Griggs and D.B. West. Extremal values of the interval number of a graph, I. *SIAM J. Alg. Discrete Methods*, 1:1–7, 1979.

[21] T. Jiang, G.-H. Lin, B. Ma, and K. Zhang. The longest common subsequence problem for arc-annotated sequences. In *In Proc. 11th Annual Symposium on Combinatorial Pattern Matching (CPM 2000)*, volume 1848 of *Lecture Notes in Computer Science*, pages 154–165. Springer-Verlag, 2000.

[22] D. Joseph, J. Meidanis, and P. Tiwari. Determining DNA sequence similarity using maximum independent set algorithms for interval graphs. In *Proceedings of the Third Scandinavian Workshop on Algorithm Theory (SWAT 92)*, Lecture Notes in Computer Science, pages 326–337. Springer-Verlag, 1992.

[23] S.C. Li and M. Li. On the complexity of the crossing contact map pattern matching problem. In *Proc. of the 6th Workshop on Algorithms in BioInformatics (WABI)*, LNBI, 2006. To appear.

[24] S. Micali and V.V. Vazirani. An $O(\sqrt{|V|}|E|)$ algorithm for finding maximum matching in general graphs. In *Proceedings of the 21st Annual Symposium on Foundation of Computer Science*, pages 17–27. IEEE, 1980.

[25] R.E. Tarjan and M Yannakakis. Simple linear-time algorithms to test chordality of graphs, test acyclicity of hypergraphs, and selectively reduce acyclic hypergraphs. *SIAM J. Comput.*, 13:566–579, 1984.

[26] W.T. Trotter and F. Harary. On double and multiple interval graphs. *J. Graph Theory*, 3:205–211, 1979.

[27] V.V. Vazirani. A theory of alternating paths and blossoms for proving correctness of the $O(\sqrt{|V|}|E|)$ maximum matching algorithm. *Combinatorica*, 14(1):71–109, 1994.

[28] S. Vialette. Pattern matching over 2-intervals sets. In *In Proc. 13th Annual Symposium Combinatorial Pattern Matching (CPM 2002)*, volume 2373 of *Lecture Notes in Computer Science*, pages 53–63. Springer-Verlag, 2002.

[29] S. Vialette. On the computational complexity of 2-interval pattern matching. *Theoretical Computer Science*, 312(2-3):223–249, 2004.

[30] J. Viksna and D.Gilbert. Pattern matching and pattern discovery algorithms for protein topologies. In O. Gascuel and B.M.E. Moret, editors, *Proceedings of the 1st Workshop on Algorithms in Bioinformatics (WABI 2001)*, volume 2149 of *Lecture Notes in Computer Science*, pages 98–111. Springer, 2001.

[31] D.B. West and D.B. Shmoys. Recognizing graphs with fixed interval number is NP-complete. *Discrete Applied Mathematics*, 8:295–305, 1984.