Characterizing minimal interval completions: Towards better understanding of profile and pathwidth*

Pinar Heggernes† Karol Suchan‡§ Ioan Todinca‡ Yngve Villanger†

Abstract

Minimal interval completions of graphs are central in understanding two important and widely studied graph parameters: profile and pathwidth. Such understanding seems necessary to be able to attack the problem of computing these parameters. An interval completion of a given graph is an interval supergraph of it on the same vertex set, obtained by adding edges. If no subset of the added edges can be removed without destroying the interval property, we call it a minimal interval completion. In this paper, we give the first characterization of minimal interval completions. We present a polynomial time algorithm for deciding whether a given interval completion of an arbitrary graph is minimal, thereby resolving the complexity of this problem.

1 Introduction

Interval graphs have a long list of applications in areas like biology, chemistry, and archeology, and many NP-complete graph problems are solvable in polynomial time on interval graphs [11]. Specifically, the problem of adding edges to a given input graph to obtain an interval graph, called an *interval completion* of the input graph, arises in Physical Mapping of DNA [10, 19], Orthogonal Packing [6], and Sparse Matrix Computations [9]. For several applications, it is desirable to embed a given graph into an interval graph by adding as few edges as possible. Such an embedding is called a *minimum interval completion*, and the number of edges it contains is called the *profile* of the input graph. Another well known variant is to find an interval completion with the smallest possible maximum clique, which corresponds to the widely used graph parameter *pathwidth*. Many NP-complete graph problems are solvable on graphs whose pathwidth is bounded by a constant [1], thus it is an important task to compute the pathwidth of an input graph. This parameter has many applications, as the pathwidth of a graph is also known as its vertex separation number, and in addition it corresponds to its node search number (see [1] for a survey).

Both profile and pathwidth are well known and well studied graph parameters, and naturally both are NP-hard to compute [8, 12]. There has been extensive work on computing these parameters for restricted graph classes [5], but our insight on how to handle arbitrary graphs

*This work is supported by the Research Council of Norway through grant 166429/V30.

†Department of Informatics, University of Bergen, N-5020 Bergen, Norway. Emails: pinar@ii.uib.no, yngvev@ii.uib.no.

‡LIFO, Université d'Orleans, PB 6759, F-45067 Orleans Cedex 2, France. Emails: todinca@lifo.univ-orleans.fr, suchan@lifo.univ-orleans.fr.

§Faculty of Applied Mathematics, AGH - University of Science and Technology, Krakow, Poland.

is limited. One good news is that for both problems, the solutions can be found within the set of minimal interval completions. This is our main motivation to study minimal interval completions. A minimal interval completion of an arbitrary graph can be computed in polynomial time [14, 22], but still the knowledge about minimal interval completions is limited. The algorithms of [14] and [22] cannot generate every minimal interval completion, and until now it was not known how to decide whether a given interval completion is minimal. In this paper, we solve exactly this problem. We characterize minimal interval completions, which enables us to answer whether or not a given interval completion is minimal, in polynomial time.[1] The techniques presented here are more general and completely different from those that lead to the polynomial time algorithms of [14] and [22].

Two other important and widely studied graph parameters are minimum fill and treewidth, and these are defined analogously to profile and pathwidth, by simply exchanging interval graphs with chordal graphs. As a comparison, minimal chordal completions were studied and a polynomial time algorithm for computing them was given already in 1976 [21], even before it was proved that minimum fill is NP-hard to compute [24]. Since then many results have been added about minimal chordal completions [13] (in particular at recent years' SODA conferences [15, 17]), which are central in understanding and trying to solve minimum fill and treewidth problems. Minimal chordal completions have have several quite different characterizations, and some of these have proved useful in trying to compute minimum fill and treewidth [4, 16], either by approximation algorithms [18] or by exact (fast) exponential time algorithms [7].

Following the history of chordal completions, our hope is that understanding and characterizing minimal interval completions will eventually lead to improved exact or approximation algorithms for computing profile and pathwidth. Our results in this paper constitute a first step in building tools that might be useful in this direction. Our techniques are general, and they can be applied to other problems as well. For example, some of them have been used for computing the pathwidth of circular-arc graphs in polynomial time [23].

In addition to characterizing minimal interval completions, we present the first polynomial time algorithm for making any given interval completion minimal by removing edges. Thus another impact of our results is that any output graph from a heuristic algorithm for computing profile or pathwidth can be enhanced by using our algorithm, which will produce a minimal interval completion that is a subgraph of the given initial interval completion. For practical purposes, such approaches have proved useful in connection with treewidth [2] and minimum fill [20], and can now be applied to pathwidth and profile by our results.

2 Definitions and terminology

We work with simple and undirected graphs $G = (V, E)$, with vertex set $V(G) = V$ and edge set $E(G) = E$, and we let $n = |V|$, $m = |E|$. For a given vertex set $X \subset V$, $G[X]$ denotes the subgraph of G induced by the vertices in X. For simplicity, we will use $G - x$ instead of $G[V \setminus \{x\}]$, and $G - X$ instead of $G[V \setminus X]$. Similarly, when we remove a single edge xy or a set of edges D from G, we will use $G - xy$ and $G - D$ instead of the correct formal notation. A vertex set X is a *clique* if $G[X]$ is a complete graph, and a *maximal clique* if no superset

[1] A polynomial time algorithm for computing minimal interval completions does not necessarily imply that this question can be answered in polynomial time.

of X is a clique. The set of all maximal cliques of G is denoted by $\mathcal{K}(G)$.

The set of neighbors of a vertex x is denoted by $N_G(x) = \{y \mid xy \in E\}$, and the *closed neighborhood* is $N_G[x] = N_G(x) \cup \{x\}$. For a vertex set X, similarly, $N_G(X) = \{y \notin X \mid xy \in E \text{ and } x \in X\}$ and $N_G[X] = N_G(X) \cup X$.

Definition 2.1 ([1]) *A* path-decomposition *of an arbitrary graph $G = (V, E)$ is a sequence $P = (X_1, X_2, \ldots, X_r)$ of subsets of V, called* bags*, such that the following three conditions are satisfied.*

1. *Each vertex x appears in some bag.*
2. *For every edge $xy \in E$ there is a bag containing both x and y.*
3. *For every vertex $x \in V$, the bags containing x appear consecutively in P.*

Such a path decomposition can be constructed for any graph G, for example by taking a unique bag containing $V(G)$. The width of a decomposition is the maximum size of a bag, minus one, and the *pathwidth* of a graph is the minimum width over all possible path decomposition.

For interval graphs, special path decompositions exist such that each bag is a maximal clique, and the largest maximal clique gives the pathwidth. A graph is an *interval graph* if intervals can be associated to its vertices such that two vertices are adjacent if and only if their corresponding intervals overlap. Let us define more formally the special kind of path decompositions mentioned:

Definition 2.2 *A* clique-path *of a graph G is a permutation $P = (K_1, \ldots, K_p)$ of the maximal cliques of G, such that the maximal cliques containing x appear consecutively in P, for every vertex x of G.*

Theorem 2.3 ([11]) *A graph G is an interval graph if and only if if has a clique-path.*

In a given clique-path $P = (K_1, \ldots, K_p)$, the maximal cliques K_1 and K_p are called *leaf cliques* or *end cliques*. An interval graph has at most n maximal cliques.

Since the vertices of every maximal clique must appear together in some bag in every path decomposition, a clique-path is an optimal path decomposition for an interval graph, with respect to pathwidth. Interval graphs can be recognized, and their clique-paths can be computed, in linear time [3]. If a clique-path of an interval graph is given, then its interval representation can be easily obtained by assigning to each vertex x the interval consisting of the maximal cliques that contain x.

To any path-decomposition P of G, we can naturally associate an interval supergraph of G. Let $\texttt{PathFill}(G, P)$ be the graph obtained by adding edges to G so that each bag of P becomes a clique. It is straight forward to verify that $\texttt{PathFill}(G, P)$ is an interval supergraph of G for every path-decomposition P. In addition, we can obtain a clique-path of $\texttt{PathFill}(G, P)$ by simply removing bags of P that are not maximal cliques of $\texttt{PathFill}(G, P)$ or that are duplicates of other bags.

An interval supergraph $H = (V, E \cup F)$ of a given graph $G = (V, E)$, with $E \cap F = \emptyset$, is called an *interval completion*. The set F is called the set of *fill edges* of H. If there is no proper subset $F' \subset F$ such that $(V, E \cup F')$ is an interval graph, then H is called a *minimal interval completion* of G. With a weaker constraint, we say that H is a *quasi-minimal interval completion* of G if no single fill edge can be removed from H without destroying interval graph property. Simple examples exist to show that quasi-minimal interval completions are not necessarily minimal.

Finally, we would like to mention that clique-paths are useful also in connection with vertex separators. A vertex set $S \subset V$ is a *separator* if $G - S$ is disconnected. The set of connected components of $G - S$ is denoted by $C(G - S)$. Given two vertices u and v, S is a u, v-*separator* if u and v belong to different connected components of $G - S$. A u, v-separator S is *minimal* if no proper subset of S is a u, v-separator. In general, S is a *minimal separator* of G if there exist two vertices u and v in G such that S is a minimal u, v-separator.

Lemma 2.4 (see e.g. [11]) *Let H be an interval graph and let $P = (K_1, \ldots, K_p)$ be any clique path of H. A set of vertices S is a minimal separator of H if and only if S is the intersection of two maximal cliques of H that are consecutive in P.*

Assume that we are given an interval completion H of an arbitrary graph G. We want to find out whether H is a minimal interval completion. First we can start by trying to remove every single fill edge and test, in linear time, whether the remaining graph is an interval graph. After a number of steps (which is at most quadratic in the number of edges of H) we reach a quasi-minimal interval completion. Thus from now on, we assume that we are given a quasi-minimal interval completion H of G, and we want to decide whether it is minimal. If it is not minimal, we know that there is one that is minimal which is a strict subgraph of H, and before we finally find a minimal one, we might explore several strict interval subgraphs of H that are not minimal.

Let us give different names to these different interval completions of G. Let H_2 be the given quasi-minimal interval completion of G. If it is not minimal, let H_0 be a minimal interval completion of G that is a subgraph of H_2. Since we are only given G and H_2, and we do not know H_0, we will probably discover several intermediate graphs H_1, where H_1 is an interval completion of G that is a strict subgraph of H_2. Hence we have the following relations between these graphs: $E(G) \subset E(H_0) \subseteq E(H_1) \subset E(H_2)$. The first subset relation is proper because we can always check before start whether or not G is already an interval graph, in linear time. Even though we do not know H_0, we know that H_2 is a non-minimal and quasi-minimal interval completion of it. In the next section, we give useful properties about two interval graphs that have this relationship.

3 Folding interval graphs

Let H_0 be any interval graph (which is the unknown minimal interval completion of some given graph G in our case), and let H_2 be a non-minimal interval completion of H_0 (we think of it as being the given non-minimal but quasi-minimal interval completion of G). In this section we give an algorithm that computes any such completion H_2, given H_0. Of course, in our problem, we are given H_2 and not H_0. However, this algorithm provides the necessary understanding that will enable us to do the opposite operation; namely, computing H_0, given G and H_2.

Every permutation Q of the maximal cliques of H_0 defines an interval completion H_2 of H_0, as described by Algorithm `FillFolding` in Figure 1. Note that $Q(i)$ denotes the maximal clique in the i^{th} position of Q.

Definition 3.1 *Let H be any interval graph, let Q be any permutation of the set of maximal cliques of H. We say that (H, Q) is a* folding *of H by Q.*

Algorithm `FillFolding`
Input: H_0 and Q;
Output: An interval completion H_2 of H_0;

$P_2 = Q$;
for each vertex x of H_0 **do**
  $s = \min\{i \mid x \in Q(i)\}$;
  $t = \max\{i \mid x \in Q(i)\}$;
  for $j = s + 1$ to $t - 1$ **do**
    $P_2(j) = P_2(j) \cup \{x\}$;
end-for
$H_2 =$ `PathFill`(H_0, P_2);

Figure 1: The `FillFolding` algorithm.

Lemma 3.2 *Given a folding (H_0, Q) of H_0, the graph $H_2 =$ `FillFolding`(H_0, Q) is an interval completion of H_0.*

Proof. Observe that after the **for** loops, P_2 is a path-decomposition of H_0, since every edge is contained in some bag, and for every vertex the bags containing it induce a subpath of P_2. Hence, since $H_2 =$ `PathFill`(H_0, P_2), it is an interval completion of H_0. ■

The graph defined by a folding is not necessarily a quasi-minimal interval completion of H_0. Nevertheless, we prove in Theorem 3.8 that every quasi-minimal interval completion of H_0 is defined by some folding.

The maximal cliques of H_0 play a crucial role in the analysis of the relationship between H_0 and H_2; in particular the ones that are contained in a unique maximal clique of H_2. The following definition is general, however we will apply it later on interval input graphs and their non-minimal interval completions.

Definition 3.3 *Let H be an interval completion of an arbitrary graph G. A maximal clique of G which is a subset of exactly one maximal clique of H is called a* core clique *of G with respect to H.*

The following is a straight forward consequence of Lemma 2.4.

Observation 3.4 *Let H be an interval completion of an arbitrary graph G. Let K be a maximal clique of G which is not a core clique with respect to H. Then there is a minimal separator S of H with $K \subseteq S$.*

Before we prove that any quasi-minimal interval completion H_2 of H_0 can be obtained via Algorithm `FillFolding`, we need to introduce a few lemmas.

Lemma 3.5 *Let H_2 be a (non-minimal) interval completion of an interval graph H_0. Let xy be a fill-edge such that only one maximal clique of H_2 contains both x and y. Then H_2 is not a quasi-minimal interval completion of H_0.*

Proof. Let $P_2 = (\Omega_1, \Omega_2, ..., \Omega_k)$ be a clique-path of H_2 and suppose that Ω_i is the only clique containing both x and y. Observe that $H_1 = H_2 - xy$ is also an interval graph. Indeed, replacing Ω_i in P_2 with Ω_x, Ω_y, where $\Omega_x = \Omega_i \setminus \{y\}$ and $\Omega_y = \Omega_i \setminus \{x\}$ yields a path-decomposition of H_1, from which we can obtain a clique-path of H_1 by simply removing redundant cliques. ∎

Lemma 3.6 *Let H_2 be a quasi-minimal interval completion of an interval graph H_0. Let Ω be a maximal clique of H_2. Then there is a core clique K of H_0 contained in Ω.*

Proof. Let $P_2 = (\Omega_1, \Omega_2, ..., \Omega_k)$ be a clique-path of H_2, and let $\Omega = \Omega_i$ for some $i \in \{1, ..., k\}$. Since Ω is a maximal clique, it contains a vertex x such that $x \notin \Omega_{i-1}$ and a vertex y such that $y \notin \Omega_{i+1}$. If $x = y$ then Ω is the unique maximal clique of H_2 containing x. In this case, it is sufficient to take a maximal clique K of H_0 containing x, and K is a core clique with respect to H_2. Consider the case $x \neq y$. If $xy \notin E(H_0)$, then by Lemma 3.5, H_2 is not quasi-minimal, and we have a contradiction to the premise of the lemma. Thus xy is an edge of H_0 contained only in Ω. Any maximal clique K of H_0 with $x, y \in K$ is contained only in Ω, and is thus a core clique. ∎

Lemma 3.7 *Let H_2 be a quasi-minimal interval completion of an interval graph H_0, and let P_2 be a clique-path of H_2. Let x be a vertex, and let P_x be the subpath of P_2 induced by the maximal cliques containing x. Then, each maximal clique of H_2 that is a leaf in P_x, contains a core clique that contains x.*

Proof. If x is simplicial in H_2 then P_x consists of a single maximal clique Ω. Thus any maximal clique K of H_0 with $x \in K$ is a core clique contained in Ω. Assume that P_x is not a singleton, and let Ω be a leaf clique of P_x. Let y be a vertex in Ω that is not contained in the clique next to Ω in P_x. Observe that $xy \in E(H_0)$, otherwise Lemma 3.5 would yield a contradiction. So we have an edge of H_0 contained only in Ω. Thus, there is a maximal clique of H_0 contained only in Ω. ∎

We are now ready to give the main result of this section. For the proof of the theorem, we need the following notation. Given any path-decomposition P and a vertex x of an arbitrary graph G, we denote by $L(x, P)$ and $R(x, P)$, respectively, the leftmost and rightmost bags of P containing x.

Theorem 3.8 *Let H_2 be a quasi-minimal interval completion of an interval graph H_0. Then there exists a folding (H_0, Q) of H_0 such that $H_2 = \mathtt{FillFolding}(H_0, Q)$.*

Proof. Let $\mathcal{K}_0 = \{1 \leq i \leq p \mid K_i\}$ and $\mathcal{K}_2 = \{1 \leq i \leq k \mid \Omega_i\}$ denote the sets of maximal cliques of H_0 and H_2, respectively. Let $P_2 = (\Omega_1, \ldots, \Omega_k)$ be a clique-path of H_2. It defines a linear order on the set $\mathcal{K}_2$. In a natural way, P_2 defines a linear pre-order on $\mathcal{K}_0$ by

$$K_a \leq K_b \text{ if } K_a \subseteq \Omega_i \text{ and } K_b \subseteq \Omega_j, \text{ where } i \leq j \leq k \text{ and } a, b \leq p.$$

Transform it into a linear order (sequence) Q as follows. First, let Q be a linear ordering of the core cliques according to the relative ordering in P_2 of the unique cliques that contain them (i.e., by using any linear extension of the preorder above). Notice that for any maximal clique K of H_0 which is not a core clique, by Lemma 3.7, there is a core-clique K' in Q, such that for every vertex $x \in K$, there are two core-cliques $K_l(x), K_r(x)$ containing x, such that

$K_l(x) \le K'$ and $K_r(x) > K'$ in Q. Indeed, by Observation 3.4 and Lemma 2.4, there are two maximal cliques Ω', Ω'' consecutive on P_2, such that $K \subseteq S = \Omega' \cap \Omega''$. Then for every vertex $x \in K$, by Lemma 3.7, there is some core clique $K_l(x)$ ($K_r(x)$) contained in a bag to the left (right, respectively) of S in P_2. Insert K as the immediate successor of K' in Q. This operation does not change the values of $L(x, Q)$ and $R(x, Q)$.

Let us define $H_2' = \texttt{FillFolding}(H_0, Q)$, and prove that $H_2' = H_2$. Notice that $xy \in E(H_2)$ if and only if the intervals defined by endpoints $L(x, P_2), R(x, P_2)$ and $L(y, P_2), R(y, P_2)$ intersect in P_2. Clearly, the same holds for H_2' and any clique-path P_2' of H_2'. By construction, $E(H_0) \subseteq E(H_2') \subseteq E(H_2)$. Let us prove that all fill-edges of H_2 are also present in H_2'. Suppose it is not the case. Then there is a fill-edge $xy \in E(H_2) \setminus E(H_2')$. Thus the intervals corresponding to x,y in P_2' are disjoint, say $R(x, P_2') < L(y, P_2')$, whereas in P_2 they intersect. It implies that $R(x, P_2) = L(y, P_2)$ and, by Lemma 3.5, H_2 in not quasi-minimal, which is a contradiction. ∎

Given only the arbitrary graph G and a quasi-minimal interval completion H_2, we know by Theorem 3.8 that H_2 is defined by a folding of H_0, a minimal interval completion of G. In general is difficult to find directly the graph H_0. Instead, we can analyze a sequence of interval completions that are between H_0 and H_2, where passing from one step to another needs a folding of a much more constrained nature than the general one – that we call *reduced folding*.

Definition 3.9 *Let (H_0, Q_0) be a folding. A clique $K \in Q$ is called a* pivot *in (H_0, Q_0) if there is a clique-path P_0 of H_0 where both cliques just next to K (one to the left, the other to the right) in P_0 are on the same side of K in Q_0.*

Definition 3.10 *A folding (H, Q) is said to be* reduced *if every pivot contains a simplicial vertex of H.*

The proofs of the next Theorem and of Lemma 3.12 are moved to Appendix A.

Theorem 3.11 *Let H_2 be a quasi-minimal interval completion of H_0 defined by a folding (H_0, Q_0). Then there is an interval graph H_1 such that $H_0 \subseteq H_1 \subset H_2$, and $H_2 = \texttt{FillFolding}(H_1, Q_1)$ for some reduced folding (H_1, Q_1) of H_1.*

Lemma 3.12 *Let $H_2 = \texttt{FillFolding}(H_1, Q_1)$ for a reduced folding (H_1, Q_1). Then every pivot of (H_1, Q_1) contains a simplicial vertex in H_2, thus the pivot is contained in exactly one maximal clique of H_2.*

4 Unfolding

Let H_2 be a quasi-minimal interval completion with a clique-path P_2, obtained by the Algorithm `FillFolding` on (H_0, Q). For the analysis presented in this section, we need to fix a clique-path P_0 of H_0. The reason for this is that with the general definition of a pivot, a pivot K may have different neighbors in distinct clique-paths of H_0. So it may happen that the neighbors of K in P_0 appear at the same side of K in the permutation Q, whereas the neighbors of K in P_0', another clique-path of H_0, appear at different sides of K in Q. For the ease of argument, from now on we should think of a folding as a triple.

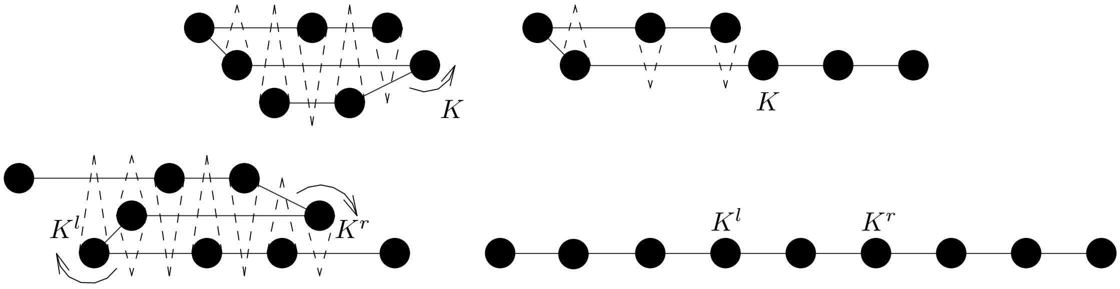

Figure 2: Each circle represents a maximal clique in the input interval graph H_0, and each line provides the information that the two maximal cliques are consecutive in the clique path P_0 of H_0. The foldings are defined by the order of the maximal cliques from left to right. Arrows indicate in which direction a sub path is rotated around the pivot when we open a fold. The upper part demonstrates how to unfold when one of the extremal maximal cliques is a pivot, like K in this case. The lower part shows how to unfold when K^r is the rightmost pivot in Q and K^l is the leftmost pivot in Q that appears after K^r in P_0.

Definition 4.1 *Let H be an interval graph, P be a clique-path of H and Q be a permutation of its maximal cliques. The triple (H, Q, P) is a* folding.

The definition of a pivot becomes more constrained.

Definition 4.2 *Let (H_0, Q, P_0) be a folding. A clique $K \in Q$ is called a* pivot *in (H_0, Q, P_0) if both cliques just next to K in P_0 are on the same side of K in Q_0.*

Definition 4.3 *Let H_0 be an interval graph with a clique-path P_0. Let (H_0, Q, P_0) be a reduced folding. If Q contains just one pivot then it is called a* 1-folding. *If Q contains exactly 2 pivots, none of which is at an end of Q, then it is called a* 2-folding.

We show in this section that if the quasi-minimal completion H_2 of G is not minimal, there is an interval graph H_1 containing G and strictly contained in H_2 such that H_2 is obtained by a reduced 1-folding or a reduced 2-folding of H_1. In the next section we give a polynomial algorithm constructing H_1.

Let us sketch the main idea before getting into the technical proofs. The graph H_2 is equal to `FillFolding`(H_0, Q, P_0) for some smaller interval completion H_0 of G and some folding of H_0. Informally, we will slightly *unfold* (H_0, Q, P_0):

Definition 4.4 *A folding (H_0, Q', P_0) is an* unfolding *of (H_0, Q, P_0) if the set of pivots of (H_0, Q', P_0) is strictly contained into the set of pivots of (H_0, Q, P_0) and, moreover, the graph `FillFolding`(H_O, Q') is a (not necessarily strict) subgraph of `FillFolding`(H_0, Q).*

We will construct an unfolding (H_0, Q', P_0), having one or two pivots less than (H_0, Q, P_0). Then H_2 is obtained by a 1 or 2-folding of $H_1 =$ `FillFolding`(H_0, Q'). The proof of the following Theorem, depicted in Figure 2, is moved to Appendix B.

Theorem 4.5 *Let H_2 be a quasi-minimal, but not minimal interval completion of an arbitrary graph G. Then there exits a reduced folding (H_1, Q_1, P_1), with one (1-folding) or two (2-folding) pivots, with $E(G) \subseteq E(H_1) \subset E(H_2)$ and such that $H_2 =$ `FillFolding`(H_1, Q_1).*

Observation 4.6 *Let (H_0, Q_0, P_0) be a reduced 1-folding and let $H_2 = \texttt{FillFolding}(H_0, Q_0)$. Then its pivot is a maximal clique in H_2. Moreover, there is a clique path of H_2 such that this pivot corresponds to a leaf.*

Proof. Since (H_0, Q_0, P_0) is a 1-folding, there is only one pivot K in Q_0 and clearly K is at one end of Q_0. By Algorithm `FillFolding`, no vertex is added to K and K is at the end of the path decomposition P_2 produced by the algorithm. Because the folding is reduced, there is a simplicial vertex x of H_0 such that the only bag of P_0 (and therefore of P_2) containing x is K. Consequently K is a maximal clique of H_2, corresponding to a leaf of P_2. ■

The next results state that, if H_2 is a quasi-minimal interval completion of G and comes from a one or two-folding of some $H_1 \subset H$, there is an edge uv in $E(H_2) \setminus E(H_1)$ with special properties. In the next section, we shall ensure that the unfolding algorithm removes this edge. The proof of Theorem 4.7 is moved in the Appendix C.

Theorem 4.7 *Let $H_1 = (V, E_1)$, $H_2 = (V, E_2)$ be interval graphs and let (H_1, Q, P_1) be a 1 or 2-folding such that $H_2 = \texttt{FillFolding}(H_1, Q)$. If H_2 is a quasi-minimal but not minimal interval completion of H_1, then there is a fill edge uv, such that one of the pivots K is a u, v-separator in H_1. Moreover, in the clique path P_1, the vertices u and v appear on different sides of K.*

5 Extracting minimal interval completions: the algorithm

Let H_2 be a quasi-minimal interval completion of G and let H_0 be a minimal interval completion of G contained in H_2. Theorem 3.8 shows that there exists a one folding (H_0, Q_0, P_0) that defines H_2, and by Theorem 4.5 there exists a reduced folding (H_1, Q_1, P_1) with one or two pivots that defines H_2. By Lemma 3.12 each pivot of (H_1, Q_1, P_1) is contained in exactly one maximal clique of H_2.

Let us now assume that (H_1, Q_1, P_1) is such that any unfolding defines a graph with fewer edges than H_2. We will focus of finding an unfolding such that some fill edge uv is removed. The edge uv is chosen such that one of the pivots of (H_1, Q_1, P_1) is a u, v-separator in H_1 (see Theorem 4.7).

We will consider the cases of 1-folding and 2-folding separately. Let us first discuss the 1-folding case. Remember from Observation 4.6 that a maximal clique K of H_2 is a pivot in P_1 if (H_1, Q_1, P_1) is an 1-folding defining H_2.

Lemma 5.1 *Let $G = (V, E)$ be an arbitrary graph, and let H_1 and H_2 be two interval completions of G, such that $E(H_1) \subset E(H_2)$, H_2 is a quasi-minimal interval completion of H_1, and (H_1, Q_1, P_1) is a 1-folding that defines H_2. Then $H_1' = \texttt{OneUnfolding}(G, H_2)$ is an interval completion of G satisfying $E(H_1') \subset E(H_2)$.*

Proof. Algorithm `OneUnfolding` always outputs an interval graph H_1' such that $E(G) \subseteq E(H_1') \subseteq E(H_2)$. We have to prove that, under the conditions of the lemma, the graph H_1' is a strict subgraph of H_2.

From Observation 4.6 it follows that the pivot in (H_1, Q_1, P_1) is a maximal clique in H_2; let this maximal clique be Ω. Moreover, Ω is at one end, say the left one, of a clique path P_2 of H_2. Let uv be one of the edges in $E(H_2) \setminus E(H_1)$, such that Ω is a u, v-separator in H_1 (see Theorem 4.7). In particular, both $u, v \notin \Omega$.

Algorithm `OneUnfolding`
Input: A graph $G = (V, E)$, and an interval completion H_2 of G
Output: An interval completion H'_1 of G such that
$E(H'_1) \subset E(H_2)$ if H_2 is defined by a 1-folding of some H_1
$H'_1 = H_2$ if no H_1 exists.

for each pair (Ω, u) where $\Omega \in \mathcal{K}(H_2)$ and $u \in V \setminus \Omega$
  Let C_u be connected comp of $G[V \setminus \Omega]$ containing u
  $H'_1 = (V, E(H_2[N_G[C_u]]) \cup E(H_2[V \setminus C_u]))$
  if H'_1 is an interval graph and $E(H'_1) \subset E(H_2)$ then
    return H'_1
return H_2

Figure 3: Opening one pivot.

Like in Algorithm `OneUnfolding`, let C_u be the connected component of $G[V \setminus \Omega]$ containing the vertex u and consider the graph $H'_1 = (V, E(H_2[N_G[C_u]]) \cup E(H_2[V \setminus C_u]))$. We have to argue that $E(H'_1) \subset E(H_2)$ and that H'_1 is an interval graph. First $E(H'_1) \subseteq E(H_2)$ since only edges in H_2 are used to create H'_1. Since Ω is a u, v-separator in H_1, we have $v \notin N_G[C_u]$. Clearly $u \notin V \setminus C_u$, so uv is not an edge of H'_1.

Let us construct a path decomposition of H'_1 such that each bag induces a clique. This shows that H'_1 it is indeed an interval graph. Take P_2 and glue it in Ω with P'_2, an reversed copy of P_2. Remove vertices from bags in order to have only the vertices of $V \setminus C_u$ in the P_2 part and the vertices of $N_G[C_u]$ in the P'_2 part. Let P'_1 denote the result. Clearly, it is a path decomposition of H'_1. By removing redundant bags we obtain a clique-path of H'_1. ∎

Algorithm `TwoUnfolding`
Input: A graph $G = (V, E)$, and an interval completion H_2 of G
Output: An interval completion H'_1 of G such that
$E(H'_1) \subset E(H_2)$ if H_2 is defined by a 2-folding of some $H_1 \subset H_2$
$H'_1 = H_2$ if no H_1 exists.

for each tuple $(\Omega^l, \Omega^r, S^l, S^r, C^l, C^r, u, v)$
  # Ω^l, Ω^r are maximal cliques of H_2
  # S^l, S^r are minimal separators of H_2, contained in Ω^l and resp. Ω^r.
  # C^l (C^r) is a component of $H_2 - S^l$ (resp. $H_2 - S^r$)
  # u, v are vertices, $u \notin \Omega^r$
  construct W^l using Equation 4 of Appendix D
  construct W^r using Equation 5 of Appendix D
  $H'_1 = (V, E(H_2[N_G[W^l] \cup S^l]) \cup E(H_2 - (W^l \cup W^r)) \cup E(H_2[N_G[W^r] \cup S^r]))$
  if H'_1 is an interval graph and $E(H'_1) \subset E(H_2)$ then
    return H'_1
return H_2

Figure 4: Opening two pivots.

The case when H_2 is defined by a 2-folding of H_1 is fully described in Appendix D, due to space restriction. A sketch of the algorithm is given in Figure 4, and its correctness is stated below.

Lemma 5.2 *Let $G = (V, E)$ be an arbitrary graph, and let H_1 and H_2 be two interval completions of G, such that $E(H_1) \subset E(H_2)$, H_2 is a quasi-minimal interval completion of H_1, and (H_1, Q_1, P_1) is a 2-folding that defines H_2. Then $H_1' =$ `TwoUnfolding`(G, H_2) is an interval completion of G satisfying $E(H_1') \subset E(H_2)$.*

Algorithm `ExtractMinimalIntervalCompletion`
Input: A graph $G = (V, E)$, and an interval completion H_2 of G.
Output: A minimal interval completion H_1 of G, with $E(H_1) \subseteq E(H_2)$.

$H_1 = H_2$
$H_0 = G$
while $(H_0 \neq H_1)$
  $H_0 = H_1$
  for each edge uv in $E(H_1) \setminus E(G)$
    if $H_1 - uv$ is an interval graph then
      $H_1 = H_1 - uv$
  $H_1 =$ `OneUnfolding`(G, H_1)
  $H_1 =$ `TwoUnfolding`(G, H_1)
return H_1

Figure 5: Extracting a minimal interval completion.

Lemmas 5.1 and 5.2 imply the main result of this paper. Algorithm `ExtractMinimal-IntervalCompletion` is given in Figure 5.

Theorem 5.3 *There exists a polynomial time algorithm that, given an arbitrary graph G and an interval completion H_2 of G, computes a minimal interval completion H_1 of G, such that $E(H_1) \subseteq E(H_2)$.*

Let us point out that, by using as initial completion the complete graph, the algorithm `ExtractingMinimalIntervalCompletion` can obtain any of the minimal interval completions of G.

References

[1] H. L. Bodlaender. A partial k-arboretum of graphs with bounded treewidth. *Theor. Comput. Sci.*, 209(1-2):1–45, 1998.

[2] H. L. Bodlaender and A. M. C. A. Koster. Safe separators for treewidth. Technical Report UU-CS-2003-027, Institute of information and computing sciences, Utrecht University, Netherlands, 2003.

[3] K. Booth and G. Leuker. Testing for the consecutive ones property, interval graphs, and graph planarity using PQ-tree algorithms. *J. Comput. Syst. Sci.* 13:335–379, 1976

[4] V. Bouchitté and I. Todinca. Treewidth and minimum fill-in: Grouping the minimal separators. *SIAM J. Comput.*, 31:212–232, 2001.

[5] J. Díaz, J. Petit, and M. J. Serna. A survey of graph layout problems. *ACM Computing Surveys*, 34:313–356, 2002.

[6] S. P. Fekete and J. Schepers. A combinatorial characterization of higher-dimensional orthogonal packing. *Mathematics of Operations Research*, 29:353–368, 2004.

[7] F. V. Fomin, D. Kratsch, and I. Todinca. Exact (exponential) algorithms for treewidth and minimum fill-in. In *ICALP 2004*, volume 3142 of *Lecture Notes in Computer Science*, pages 568–580. Springer, 2004.

[8] M. R. Garey and D. S. Johnson. *Computer and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman, 1979.

[9] J. A. George and J. W. H. Liu. *Computer Solution of Large Sparse Positive Definite Systems.* Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1981.

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

[11] M. C. Golumbic. *Algorithmic Graph Theory and Perfect Graphs.* Academic Press, San Diego, 1980.

[12] J. Gustedt. On the pathwidth of chordal graphs. *Discrete Appl. Math.*, 45(3):233–248, 1993.

[13] P. Heggernes. Minimal triangulations of graphs: A survey. *Discrete Math.*, 306(3):297–317, 2006.

[14] P. Heggernes, K. Suchan, I. Todinca, and Y. Villanger. Minimal interval completions. In *ESA 2005*, volume 3669 of *Lecture Notes in Computer Science*, pages 403–414. Springer, 2005.

[15] P. Heggernes, J. A. Telle, and Y. Villanger. Computing minimal triangulations in time $O(n^{\alpha} \log n) = o(n^{2.376})$. In *Proceedings of SODA 2005 - 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 907–916, 2005.

[16] H. Kaplan, R. Shamir, and R. E. Tarjan. Tractability of parameterized completion problems on chordal, strongly chordal, and proper interval graphs. *SIAM J. Comput.*, 28(5):1906–1922, 1999.

[17] D. Kratsch and J. P. Spinrad. Between $O(nm)$ and $O(n^{\alpha})$. In *Proceedings of SODA 2003 - 14th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 709–716, 2003.

[18] A. Natanzon, R. Shamir, and R. Sharan. A polynomial approximation algorithm for the minimum fill-in problem. *SIAM J. Comput.*, 30(4):1067–1079, 2000.

[19] A. Natanzon, R. Shamir, and R. Sharan. Complexity classification of some edge modification problems. *Disc. Appl. Math.*, 113:109–128, 2001.

[20] B. W. Peyton. Minimal orderings revisited. *SIAM J. Matrix Anal. Appl.*, 23(1):271–294, 2001.

[21] D. Rose, R.E. Tarjan, and G. Lueker. Algorithmic aspects of vertex elimination on graphs. *SIAM J. Comput.*, 5:146–160, 1976.

[22] K. Suchan and I. Todinca. Minimal interval completion through graph exploration. In *Proceedings ISAAC 2006*, 2006. Springer LNCS, to appear.

[23] K. Suchan, I. Todinca, *Pathwidth of circular-arc graphs.* Technical Report RR-2006-10, LIFO - University of Orleans, France, 2006.
http://www.univ-orleans.fr/SCIENCES/LIFO/rapports.php?lang=en.

[24] M. Yannakakis. Computing the minimum fill-in is NP-complete. *SIAM J. Alg. Disc. Meth.*, 2:77–79, 1981.

A Reduced folding : proofs of Theorem 3.11 and Lemma 3.12

Proof.(of Theorem 3.11) Recall that H_2 is a quasi-minimal interval completion of H_0 defined by a folding (H_0, Q_0). Let H_1 be an interval graph such that $H_0 \subseteq H_1 \subset H_2$ and H_1 is inclusion-maximal for this property. Observe that H_2 is a quasi-minimal interval completion of H_1, otherwise H_2 would not be a quasi-minimal interval completion of H_0. By Theorem 3.8, there is a folding (H_1, Q_1) of H_1 such that $H_2 = \texttt{FillFolding}(H_1, Q_1)$. We claim that (H_1, Q_1) is reduced.

By contradiction, suppose there is a pivot K of (H_1, Q_1) containing no simplicial vertex of H_1. Let P_1 be a clique-path of H_1 such that the two cliques next to K in P_1 are on the same side of K in Q_1. Let K' (resp. K'') denote the neighbor of K in P_1 which is closest (resp. furthest) to it in Q_1. Choose $u \in K \setminus K'$ and $v \in K' \setminus K$. The vertices u and v are not adjacent in H_1. Since u is not simplicial in H_1 we must have $u \in K''$. Consequently uv is an edge of H_2: indeed K' is between K and K'' in Q_1, $u \in K \cap K''$, thus in the graph $H_2 = \texttt{FillFolding}(H_1, Q_1)$, the set $K' \cup \{u\}$ induces a clique. Denote by H the graph obtained from H_1 by adding the edge uv. Add on the clique path P_1, a bag between K and K' containing the vertex subset $K \cap K' \cup \{u, v\}$; we obtain a clique-path of H. In particular H is an interval graph. If $H = H_2$ then H_2 is not a quasi-minimal completion of H_1 – a contradiction. It remains that H is strictly contained in H_2, but also $H_1 \subset H$, contradicting our choice of H_1. ∎

Proof. (of Lemma 3.12). Let x be a simplicial vertex of H_1 contained in the pivot K, so K is the unique maximal clique of H_1 containing x. In the path decomposition of H_2 produced by the algorithm `FillFolding` on (H_1, Q_1), the unique bag containing x is the bag corresponding to K. Therefore K is contained in a unique maximal clique of H_2. ∎

B Unfolding: proof of Theorem 4.5

We know from Theorem 3.11 that there exists a reduced folding (H_0, Q, P_0) satisfying $H_2 = \texttt{FillFolding}(H_0, Q)$ and $E(G) \subseteq E(H_0) \subset E(H_2)$. We may assume that for any unfolding (H_0, Q', P_0), the graph $H_2' = \texttt{FillFolding}(H_0, Q')$ is strictly contained in H_2.

We distinguish two different cases, as depicted in Figure 2. The maximal cliques of H_0 are numbered according to their order in the path P_0. The permutation Q corresponds to the ordering of the maximal cliques from left to right. The permutation Q is transformed in the permutation Q', as depicted below.

First case: a maximal clique at one end of Q is a pivot. Choose this as the left end and call the maximal clique K. (see the upper part of Figure 2.) Clearly, K is not one of the end maximal cliques in P_0. Let P^l be the subpath of P_0, from the left end to the clique K (included) and P^r be the subpath from K (included) to the right end of P_0. Let Q^l be the order defined by Q on the maximal cliques of P^l, and let Q^r be the order on the maximal cliques of P^r. Glue the reverse order of Q^l with Q^r in K (i.e, concatenate the reverse of Q^l with Q^r and merge the two consecutive copies of K) to obtain the new permutation Q' and the folding (H_0, Q', P_0).

Let us now argue that (H_0, Q', P_0) is an unfolding of (H_0, Q, P_0). The pivot K is not a pivot of the new folding, and clearly every pivot of (H_0, Q', P_0) is a pivot of (H_0, Q, P_0). We show that $H_1 = \texttt{FillFolding}(H_0, Q')$ is a strict subgraph of H_2. Let V^l and V^r be the union of maximal cliques in Q^l and Q^r, let $H^l = \texttt{FillFolding}(H[V^l], Q^l)$ and $H^r = \texttt{FillFolding}(H[V^r], Q^r)$. Clearly, H^l and H^r are subgraphs of H_2. The maximal clique K of H_0 separates, in the graph H_0, the vertex subsets V^l and V^r. Hence it also separates V^l and V^r in H_1; in particular K is a maximal clique of H_1. Each edge of H_1 is an edge of H^l or of H^r, thus the graph H_1 is a subgraph of H_2. By our assumption, the graph H_1, obtained by an unfolding of (H_0, Q, P_0), is a strict subgraph of H_2.

It remains to prove that there exists a reduced 1-folding (H_1, Q_1, P_1) of H_1 such that $H_2 = \texttt{FillFolding}(H_1, Q_1)$. Consider the path decomposition PD_1 of H_1 produced by the algorithm FillFolding on (H_0, Q'). The bags of PD_1 are in one-to-one correspondence with the maximal cliques of H_0, as they appear in the permutation Q'. In particular, the original permutation Q induces a permutation QB_1 on the bags of PD_1. Let Q_1 be obtained from QB_1 as the sublist of bags corresponding to maximal cliques of H_1. The clique-path P_1 is constructed by removing the bags of PD_1 which are not maximal cliques of H_1. By construction, (H_1, Q_1, P_1) is a folding of H_1 having K as unique fold, in particular it is a reduced folding. We show that $H_2 = \texttt{FillFolding}(H_1, Q_1)$. Suppose that we apply the algorithm FillFolding on (H_1, QB_1) (although QB_1 is not a permutation of the set of maximal cliques of H_1). We obtain the same graph as $\texttt{FillFolding}(H_0, Q_0)$. Indeed, assign to each vertex u the interval $[s_{QB_1}(u), t_{QB_1}(u)]$, where $s_{QB_1}(u)$ (resp. $t_{QB_1}(u)$) is the minimum (resp. maximum) index of a bag of QB_1 containing u. This interval is the same as $[s_Q(u), t_Q(u)]$, obtained by the same construction applied to the permutation Q. Now, by our construction of Q_1, it can be easily checked that the graph $\texttt{FillFolding}(H_1, Q_1)$ is the same as $\texttt{FillFolding}(H_1, QB_1)$. Hence $H_2 = \texttt{FillFolding}(H_1, Q_1)$.

Second case: neither of the maximal cliques at the ends of Q are pivots. Let K^l be the leftmost pivot, according to the ordering Q (see the lower part of Figure 2). Since no pivot of Q is the left or right most maximal clique, then there exists exactly one end of P_0 that contains maximal cliques to the right of K^l in Q, let this be the right end of P_0. Let K^r be the rightmost pivot in Q, such that K^l is to the right of K^r in P_0. Let P^l be the subpath of P_0, from the left end to the clique K^r included (we remind that K^r appears left to K^l in P_0). Similarly P^r is the subpath of P_0 from K_l (included) to the right end, and P^c is the subpath of P_0 starting in K^r and ending in K^l. Let Q^l (resp. Q^r, Q^c) be the order defined by Q on the maximal cliques of P^l (resp. P^r, P^c). Glue the reverse order of Q^r with Q^c in K_l and glue this with the reverse order of Q^l in K^r, and obtain the new order Q'' and folding (H_0, Q'', P_0). Ir remains to show that the unfolding (H_0, Q'', P_0) satisfies the required properties.

Like in the first case, (H_0, Q'', P_0) is an unfolding of (H_0, Q, P_0). Each pivot of (H_0, Q'', P_0), except for K^l and K^r, is a pivot in (H_0, Q, P_0). Indeed such a pivot is contained in Q^l, Q^c, or Q^r and the maximal cliques in Q^l, Q^c and Q^r induce connected sub paths of P_0. Let V^l, V^c, V^r be the union of maximal cliques in Q^l, Q^c, and Q^r, and let H^l, H^c and H^r be the graphs $\texttt{FillFolding}(H[V^l], Q^l)$, $\texttt{FillFolding}(H[V^c], Q^c)$ and $\texttt{FillFolding}(H[V^r], Q^r)$ respectively. $H_1 = \texttt{FillFolding}(H_0, Q'')$ is a subgraph of H_2, since every edge and vertex of H_1 is contained in either H^l, H^c, or H^r, and $H^l = H_2[V^l], H^c = H_2[V^c]$, and $H^r = H_2[V^r]$. We have assumed that for any unfolding of (H_0, Q, P_0), the filled graph is a strict subgraph of H_2. Consequently H_1 is a strict subgraph of H_2.

We prove that there exists a reduced 2-folding (H_1, Q_1, P_1) defining the graph H_2 (i.e., $H_2 = \texttt{FillFolding}(H_1, Q_1)$). The construction is very similar to the first case. Let PD_1 be the path decomposition of H_1 produced by the algorithm FillFolding on (H_0, Q''). Its bags are in one-to-one correspondence with the maximal cliques of H_0, as they appear in the permutation Q''. Let QB_1 be the permutation of these bags corresponding to the original permutation Q, and Q_1 be the sublist of QB_1 formed by the bags corresponding to maximal cliques of H_1. The clique-path P_1 is obtained from PD_1 by removing the bags which are not maximal cliques of H_1. The folding (H_1, Q_1, P_1) has only two folds, namely K^l and K^r. As in the first case, one can check that $H_2 = \texttt{FillFolding}(H_1, QB_1)$, and eventually $\texttt{FillFolding}(H_1, Q_1)$ yields the same graph as $\texttt{FillFolding}(H_1, QB_1)$. Consequently (H_1, Q_1, P_1) is a reduced 2-folding of H_1 such that $H_2 = \texttt{FillFolding}(H_1, Q_1)$.

C Proof of Theorem 4.7

Let us start with the following observation.

Observation C.1 *Let (H_1, Q, P_1) be a 2-folding that defines $H_2 = \texttt{FillFolding}(H_1, Q)$, and let P_2 be the clique path of H_2 obtained by the folding algorithm. Let K^r (K^l) be the leftmost (rightmost) pivot according to the ordering P_1 (hence K^l appears before K^r in Q). Let P^l be the subpath of P_1 formed by the cliques appearing strictly before K^l in Q and P^r be the subpath of P_1 formed by the cliques appearing strictly after K^r in Q. Then P^l and P^r are end paths of both P_1 and P_2.*

Proof. It follows directly from Algorithm `FillFolding` applied to (H_1, Q) that P^l and P^r are the end clique paths of P_2 as well. ∎

Proof.(of Theorem 4.7). Thus, if P_2 is the path decomposition of $H_2 = \texttt{FillFolding}(H_1, Q)$ obtained by the folding algorithm on a 2-folding (H_1, Q, P_1), we can write

$$P_1 = P^l - -P_1^l - -K^r - -P_1^c - -K^l - -P_1^r - -P^r \tag{1}$$

$$P_2 = P^l - -\Omega^l - -P_2^c - -\Omega^r - -P^r \tag{2}$$

Notice that a 1-folding can be treated as a 2-folding with one side of P_1, say $P_1^r - -P^r$, empty. In such a situation, K^l is not a pivot according to the previous definition, but if we extend it and define K^l to be a pivot, then the following argument holds for the case of 1-folding as well.

Suppose, on the contrary, that no pivot is a x, y-separator in H_1 for any fill edge xy. Let us first construct a clique path P_2' of H_2 based on P_1. We will use it later to prove that H_2 is not quasi-minimal.

Let P_2 be the clique path obtained by Algorithm `FillFolding` on (H_1, Q) (see Figure 1), so P_1 and P_2 are described by Equations 1, 2. Let $PL = P^l - -P_1^l - -K^r$, $PC = K^r - -P_1^c - -K^l$ and $PR = K^l - -P_1^r - -P^r$. Let xy be any fill edge with $x \in \mathrm{V}(PL) \setminus K^r$, then $y \in K^r$, since K^r is not a x, y-separator. So $N_{H_2}(\mathrm{V}(PL) \setminus K^r) \subseteq \mathrm{V}(PL)$. By Algorithm `FillFolding`, the bags of PL that have y added in the folding (H_1, Q) form an interval. We add y to the bags of this interval, for every such fill edge xy. In this way we simulate the folding (H_1, Q) on $H_1[\mathrm{V}(PL)]$ and obtain a clique path PL' of $H_2[\mathrm{V}(PL)]$. In an analogous way, we simulate the folding on PR and PC to obtain PR' and PC', clique paths of $H_2[\mathrm{V}(PR)]$ and $H_2[\mathrm{V}(PC)]$, respectively. Altogether, by gluing PL' with PC' and PR' on K^r and K^l, (and possibly removing redundant bags), and then reversing this path, we obtain a clique path P_2' of H_2. Indeed, $N_{H_2}(\mathrm{V}(PL) \setminus K^r) \subseteq \mathrm{V}(PL)$, $N_{H_2}(\mathrm{V}(PC) \setminus (K^r \cup K^l)) \subseteq \mathrm{V}(PC)$ and

$N_{H_2}(\mathrm{V}(PR) \setminus K^l) \subseteq \mathrm{V}(PR)$, so every edge of H_2 is contained in some bag of P_2'. Moreover, it is easy to verify that, for every vertex $x \in V$, the subset of bags containing x induces a subpath of P_2'.

Suppose $H_2[\mathrm{V}(PL')]$ contains a fill edge $x'y'$ with $y' \in K^r$. Let Ω be the maximal clique containing y' leftmost in P_2'. There is a vertex x'' which does not appear in a bag right of Ω, since Ω is a maximal clique. Moreover, $x''y' \notin E(H_1)$, since $x''y' \in E(H_1)$ implies $x'y' \in E(H_1)$ by the construction of P_2'. Notice that $\Omega \neq K^r$, since K^r is a maximal clique of $H_2[\mathrm{V}(PL')]$. Indeed, by definition of reduced folding, K^r contains a vertex v_s simplicial in H_2. It is easy to verify that, by construction of (H_1, Q, P_1), the closed neighborhood of v_s in $H_2[\mathrm{V}(PL')]$ is exactly K^r. Since Ω is the leftmost bag containing y', there is a clique $\Omega_{x'y'}$ containing both x' and y' between Ω and K^r in P_2'. By construction of P_2', if $x''y' \in E(H_1)$ then x' and y' appear together in a corresponding bag $K_{x'y'}$ of P_1 as well. So $x''y'$ is a fill edge. Let P_2'' be the clique path obtained from P_2' by removing Ω and putting $(\Omega \setminus \{x''\}) - -(\Omega \setminus \{y'\})$ instead. Clearly, it is a clique path of $H_2 - x''y'$, hence H_2 is not a quasi-minimal interval completion of H_1 - a contradiction. So $H_2[\mathrm{V}(PL')]$ does not contain any fill edges. An analogous argument shows that there is no fill edge in $H_2[\mathrm{V}(PR')]$.

Finlay, suppose only $H_2[\mathrm{V}(PC')]$ contains a fill edge $x'y'$. So one of the vertices, say y' is contained in one of the pivots, say K^r, in H_1. So $x' \in \mathrm{V}(PC) \setminus (K^r \cup K^l)$, since $H_2[\mathrm{V}(PL')]$ and $H_2[\mathrm{V}(PR')]$ contain no fill edges. Notice that y' is not contained in K^l, otherwise, by properties of clique path P_1, $x'y'$ would be an edge already in H_1. So, again, we pick Ω to be the leftmost in PC' maximal clique containing y' and a vertex x'' which does not appear in a bag right of Ω. Notice that $N_{H_2}(y') \cap \mathrm{V}(PR') = \emptyset$, since y' is not contained in K^l and there are no fill edges in $H_2[\mathrm{V}(PR')]$. Let P_2'' be the clique path obtained from P_2' by removing Ω and putting $(\Omega \setminus \{x''\}) - -(\Omega \setminus \{y'\})$ instead. Clearly, it is a clique path of $H_2 - x'y''$, which contradicts the quasi-minimality of H_2. A contradiction. ∎

D Two unfolding : proof of Lemma 5.2

For better reading, we rediscuss this case completely.

Let $H_2 = (V, E_2)$ be an quasi-minimal interval completion of a non-interval graph $G = (V, E)$. Suppose that H_2 is not minimal and choose the graph $H_1 = (V, E_1)$ such that $H_2 = \texttt{FillFolding}(H_1, Q)$, where (H_1, Q, P_1) is a reduced 2-folding and $E \subset E_1 \subset E_2$. Let P_2 be the clique path of H_2 obtained by the algorithm $\texttt{FillFolding}(H_1, P_1)$.

So by Observation C.1 we can denote these clique paths as:

$$P_1 = P^l - -P_1^l - -K^r - -P_1^c - -K^l - -P_1^r - -P^r$$

$$P_2 = P^l - -\Omega^l - -P_2^c - -\Omega^r - -P^r$$

where K^l, K^r are the pivots and Ω^r, Ω^l are the maximal cliques of H_2 containing them. We have $K^l \subseteq \Omega^l, K^r \subseteq \Omega^r$.

Let S^l (S^r) be the separator between P^l and Ω^l (Ω^r and P^r) in P_2. Notice that P^l, S^l, S^r, P^r appear also in P_1. Let B^l (B^r) be the interval of cliques that corresponds to the block of $H_2 - S^l$ ($H_2 - S^r$) that is contained in P^l (P^r), the closest to Ω^l (Ω^r) in P_2. So we have:

$$P_2 = P^{ll} - -B^l - -\Omega^l - -P_2^c - -\Omega^r - -B^r - -P^{rr} \tag{3}$$

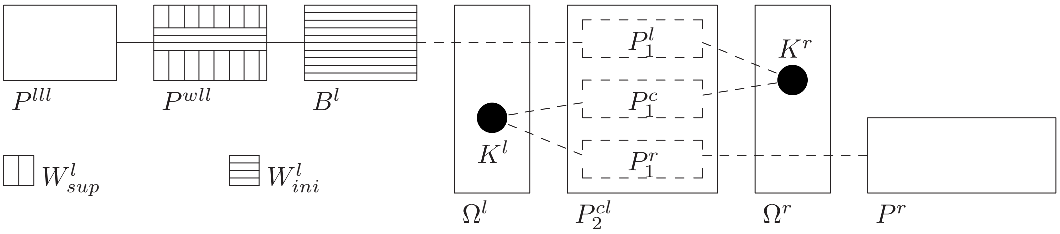

Figure 6: 2-folding

Notice that $C^l = \mathrm{V}(B^l) \setminus \Omega^l$ ($C^r = \mathrm{V}(B^r) \setminus \Omega^r$) is a connected component of $H_2 - \Omega^l$ ($H_2 - \Omega^r$). So, given H_2, Ω^l and Ω^l, we can effectively compute the candidates for C^l and C^r. From now on we assume that $\Omega^l, \Omega^r, C^l, C^r, S^l, S^r$ are as described above. We want to find an unfolding of H_2, an interval graph $H_1' = (V, E(H_2[N_G[W^l] \cup S^l]) \cup E(H_2[N_G[W^r] \cup S^r]) \cup E(H_2 - (W^l \cup W^r)))$, with some well chosen W^l, W^r.

Let uv be an edge of $E(H_2) \setminus E(H_1)$. Like in Theorem 4.7, u and v are chosen such that one of the pivots of H_1, say K^r, separates u and v in H_1 and also in the clique path P_1. Suppose w.l.o.g. that u is in $\mathrm{V}(P^l - -P_1^l) \setminus K^r$. In particular $u \notin \Omega^r$, and let C_u be the connected component of $G - \Omega^r$ containing the vertex u. Let $\mathcal{C}_v$ be the union of the connected components of $G - \Omega^r$ that contain or see the vertex v (we may have $v \in \Omega^r$, in which case there are several such components). We have proved:

Claim 1 *C_u is contained in $\mathrm{V}(P^l - -P_1^l) \setminus K^r$. $\mathcal{C}_v$ is contained in $\mathrm{V}(P_1^c - -K^l - -P_1^r - -P^r)$. Moreover, u and v do not appear in K^r.*

Definition D.1

$$W_{ini}^l = \bigcup \{C \mid C \in C(G - \Omega^r), C \cap B^l \neq \emptyset\} \cup C_u$$

$$W_{ini}^r = \bigcup \{C \mid C \in C(G - \Omega^l), C \cap B^r \neq \emptyset\}$$

When unfolding, we want the blocks corresponding to components of $H_2 - S^l$ that are in $P^{ll} - -B^l$ to stay blocks in $H_1' - S^l$. Let us investigate the connected components of $H_2 - S^l$. They induce a partition of $P^{ll} - -B^l$ into blocks of $H_2 - S^l$. Some of these blocks intersect connected components of $G - \Omega^r$ that are contained in W_{ini}^l. For example, all connected components of $G - \Omega^r$ that intersect B^l are also in W_{ini}^l. But there may be a block B^{lm} of $H_2 - S^l$ containing some components of $G - \Omega^r$ that are and some that are not in W_{ini}^l. In this case, B^{lm} is not a block of $H_1' - S^l$, where $H_1' = (V, E(H_2[N_G[W_{ini}^l]]) \cup E(H_2[N_G[W_{ini}^r]]) \cup E(H_2[V \setminus (W_{ini}^l \cup W_{ini}^r)]))$. In order to prevent this, we augment W_{ini}^l with W_{sup}^l, and W_{ini}^l with W_{sup}^l as defined below:

Definition D.2

$$W_{sup}^l = \bigcup \{C \mid C \in C(G - \Omega^r), C \cap \Omega^l = \emptyset, N_{H_2}[C] \cap W_{ini}^l \neq \emptyset, N_{H_2}[C] \cap (W_{ini}^r \cup \mathcal{C}_v) = \emptyset\}$$

$$W^l = W_{ini}^l \cup W_{sup}^l \tag{4}$$

$$W_{sup}^r = \bigcup \{C \mid C \in C(G - \Omega^l), C \cap \Omega^r = \emptyset, N_{H_2}[C] \cap W_{ini}^r \neq \emptyset, N_{H_2}[C] \cap W^l = \emptyset\}$$

$$W^r = W_{ini}^r \cup W_{sup}^r \tag{5}$$

We are now able to construct the unfolding.

Definition D.3

$$H'_1 = (V, E(H_2[N_G[W^l] \cup S^l]) \cup E(H_2[N_G[W^r] \cup S^r]) \cup E(H_2 - (W^l \cup W^r))).$$

Clearly, $B^l \subseteq N_G[W^l_{ini}] \cup S^l$, $B^r \subseteq N_G[W^r_{ini}] \cup S^r$. Moreover, $W^l_{ini} \subseteq \mathrm{V}(P^l - -P^l_1) \setminus K^r$ and $W^r_{ini} \subseteq \mathrm{V}(P^r_1 - -P^r) \setminus K^l$, so $W^l_{ini} \cap W^r_{ini} = \emptyset$. Notice that by similar argument $N(W^l_{ini}) \cap W^r_{ini} = \emptyset$ and $N(W^r_{ini}) \cap W^l_{ini} = \emptyset$. Moreover, by construction of $W^l \cap W^r$, this propagates to W^l and W^r:

Claim 2 $W^l \cap W^r = \emptyset$.

Let us now put some auxiliary notations.

Definition D.4 *Let* $W^c = (\Omega^l \cup \mathrm{V}(P^c_2) \cup \Omega^r) \setminus (W^l \cup W^r)$. *Let* $W^{ll} = \mathrm{V}(P^{ll})$ *and* $W^{rr} = \mathrm{V}(P^{rr})$.

Let us construct a path decomposition of the graph H'_1 in which all bags are cliques, let us call such a decomposition a *good path decomposition*, by first constructing good path decompositions of $H_2[N_G[W^{ll} \cup W^l]]$, $H_2[N_G[W^c]]$, $H_2[N[W^r \cup W^{rr}]]$, and then gluing them together.

Let us further refine the description of P^{ll} and denote by P^{lll} the subsequence of P^{ll} induced by blocks of $H_2 \setminus S^l$ that have empty intersection with W^l, an by P^{wll} the subsequence induced by blocks of $H_2 \setminus S^l$ contained in $N_G[W^l] \cup S^l$. The cliques contained in $N_G[W^l] \cup S^l$ are consecutive indeed. Take a clique Ω_2, $\Omega_2 \cap W^l_{ini} = \emptyset$, with another clique Ω_1, $\Omega_1 \cap W^l_{ini} \neq \emptyset$, to the left of it in P_2. Take $x \in \Omega_1 \cap W^l_{ini}$ and C_x, with $C_x \in C(G - \Omega^r), C_x \cap C^l \neq \emptyset, x \in C_x$. Since Ω_2 separates Ω_1 from B^l, it intersects C_x. Therefore $\Omega_2 \cap W^l_{ini} \neq \emptyset$ and $\Omega_2 \subseteq N_G[W^l] \cup S^l$.

Claim 3 $P^{wl} = P^{lll} - -P^{wll} - -B^l - -P^{lc}_2$, *where* P^{lc}_2 *comes from* P^c_2 *by removing from every bag the vertices not in* $N_G[W^l]$, *is a good path decomposition of* $H_2[N_G[W^{ll} \cup W^l]]$.

It follows the construction that $\mathrm{V}(P^{wll} - -B^l - -P^{lc}_2)$ is contained in $N_G[W^l] \cup S^l$. Moreover, $\mathrm{V}(P^{lll})$ is contained in $V \setminus (W^l \cup W^r)$. Indeed, by construction, there is: $\mathrm{V}(P^{lll}) \cap W^l = \emptyset$, $\mathrm{V}(P^{lll}) \cap W^r_{ini} = \emptyset$, $W^r_{sup} \subseteq \mathrm{V}(P^c_2 - -\Omega^r - -P^r) \setminus \Omega^r$, and $\mathrm{V}(P^{lll}) \cap S^l \subseteq \Omega^r$. So $\mathrm{V}(P^{lll}) \cap (W^l \cup W^r) = \emptyset$ and the corresponding cliques of H_2 remain cliques of H'_1. By construction, $N_G[W^{ll} \cup W^l] = \mathrm{V}(P^{wl})$. Moreover, for every $x \in \mathrm{V}(P^{wl})$, the cliques containing it induce an interval in P^{wl}, since they did in $P^{ll} - -B^l - -P^c_2$. In a similar manner, $P^{wr} = P^{rc}_2 - -B^r - -P^{wrr} - -P^{rrr}$ is a good path decomposition of $H_2[N_G[W^r \cup W^{rr}]]$.

Claim 4 $P^{wr} = P^{rc}_2 - -B^r - -P^{wrr} - -P^{rrr}$, *where* P^{rc}_2 *comes from* P^c_2 *by removing from every bag the vertices not in* $N_G[W^r]$, *is a good path decomposition of* $H_2[N_G[W^{rr} \cup W^r]]$.

Finally, let P^{cc}_2, Ω^{cl}, Ω^{cr} come from P^c_2, Ω^l, Ω^r by removing the vertices in $W^l \cup W^r$. Then, by reasoning similar to that above, we have:

Claim 5 $\Omega^{cl} - -P^{cc}_2 - -\Omega^{cr}$ *is a good path decomposition* $H_2[W^c]$.

Finally, we can glue these three good path decompositions together to obtain:

Claim 6 H'_1 *is an interval graph with a good path decomposition*

$$P'_1 = \overleftarrow{P^{wr}} - -\Omega^{cl} - -P^{cc}_2 - -\Omega^{cr} - -\overleftarrow{P^{wl}},$$

where the arrow indicates that the corresponding path has been reversed.

By construction and discussion above, the vertices of W^r appear only in bags of P^{wr} and the vertices of W^l appear only in bags of P^{wl}. Moreover $N_G(W^l) \cap W^r = \emptyset$ and $N_G(W^r) \cap W^l = \emptyset$. So removing $W^l \cup W^r$ from bags of $\Omega^l - -P_2^c - -\Omega^r$ to obtain $\Omega^{cl} - -P_2^{cc} - -\Omega^{cr}$ does not cause any discontinuity of the subgraph induced in P_1' by bags containing x, for any $x \in V$. Hence, P_1' is a good path decomposition. Removing redundant bags yields a clique path, so H_1' is an interval graph.

Claim 7 *uv is not an edge of H_1'.*

The graph H_1' is constructed from three edge subsets of H_2, we have to show that the edge uv is not in any of them.

First, we prove that $uv \notin E(H_2[N_G[W^l] \cup S^l])$. We have $v \notin N_G[W^l]$. Indeed $v \notin N_G[W^l_{ini}]$ because all the components of $G - \Omega^r$ forming W^l_{ini} are contained in $\mathrm{V}(P^l - -P_1^l)$, and $v \notin N_G[W^l_{sup}]$ by construction of W^l_{sup}. Also $v \notin S^l$ because v is contained in the subpath $P_1^c - -K^l - -P_1^r - -P^r$ of P_1, and $v \notin K^r$.

Second, uv is not in $E(H_2[N_G[W^r] \cup S^r])$. Indeed $u \notin N_G[W^r]$ by construction of W^r_{sup} and by the fact that W^r_{ini} is a subset of $\mathrm{V}(P_1^r - -P^r)$. Clearly $u \notin S^r$ by Claim 1.

Third, since $u \in W^l$ the edge uv does not appear in $E(H_2 - (W^l \cup W^r))$.

We proved in Claim 7 that the edge uv does not exist in H_1'. Algorithm `TwoUnfolding` tries all possibilities for Ω^l, Ω^r, S^l, S^r, C^l, C^r and u and v. At some iteration it will make the good choice and construct the graph H_1', an interval completion of G strictly contained in H_1.