

# ON-LINE DIMENSION OF SEMI-ORDERS

BARTŁOMIEJ BOSEK, KAMIL KLOCH, TOMASZ KRAWCZYK, AND PIOTR MICEK

ABSTRACT. We analyze the on-line dimension of partially ordered sets as a value of a two-person game between Algorithm and Spoiler. The game is played in rounds. Spoiler presents an on-line order of width at most  $w$ , one point at a time. Algorithm maintains its realizer, i.e., the set of  $d$  linear extensions which intersect to the presented order. Algorithm may not change the ordering of the previously introduced elements in the existing linear extensions. The value of the game  $\text{val}(w)$  is the least  $d$  such that Algorithm has a strategy against Spoiler presenting any order of width at most  $w$ . For interval orders Hopkins showed that  $\text{val}(w) \leq 4w - 4$ . We analyze the on-line dimension of semi-orders i.e., interval orders admitting a unit-length representation. For up-growing semi-orders of width  $w$  we prove a matching lower and upper bound of  $w$ . In the general (not necessarily up-growing) case we provide an upper bound of  $2w$ .

## 1. INTRODUCTION

Let  $\mathbf{P}$  be a finite partially ordered set. The *width*  $\mathbf{P}$  is the size of the largest antichain in  $\mathbf{P}$  and the height of  $\mathbf{P}$  is the size of the largest chain in  $\mathbf{P}$ . By Dilworth's theorem any order of width  $w$  can be partitioned into  $w$  chains and dually, any order of height  $h$  can be partitioned into  $h$  antichains. When  $\mathbf{P} = (X, P)$  and  $\mathbf{Q} = (X, Q)$  are partial orders on the same set  $X$  then we call  $\mathbf{Q}$  an *extension* of  $\mathbf{P}$  if  $P \subseteq Q$ , i.e., if  $x \leq y$  in  $\mathbf{P}$  implies  $x \leq y$  in  $\mathbf{Q}$ , for all  $x, y \in X$ . Among all extensions of a given poset, those which are additionally linear orders are of special importance. They are called *linear extensions*. For a poset  $\mathbf{P}$  consisting of  $n$  elements  $x_1, \dots, x_n$  we write  $L = (x_1, \dots, x_n)$  as an abbreviation for a linear extension  $\mathbf{L} = (X, L)$  of  $\mathbf{P}$  in which  $x_1 < \dots < x_n$ . A set  $\mathcal{R}$  of linear extensions of a poset  $\mathbf{P}$  intersecting to  $\mathbf{P}$  is called a *realizer* of  $\mathbf{P}$ . The *dimension* of a poset  $\mathbf{P}$ , denoted by  $\text{dim}(\mathbf{P})$ , is the least integer  $k$  for which there exists a realizer of  $\mathbf{P}$  consisting of  $k$  linear extensions. Another classic theorem of Dilworth says that  $\text{dim}(\mathbf{P}) \leq \text{width}(\mathbf{P})$ . The notion of dimension in posets is often compared with chromatic number of graphs. Dimension theory greatly influenced the research on combinatorial properties of posets and graphs. For a comprehensive account on the topic and an extensive bibliography work we refer the reader to Trotter's monograph [Tro92].

All mentioned basic parameters of orders: width, height and dimension have their witnessing structures. These structures in are: chain decomposition, antichain decomposition and a realizer, all of the smallest possible size equal to the respective parameter. In the on-line setting the sole question about the value of those parameters does not pose new challenge as all of them can be computed after each round of the game exactly in the same way as in the off-line case. Therefore, instead of asking only for the scalar values we additionally require to build (and update on-line) an appropriate witnessing structure.

From the three on-line problems introduced, the majority of researchers' efforts has been devoted to the on-line width game. However some of the results in this field are very much related to the on-line dimension problem. Therefore, before

---

*Date:* July 3, 2009.

*Key words and phrases.* on-line, dimension, semi-order.

drawing our attention to the latter, we give a brief overview of the on-line width and on-line height results.

The *on-line width* is the value of the following two-person game. We call the players Spoiler and Algorithm. First Spoiler announces parameter  $w$  – the width of the order to be presented. Algorithm responds with  $k$  – the size of the chain partition maintained in the game. The game is then played in rounds. In each round Spoiler presents a new point enlarging the order of width at most  $w$  presented so far. Algorithm, in turn, makes an irrevocable assignment of this point to some chain in the maintained chain partition. He loses the game if the new point cannot be incorporated into any of the maintained  $k$  chains. The value of the game  $\text{val}(w)$ , for width  $w$  orders, is the least  $k$  such that there is a winning strategy for Algorithm using at most  $k$  chains on orders of width at most  $w$ .

The study of chain partitioning games reaches back to the early 80's when Kierstead [Kie81] and Szemerédi (published in [Kie86]) proved so far the best estimates for on-line orders of width  $w$ :  $\binom{w+1}{2} \leq \text{val}(w) \leq \frac{5^w-1}{4}$ . This is already a complicated result and no progress has been made in the last 25 years. Better estimates are only known for small values of  $w$ . In [Kie81] Kierstead showed that 5 is a lower and 6 an upper bound in the case of orders of width 2. Felsner [Fel97] closed the gap by reducing the upper bound to 5. For orders of width 3, Bosek [Bos08] improved the upper bound from 31 to 16. The big question remains whether  $\text{val}(w)$  is bounded by a polynomial of  $w$ .

Since the chain partitioning problem resists the attempts to be solved in its general form, some restricted variants are being analyzed. Felsner [Fel97] proposed a modified chain partitioning game in which Spoiler's power is limited by the condition that the new element has to be a maximal element of the order presented so far. In other words, a possible comparability of a new element  $x$  to an old element  $y$  has to be of the form  $x > y$ . On-line posets with this property are called *up-growing*. Felsner [Fel97] showed that the value of the chain partitioning game for up-growing orders is  $\binom{w+1}{2}$ .

Another way of simplifying the problem is to narrow the class of posets that is being considered. In particular, interval orders and semi-orders are of our interest. Poset  $\mathbf{P} = (X, \leq)$  is an *interval order* if it can be represented as a set of intervals of the real line so that for  $x, y \in X$  we have  $x < y$  if and only if the interval corresponding to  $x$  is entirely to the left of the the interval corresponding to  $y$ . *Semi-orders*, in turn, are interval orders which admit a unit-length interval representation. The precise result  $\text{val}(w) = 3w - 2$  for the on-line chain partitioning game of interval orders was settled in the early 80's by Kierstead and Trotter [KT81]. Recently, Baier et al. [BBM07] resolved the case of up-growing interval orders, they proved  $\text{val}(w) = 2w - 1$ . For up-growing semi-orders, Felsner et al. [FKMM] proved a matching lower and upper bound of  $\lfloor \frac{1+\sqrt{5}}{2} w \rfloor$ .

The analysis of the on-line height of posets happens to be much easier. This time Spoiler announces the height of the order to be presented and Algorithm maintains its antichain decomposition. The value of the game  $\text{val}(h)$  is the least integer  $k$  such that there is a winning strategy for Algorithm using  $k$  antichains on orders of height at most  $h$ . Schmerl and Szemerédi proved that  $\text{val}(h) = \binom{h+1}{2}$  (published in Kierstead's [Kie86], see also p. 441 of [Tro95]). The beautiful result of Gyárfás and Lehel [GL88]: 'any chordal graph  $\mathbf{G}$  can be covered on-line with  $2\alpha(\mathbf{G}) - 1$  cliques (where  $\alpha(\mathbf{G})$  is the maximal size of independent set in  $\mathbf{G}$ )' implies that the on-line height of interval orders is at most  $2h - 1$  and this bound is tight (see [KQ95]).

We finally focus on the on-line dimension of posets. Like above, the problem is being defined by means of our two-person game. This time Algorithm fixes the size

$k$  of the on-line realizer he is going to maintain. Spoiler wins if Algorithm cannot incorporate the new point into  $k$  linear extensions while keeping them a realizer.

Knowing only the dimension  $d$  of order presented by Spoiler, the above game renders hopeless for Algorithm.

**Proposition 1.** *For every  $k$  there is a strategy for Spoiler presenting an on-line order of dimension 2 and winning with Algorithm using a realizer of size  $k$ .*

*Proof.* For a fixed  $k$  Spoiler introduces a  $(k + 1)$ -element antichain  $a_1, \dots, a_{k+1}$ . At least one of the  $a_i$ 's is not on the top of any linear extension in Algorithm's realizer, say  $a_1$ . Now Spoiler presents point  $b$  above all the  $a_i$ 's except  $a_1$ . Algorithm has to put  $b$  on the top of all maintained  $k$  linear extensions. On the other hand, since  $a_1$  and  $b$  are incomparable, we must have  $b < a_1$  in at least one of those linear extensions. Clearly, this is impossible. The on-line order presented by Spoiler has dimension 2 (see Figure 1).  $\square$

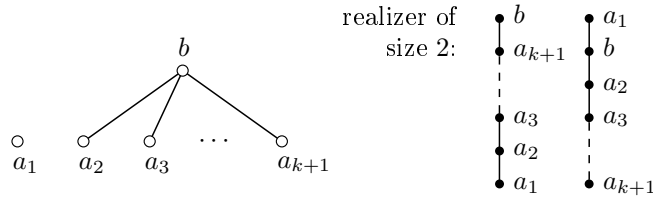


FIGURE 1. Order of dimension 2 presented by Spoiler

The width of the order presented by Spoiler in the above strategy depends on the number  $k$  of linear extensions used by Algorithm (see Figure 1). One can ask if knowing the width  $w$  of the presented order Algorithm could produce a realizer whose size would be bounded in terms of  $w$ ? The answer, given in [KMT84], is again: no. The winning strategy for Spoiler produces a poset with a 3-crown (see Figure 2). (A  $n$ -crown is a poset with points  $\{a_1, \dots, a_n, b_1, \dots, b_n\}$  such that  $a_i, a_{i+1} < b_i$  for  $i < n$  and  $a_n, a_1 < b_n$ ). In the very same paper it is proved that if Spoiler presents an on-line order of width  $w$  without an  $n$ -crown as a subposet for any  $n \geq 3$  then the number of linear extensions needed by Algorithm is indeed bounded in terms of  $w$ .

**Theorem 2** (Kierstead, McNulty, Trotter [KMT84]).

- (i) *For every  $k$  there is a strategy for Spoiler presenting an on-line order of width 3 and winning with Algorithm using a realizer of size  $k$ .*
- (ii) *For on-line orders of width at most  $w$  and without an  $n$ -crown as a subposet for  $n \geq 3$  there is a strategy for Algorithm using  $c!$  linear extensions, where  $c$  is the value of on-line chain partitioning game on width  $w$ , i.e.,  $c \leq \frac{5^w - 1}{4}$ .*

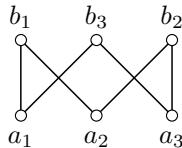


FIGURE 2. A 3-crown

Results from Theorem 2 lead to the structural gap left in the area: what happens if only 3-crowns are being forbidden? Does Algorithm have a winning strategy (in terms of  $w$ ) on 3-crown-free on-line orders of width  $w$ ? Another emerging

question is how good strategies (in terms of  $w$ ) for Algorithm can be devised on other (narrowed) classes of orders defined in terms of forbidden structures, like interval orders or semi-orders?

Formally, the on-line dimension of a class  $\mathcal{C}$  of orders, or the *value* of the on-line dimension game for  $\mathcal{C}$ , denoted by  $\text{val}(w)$ , is the least integer  $k$  for which Algorithm has a winning strategy using  $k$  chains on orders from  $\mathcal{C}$  of width at most  $w$ .

Recall that poset  $\mathbf{P} = (X, \leq)$  is an *interval order* if there is a function  $I$  which assigns to each  $x \in X$  a closed interval  $I(x) = [l_x, r_x]$  of the real line so that  $x < y$  in  $\mathbf{P}$  if and only if  $I(x) < I(y)$ , i.e.,  $r_x < l_y$ . An interval representation  $I$  is *correlating* if elements sharing the same sets of predecessors (successors) are mapped to intervals with the same left (right) end points. It is easy to see that every finite interval order admits a correlating interval representation. For an integer  $n \geq 2$ , let  $\mathbf{I}_n$  denote the *canonical* interval order determined by the set of all closed intervals with distinct integer end points from  $\{1, \dots, n\}$ .

An interval order  $\mathbf{P} = (X, \leq)$  is called a *semi-order* if it has an interval representation  $\{[l_x, r_x] : x \in X\}$  such that  $r_x = l_x + 1$  for every  $x \in X$ . By possibly locally stretching some of the intervals one can easily show that  $\mathbf{P}$  is a semi-order if and only if it admits a *proper* interval representation, i.e., a representation in which no interval is properly contained in another one.

It is not entirely naive to ask whether there are interval orders which have large dimension. However, it can be proved that  $\dim(\mathbf{I}_n) = \log \log(n) + (\frac{1}{2} + o(1)) \log \log \log(n)$  (see [Tro92]). On the other hand Füredi, Hajnal, Rödl and Trotter [FHRT92] bounded from above the dimension of any interval order of height  $h$  by  $\log \log h + (\frac{1}{2} + o(1)) \log \log \log h$ . The dimension of any semi-order is at most 3 (by Rabinovitch [Rab78]).

The on-line dimension of interval orders is still far from being understood. As interval orders do not induce  $n$ -crown for  $n \geq 3$  their on-line dimension is bounded in terms of the width (by Theorem 2). The results of Hopkins [Hop81] and Kierstead, McNulty, Trotter [KMT84] give us some better bounds (see Table 1).

up-growing	bounds	remarks
–	$\frac{4}{3}w \leq ? \leq 4w - 4$	[KMT84], [Hop81]
+	$\frac{8}{7}w \leq ? \leq 2w - 2$	unpublished results of authors

TABLE 1. On-line dimension of interval orders

For semi-orders the lower bound  $\frac{4}{3}w$  still applies, while the upper bound of  $2w$  can be derived from the argument of Hopkins (see Section 3). In this paper we prove that the value of the on-line dimension game is exactly  $w$  (see Table 2). The lower bound is trivial and follows from Proposition 1. The proof of the upper bound is split into two parts, to better pinpoint its difficulties. First, in Section 4, we show a more relaxing bound of  $w + 1$ . The improved result of  $w$  follows thereafter in Section 5.

On-line dimension game on interval orders (semi-orders) has a natural variant in which Spoiler, instead of points, reveals the underlying interval representation of the order. This of course provides more information to Algorithm and typically the value of the resulting game is different. Authors cover this new variant of the game in another manuscript [BKKM].

up-growing	bounds	remarks
–	$\frac{4}{3}w \leq ? \leq 2w$	[KMT84], Theorem 9
+	$w$	Theorem 18

TABLE 2. On-line dimension of semi-orders

## 2. BASICS

For a poset  $\mathbf{P} = (X, \leq)$  and  $x \in X$  we denote the set of all predecessors of  $x$  in  $\mathbf{P}$  by  $x\downarrow$  and the set of all its successors in  $\mathbf{P}$  by  $x\uparrow$ , i.e.:  $x\downarrow = \{y : y < x \text{ in } \mathbf{P}\}$  and  $x\uparrow = \{y : y > x \text{ in } \mathbf{P}\}$ .

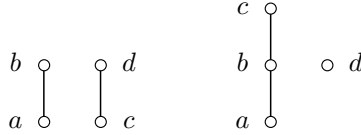
We will make an extensive use of the well-known characterization of interval orders and semi-orders in terms of forbidden structures.

**Theorem 3.** *Let  $\mathbf{P} = (X, \leq)$  be a poset. The following statements are equivalent.*

- (i)  $\mathbf{P}$  is an interval-order,
- (ii)  $\mathbf{P}$  is a  $(\mathbf{2} + \mathbf{2})$ -free order, i.e.,  $\mathbf{P}$  does not contain elements  $a, b, c, d \in X$  such that  $a < b, c < d$ , with no more comparabilities (see Figure 3),
- (iii) For any  $x, y \in X$  either  $x\downarrow \subseteq y\downarrow$  or  $x\downarrow \supseteq y\downarrow$ , i.e., the down sets are linearly ordered with respect to inclusion.
- (iv) For any  $x, y \in X$  either  $x\uparrow \subseteq y\uparrow$  or  $x\uparrow \supseteq y\uparrow$ , i.e., the up sets are linearly ordered with respect to inclusion.

**Theorem 4** (Scott and Suppes [SS58]). *Let  $\mathbf{P} = (X, \leq)$  be an interval order. Then the following statements are equivalent.*

- (i)  $\mathbf{P}$  is a semi-order,
- (ii)  $\mathbf{P}$  is a  $(\mathbf{3} + \mathbf{1})$ -free order, i.e.,  $\mathbf{P}$  does not contain elements  $a, b, c, d \in X$  such that  $a < b < c$  and  $d \parallel a, b, c$  (see Figure 3).

FIGURE 3.  $(\mathbf{2} + \mathbf{2})$  and  $(\mathbf{3} + \mathbf{1})$  orders

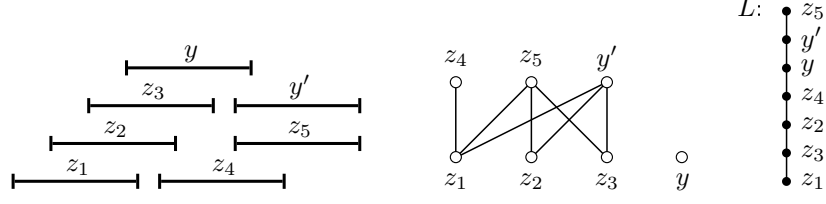
Let  $\mathbf{P}$  be a semi-order and let  $x, y \in X$ . Theorem 4 allows us to define an (irreflexive) extension  $\prec$  of  $\mathbf{P}$  as follows.

$$x \prec y \text{ iff } x\downarrow \subsetneq y\downarrow \text{ or } x\uparrow \supsetneq y\uparrow.$$

Reflexive extension of  $\prec$  is defined so that  $x \preceq y$  iff  $x \prec y$  or  $x$  and  $y$  have the same up sets and down sets. Up to twin equivalence the ordering  $\preceq$  define a linear extension of  $\mathbf{P}$ .

Let  $\mathbf{P} = (X, \leq)$  be a semi-order and  $L$  be a linear extension of  $\mathbf{P}$ . We say that point  $y \in X$  is *strictly high* in  $L$  if for every  $z \in X$ ,  $y < z$  in  $L$  implies  $y < z$  in  $\mathbf{P}$ . Point  $y \in X$  is *high* in  $L$  if for every  $z \in X$ ,  $y < z$  in  $L$  implies  $y \prec z$  in  $\mathbf{P}$  (equivalently,  $z \preceq y$  in  $\mathbf{P}$  implies  $z \leq y$  in  $L$ ). In other words, if  $I$  is a correlating interval representation of  $\mathbf{P}$  then  $y$  is high in  $L$  if  $z <_L y$  for every  $z$  lying weakly to the left of  $y$ , i.e., for every  $z \preceq y$  and  $z \neq y$  (see Figure 4).

By the *dual* of a poset  $\mathbf{P} = (X, P)$  we mean a poset  $\mathbf{P}^d = (X, P^d)$  where  $(x, y) \in P^d$  iff  $(y, x) \in P$ . Using the concept of the dual poset  $\mathbf{P}^d$  of  $\mathbf{P}$  and the linear extension  $L^d$  of  $\mathbf{P}^d$  we introduce two more definitions.

FIGURE 4.  $y$  is high in  $L$ ,  $y'$  is not high in  $L$ 

Point  $x$  is *low* (strictly low) in  $L$  if and only if it is high (strictly high) in  $L^d$ .

Let  $\mathbf{P}' = (X \cup \{x\}, \leq')$  be an extension of  $\mathbf{P}$ . Define  $L'_{\text{high}}$  and  $L'_{\text{down}}$ , two linear extensions of  $\mathbf{P}'$  obtained from a linear extension  $L$  of  $\mathbf{P}$ :

$L'_{\text{high}}$ : add  $x$  *high* in  $L$ , i.e., let  $x$  go as deep as possible in  $L$  so that  $x$  is high in  $L'_{\text{high}}$ , that is, put  $x$  just above the largest  $y$  in  $L$  for which  $y \leq x$ ,

$L'_{\text{down}}$ : add  $x$  *strictly low* in  $L$ , i.e., let  $x$  go as deep as possible in  $L$  so that  $L'_{\text{down}}$  remains a linear extension of  $\mathbf{P}'$ , that is, put  $x$  just above the largest  $y$  in  $L$  for which  $y <_{\mathbf{P}'} x$ .

Three important properties of these extensions are summarized in the following observation.

**Observation 5.** *Let semi-orders  $\mathbf{P}$ ,  $\mathbf{P}' = (X \cup \{x\}, \leq')$  and linear extensions  $L$ ,  $L'_{\text{high}}$  and  $L'_{\text{down}}$  be defined as above. If  $y \in X$  is high in  $L$  then*

- (i)  $y$  is high in  $L'_{\text{high}}$  if and only if  $x$  and  $y$  are not twins,
- (ii)  $y$  is high in  $L'_{\text{down}}$ ,
- (iii) if  $x \parallel y$  then  $x < y$  in  $L'_{\text{down}}$ .

*Proof.* Let  $h$  and  $d$  be the points immediately preceding  $x$  in the linear extensions  $L'_{\text{high}}$  and  $L'_{\text{down}}$ , respectively.

If  $y \prec x$  then, according to definition of being high,  $y$  remains high in *any* linear extension  $L'$  of  $\mathbf{P}'$  obtained from  $L$ . In particular,  $y$  remains high in  $L'_{\text{high}}$  and  $L'_{\text{down}}$ .

For  $x \preceq y$  we have  $d < x \preceq y$ , so  $d < y$ . Hence  $x < y$  in  $L'_{\text{down}}$  as  $x$  is just above  $d$  in  $L'_{\text{down}}$  and  $y$  must be above  $d$  as well. This proves (ii).

In order to prove (i) assume that  $x \prec y$ . Then  $h \preceq x \prec y$ . Since  $y$  is high in  $L$ , we get  $h < y$  in  $L$ . Point  $x$  is just above  $h$  in  $L'_{\text{high}}$ , so that  $x < y$  in  $L'_{\text{high}}$  and therefore  $y$  remains high in  $L'_{\text{high}}$ .

To prove the converse in (i) assume that  $x \approx y$ . Since  $x$  is high in  $L'_{\text{high}}$ , it dominates  $y$  in  $L'_{\text{high}}$  and  $y$  is no longer high in  $L'_{\text{high}}$ .

It remains to prove (iii). Recall that  $d < x$ . If  $x \parallel y$  then  $d \prec y$ . Since  $y$  is high in  $L$ , we know that  $d < y$  in  $L$ . This implies  $x < y$  in  $L'_{\text{down}}$  and proves (iii).  $\square$

Define linear extensions  $L'_{\text{low}}$  and  $L'_{\text{up}}$  dually to  $L'_{\text{high}}$  and  $L'_{\text{down}}$ , respectively.

$L'_{\text{low}}$ : add  $x$  *low* in  $L$ , i.e., let  $x$  go as high as possible in  $L$  so that  $x$  is low in  $L'_{\text{low}}$ , that is, put  $x$  just below the smallest  $y$  in  $L$  for which  $y \succ x$ ,

$L'_{\text{up}}$ : add  $x$  *strictly high* in  $L$ , i.e., let  $x$  go as high as possible in  $L$  so that  $L'_{\text{up}}$  remains a linear extension of  $\mathbf{P}'$ , that is, put  $x$  just below the smallest  $y$  in  $L$  for which  $y >_{\mathbf{P}'} x$ .

Clearly, Observation 5 has its counterpart for linear extensions for  $L'_{\text{low}}$  and  $L'_{\text{up}}$ .

## 3. GENERAL CASE

Let  $\mathbf{P} = (X, \leq)$  be a semi-order and  $\mathcal{R}$  be its realizer. Suppose that  $\mathbf{P}$  is extended to  $\mathbf{P}' = (X \cup \{x\}, \leq)$  so that  $\text{width}(\mathbf{P}') = \text{width}(\mathbf{P})$ . Our first concern is to understand under which conditions for  $\mathcal{R}$  this new point  $x$  can be inserted to existing linear extensions from  $\mathcal{R}$  without creating a new one, so the new set  $\mathcal{R}'$  of extensions is a realizer of  $\mathbf{P}'$ . Clearly, for every  $y \in \text{Inc}(x)$  there must exist  $L'_i, L'_j \in \mathcal{R}'$  in which  $x < y$  and  $x > y$ , respectively. It is easy to observe that instead of considering all points from  $\text{Inc}(x)$  it suffices to focus on minimal and maximal points in  $\text{Inc}(x)$  in order to get that  $\mathcal{R}'$  is a realizer of  $\mathbf{P}'$ . This easy fact is worded in the following observation.

**Observation 6.** *In the setting described above,  $\mathcal{R}'$  is a realizer of  $\mathbf{P}'$  if for every  $y \in \min(\text{Inc}(x))$  there exists  $L'_i \in \mathcal{R}'$  in which  $x < y$  and for every  $y \in \max(\text{Inc}(x))$  there exists  $L'_j \in \mathcal{R}'$  in which  $x > y$ .*

It turns out that updating  $\mathcal{R}$  to  $\mathcal{R}'$  (of the same size) is not always possible without further assumptions on the behavior of  $\mathcal{R}$ . Consider the following example. Let  $\mathbf{P} = (\{a, b, c\}, \leq)$  be a 3-element antichain. Define  $L_1 = (a, b, c)$ ,  $L_2 = (c, b, a)$ ,  $L_3 = (b, a, c)$ . If Spoiler presents  $x$  so that  $x > a, c$  and  $x \parallel b$  then  $x$  cannot be added to  $L_1, L_2, L_3$  so that  $\mathcal{R}'$  is a realizer of  $\mathbf{P}'$  as  $x$  has to be below  $b$  in one of the  $L'_i$ 's but then  $x$  would be also below  $a$  or  $c$ .

The problem in the above example arises from the fact that there is no  $L_i$  in which  $x$  could be put below  $b$ . On the other hand, if  $b$  had been high in some  $L_j$  then putting  $x$  as deep as possible in  $L_j$  would guarantee  $x < b$  in  $L'_j$ , by Observation 5(iii). The latter remark is the reason for introducing the concept of *coloring* functions. Maintaining such functions during on-line dimension game will guarantee the possibility of extending the realizer without increasing its size.

Let  $\mathbf{P} = (X, \leq)$  be a semi-order of width at most  $w$  and  $\mathcal{R} = \{L_1, \dots, L_w\}$  be the set of  $w$  linear extensions of  $\mathbf{P}$ . A function  $c : X \rightarrow \{1, \dots, w\}$  is a *high-coloring* function for  $(\mathbf{P}, \mathcal{R})$  if  $y$  is high in  $L_{c(y)}$ , for all  $y \in X$ . We usually say that  $y$  has color  $c(y)$ , while obviously  $y$  may be high in some other linear extension from  $\mathcal{R}$  as well. Dually, a *low-coloring* function  $c : X \rightarrow \{1, \dots, w\}$  for  $(\mathbf{P}, \mathcal{R})$  satisfies that  $y$  is low in  $L_{c(y)}$ , for any  $y \in X$ . It is easy to see that a low-coloring function for  $(\mathbf{P}, \mathcal{R})$  is simultaneously a high-coloring function for  $(\mathbf{P}^d, \mathcal{R}^d)$ , where  $\mathcal{R}^d = \{L^d : L \in \mathcal{R}\}$ .

Let  $c : X \rightarrow \{1, \dots, w\}$  be a high-coloring function for  $(\mathbf{P}, \mathcal{R})$ . Suppose that  $\mathbf{P}$  is extended to  $\mathbf{P}' = (X \cup \{x\}, \leq)$  so that  $\text{width}(\mathbf{P}') = w$ . As previously suggested, Algorithm can use the function  $c$  to insert  $x$  below all points incomparable with  $x$ . However, in order to maintain the high-coloring function  $c'$  for  $(\mathbf{P}', \mathcal{R}')$ , the new point  $x$  must be put high in some linear extension. Since  $\{x\} \cup \min(\text{Inc}(x))$  is an antichain in  $\mathbf{P}'$ , we have  $|\min(\text{Inc}(x))| < w$ . Hence there is a color in  $\{1, \dots, w\}$  but not in  $c(\min(\text{Inc}(x)))$  which can possibly be used to color  $x$  highly. We verbalize this inspiration in the following proposition.

**Proposition 7.** *In the setting defined above, fix  $i_0 \in \{1, \dots, w\} - c(\min(\text{Inc}(x)))$  and define  $\mathcal{R}' = \{L'_1, \dots, L'_w\}$  as follows:*

- put  $x$  high in  $L_{i_0}$ , i.e.,  $L'_{i_0} := (L_{i_0})'_{\text{high}}$ ,
- put  $x$  down in  $L_i$ , for all  $i \neq i_0$ , i.e.,  $L'_i := (L_i)'_{\text{down}}$ .

*Then there is a high-coloring function  $c'$  for  $(\mathbf{P}', \mathcal{R}')$  with  $c'|_X = c$ . Moreover, for each  $y \in \min(\text{Inc}(x))$  there is  $L'_i \in \mathcal{R}'$  in which  $x < y$ .*

*Proof.* To see the latter note that by Observation 5(iii) we have  $x < y$  in  $L'_{c(y)}$  for  $y \in \min(\text{Inc}(x))$ . Define  $c'$  by:

$$c'(y) = \begin{cases} c(y), & \text{if } y \in X, \\ i_0, & \text{if } y = x. \end{cases}$$

To prove that  $c'$  is a high-coloring function for  $(\mathbf{P}', \mathcal{R}')$  we need to show that  $y$  is high in  $L'_{c'(y)}$  for every  $y \in X \cup \{x\}$ . Clearly,  $x$  is high in  $L'_{i_0}$ . Let  $y \in \min(\text{Inc}(x))$ . Point  $x$  was put down in  $L_{c(y)}$ . From Observation 5(ii) we get that  $y$  remains high in  $L'_{c'(y)}$ . Finally, let  $y \in X - \min(\text{Inc}(x))$ . Then  $x$  and  $y$  are not twins. Again, from Observation 5(i), it follows that  $y$  is high in  $L'_{c'(y)}$ .  $\square$

Obviously, Proposition 7 has its dual counterpart for a low-coloring function:

**Proposition 8.** *For a low-coloring function  $c : X \rightarrow \{1, \dots, w\}$  of  $(\mathbf{P}, \mathcal{R})$  fix  $i_0 \in \{1, \dots, w\} - c(\max(\text{Inc}(x)))$  and define  $\mathcal{R}' = \{L'_1, \dots, L'_w\}$  as follows:*

- put  $x$  low in  $L_{i_0}$ , i.e.,  $L'_{i_0} := (L_{i_0})'_{\text{low}}$ ,
- put  $x$  strictly high in  $L_i$ , for all  $i \neq i_0$ , i.e.,  $L'_i := (L_i)'_{\text{up}}$ .

*Then there is a low-coloring function  $c'$  for  $(\mathbf{P}', \mathcal{R}')$  with  $c'|_X = c$ . Moreover, for each  $y \in \max(\text{Inc}(x))$  there is  $L'_i \in \mathcal{R}'$  in which  $x > y$ .*

Combining Propositions 7 and 8 we get the next theorem.

**Theorem 9.** *Let  $\text{val}(w)$  denote the value of the on-line dimension game on the class of semi-orders of width at most  $w$ , presented without representation. Then*

$$\text{val}(w) \leq 2w.$$

*Proof.* Let  $\mathbf{P} = (X, \leq)$  be a semi-order of width at most  $w$ . To maintain a realizer  $\mathcal{R}$  of size  $2w$  we actually require that  $\mathcal{R}$  can be split into  $\mathcal{R} = \mathcal{R}_l \cup \mathcal{R}_h$  with  $|\mathcal{R}_l| = |\mathcal{R}_h| = w$  such that there is a low-coloring function  $c_l : X \rightarrow \{1, \dots, w\}$  for  $(\mathbf{P}, \mathcal{R}_l)$  and a high-coloring function  $c_h : X \rightarrow \{1, \dots, w\}$  for  $(\mathbf{P}, \mathcal{R}_h)$ . The realizer  $\mathcal{R}$  and the functions  $c_l, c_h$  are trivially defined when  $|X| \leq 1$ .

Consider an extension  $\mathbf{P}' = (X \cup \{x\}, \leq)$  of  $\mathbf{P}$  such that  $\text{width}(\mathbf{P}') \leq w$ . Define  $\mathcal{R}'_h$  and  $c'_h$  as in the proof of Proposition 7 so that  $c'_h$  is a high-coloring function for  $(\mathbf{P}', \mathcal{R}'_h)$  and for each  $y \in \min(\text{Inc}(x))$  there is  $H' \in \mathcal{R}'_h$  in which  $x < y$ . Dually, define  $\mathcal{R}'_l$  and  $c'_l$  as in the proof of Proposition 8 so that  $c'_l$  is a low-coloring function for  $(\mathbf{P}', \mathcal{R}'_l)$  and for each  $y \in \max(\text{Inc}(x))$  there is  $L' \in \mathcal{R}'_l$  in which  $x < y$ . By Observation 6 we know that  $\mathcal{R}' = \mathcal{R}'_h \cup \mathcal{R}'_l$  is a realizer of  $\mathbf{P}'$ .  $\square$

#### 4. UP-GROWING CASE, UPPER BOUND: $w + 1$

Let  $\mathbf{P} = (X, \leq)$  be a semi-order. For  $x \in X$  define a partition of the set  $\text{Inc}(x)$  into  $\text{Inc}_0(x) \cup \text{Inc}_1(x)$  as follows:

$$\text{Inc}_0(x) = \min(\text{Inc}(x)), \quad \text{Inc}_1(x) = \text{Inc}(x) - \text{Inc}_0(x).$$

Obviously,  $\text{Inc}_0(x)$  is an antichain. Moreover, by Theorem 4 we know that  $\mathbf{P}$  is  $(\mathbf{3} + \mathbf{1})$ -free and hence  $\text{Inc}_1(x)$  is an antichain as well. Note also that for every  $y \in \text{Inc}_1(x)$  we have  $y \succ x$ .

Let  $\mathbf{P} = (X, \leq)$  be a semi-order of width at most  $w$  and  $\mathcal{R} = \{L_1, \dots, L_w\}$  be a realizer of  $\mathbf{P}$  of size  $w$ . Suppose  $\mathbf{P}$  is extended to  $\mathbf{P}' = (X \cup \{x\}, \leq)$  so that  $\text{width}(\mathbf{P}') \leq w$  and  $x$  is maximal in  $\mathbf{P}'$ . As in Section 3 we would like to know under which conditions for  $\mathcal{R}$  this new point  $x$  can be inserted into existing linear extensions from  $\mathcal{R}$  without creating a new,  $(w + 1)$ -st one, so that the new set of extensions  $\mathcal{R}'$  is a realizer of  $\mathbf{P}'$ . The next observation gives such condition.



**Observation 10.** *In the setting described above,  $\mathcal{R}'$  is a realizer of  $\mathbf{P}'$  whenever for every  $y \in \text{Inc}_0(x)$  there exists  $L'_i \in \mathcal{R}'$  in which  $x < y$  and there exists  $L'_j \in \mathcal{R}'$  where  $x$  is on the top.*

In the up-growing setting it is enough to maintain only a high-coloring function  $c : X \rightarrow \{1, \dots, w\}$  of  $(\mathbf{P}, \mathcal{R})$  to construct a realizer  $\mathcal{R}'$  of  $\mathbf{P}' = (X \cup \{x\}, \leq)$ . In the rest of this Chapter we simply say *coloring function* for a *high-coloring function*.

The following statement only slightly differs from Proposition 7. In particular, the new point  $x$  is put not only high, but actually on the top of  $L_{i_0}$ . Such an insertion of  $x$  is now possible as  $\mathbf{P}$  is an up-growing semi-order. This slight modification of Algorithm's behavior will guarantee that  $\mathcal{R}'$  becomes a realizer of  $\mathbf{P}'$ . However, we will sometimes no longer be able to maintain a coloring function for  $\mathbf{P}'$ .

**Proposition 11.** *In the setting defined above, fix one  $i_0 \in \{1, \dots, w\} - c(\text{Inc}_0(x))$  and define  $\mathcal{R}' = \{L'_1, \dots, L'_w\}$  as follows:*

- put  $x$  on the top of  $L_{i_0}$ , i.e.,  $L'_{i_0} := (L_{i_0})'_{\text{up}}$ ,
- put  $x$  down in  $L_i$ , for all  $i \neq i_0$ , i.e.,  $L'_i := (L_i)'_{\text{down}}$ .

*Then  $\mathcal{R}'$  is a realizer of  $\mathbf{P}'$ .*

*Proof.* From Observation 5(iii) we have  $x < y$  in  $L'_{c(y)}$  for  $y \in \text{Inc}_0(x)$ . On the other hand, we know that  $x$  is put on the top of  $L_{i_0}$ . This proves that  $\mathcal{R}'$  is a realizer of  $\mathbf{P}'$ , by Observation 10.  $\square$

The proof of Proposition 11 was based on the fact that every  $y \in X$  was high in at least one linear extension from  $\mathcal{R}$ , namely in  $L_{c(y)}$ . If one could show that there still exists a coloring function  $c'$  for  $(\mathbf{P}', \mathcal{R}')$  then, like in the proof of Theorem 9, a simple iteration of Proposition 11 would prove  $\text{val}(w) \leq w$  for the dimension of up-growing semi-orders. Unfortunately, Example 12 shows such  $c'$  need not exist if no additional care is taken for  $L_1, \dots, L_w$ .

**Example 12.** Let  $\mathbf{P} = (X, \leq)$  with  $X = (a, b, c, d, e, f)$  be an up-growing semi-order of width 3 shown by Figure 5. (To emphasize the on-line nature of  $\mathbf{P}$  we treat  $X$  as a sequence, rather than a set of points). If an on-line algorithm builds

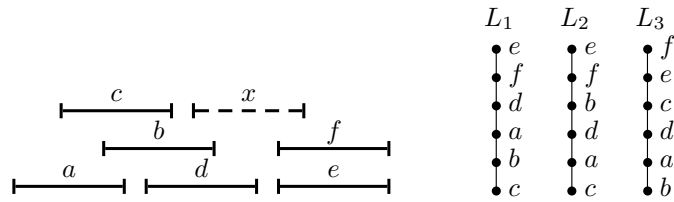


FIGURE 5. Order  $\mathbf{P}$  and linear extensions  $L_1, L_2, L_3$  from Example 12

a realizer  $\mathcal{R}$  of  $\mathbf{P}$  according to Proposition 11 then, depending on the choice of  $i_0$  at each step, the set  $\mathcal{R} = \{L_1, L_2, L_3\}$  may look just like in Figure 5. Point  $b$  is high only in  $L_2$ , point  $d$  is high only in  $L_1$ , point  $f$  is high only in  $L_3$ . If  $X$  is extended to  $X' = (a, b, c, d, e, f, x)$ , where  $x \parallel b, d, e, f$  and  $x > a, c$  then, according to Proposition 11,  $x$  is put down in  $L_1$  and  $L_2$ ,  $x$  is put on the top of  $L_3$ . In the new order  $\mathbf{P}' = (X', \leq)$  point  $f$  is not high in  $L'_3$  and the value  $c'(f)$  cannot be defined.

As already shown in Proposition 7, there is an easy way to maintain a coloring function when passing from  $(\mathbf{P}, \mathcal{R})$  to  $(\mathbf{P}', \mathcal{R}')$ . However, this can be achieved at the expense that  $\mathcal{R}'$  is no longer a realizer for  $\mathbf{P}$ .

Recall the poset  $\mathbf{P}$  from Example 12. If the new point  $x$  was put high in  $L_3$  then point  $f$  would remain high in  $L_3$ . However, the set  $\mathcal{R}' = \{L'_1, L'_2, L'_3\}$  would no more be a realizer of  $\mathbf{P}'$ , as in all of the  $L'_i$ 's we would have  $x < e, f$ .

At the cost of one extra linear extension, in which the new point  $x$  will be always put on the top, we can combine the techniques from Propositions 7 and 11 to get the next result.

**Theorem 13.** *Let  $\text{val}(w)$  denote the value of the on-line dimension game on the class of up-growing semi-orders of width at most  $w$ , presented without representation. Then*

$$\text{val}(w) \leq w + 1.$$

*Proof.* Let  $\mathbf{P} = (X, \leq)$  be an up-growing semi-order of width at most  $w$ , let  $\mathcal{R} = \{L_1, \dots, L_w, L_\star\}$  be a realizer of  $\mathbf{P}$  of size  $w + 1$ . For our convenience (and to remain consistent with the notation used in next sections), define  $\Gamma_0 = \{1, \dots, w\}$  and  $\Gamma = \Gamma_0 \cup \{\star\}$ . Assume there exists a coloring function  $c : X \rightarrow \Gamma_0$  for  $(\mathbf{P}, \mathcal{R} - \{L_\star\})$ . (Both  $\mathcal{R}$  and the function  $c$  are defined in an obvious way when  $|X| \leq 1$ ).

Consider an up-growing extension  $\mathbf{P}' = (X \cup \{x\}, \leq)$  of  $\mathbf{P}$  such that  $\text{width}(\mathbf{P}') \leq w$ . We present an algorithm which produces a realizer  $\mathcal{R}'$  of  $\mathbf{P}'$  of size  $w + 1$  and a coloring function  $c' : X' \rightarrow \Gamma_0$  of  $(\mathbf{P}', \mathcal{R}' - \{L'_\star\})$  (see Algorithm 1). A simple iteration of the algorithm proves the thesis.

---

**Algorithm 1:** Dimension of up-growing semi-orders:  $\text{val}(w) \leq w + 1$

---

- 1  $\gamma \leftarrow$  any element from  $\Gamma_0 - c(\text{Inc}_0(x))$
  - 2 put  $x$  *high* in  $L_\gamma$
  - 3 **foreach**  $j \in \Gamma_0 - \{\gamma\}$  **do** put  $x$  *down* in  $L_j$
  - 4 put  $x$  *on the top* of  $L_\star$
  - 5  $c'(x) \leftarrow \gamma$
  - 6 **foreach**  $y \in X$  **do**  $c'(y) \leftarrow c(y)$
- 

We show that Algorithm 1 is correct. Recalling the proof of Proposition 7 it is easy to see that  $c'$  is indeed a coloring function of  $(\mathbf{P}', \mathcal{R}' - \{L'_\star\})$ . From the proof of Proposition 11, with  $\{L_1, \dots, L_w\}$  replaced by  $(\{L_1, \dots, L_w\} - \{L_\gamma\}) \cup \{L_\star\}$ , it follows that  $\mathcal{R}'$  is a realizer of  $\mathbf{P}'$ .  $\square$

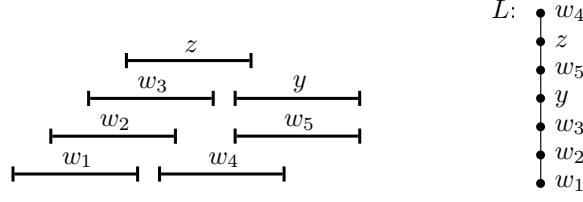
## 5. UP-GROWING CASE, UPPER BOUND: $w$

In the proof of Theorem 13 we made an extensive use of the additional  $(w + 1)$ -st linear extension  $L_\star$  to put the new maximal point  $x$  above every existing element of  $X$ . Then, at most  $w - 1$  linear extensions were needed to put  $x$  below  $\text{Inc}_0(x)$ , the one remaining extension was used to put  $x$  high. Example 12 shows that following the same pattern with only  $w$  linear extensions is not always possible. To stay within  $w$  linear extensions we need to refine some of our notions.

Let  $\mathbf{P} = (X, \leq)$  be a semi-order and  $L$  be a linear extension of  $\mathbf{P}$ . For two elements  $y, z \in X$  such that  $y \not\leq z$  we say that  $y$  is *high* in  $L$  with respect to  $z$  if  $w > y$  in  $L$  implies  $w > z$  in  $\mathbf{P}$ , for every  $w \in X - \{z\}$  (see Figure 6).

We will see that the concept of being high with respect to another element is the needed refinement of the notion of being high. Indeed,  $y$  is high in  $L$  if and only if  $y$  is high in  $L$  with respect to  $y$ . Furthermore, Observation 5 can be generalized as follows.

**Observation 14.** *Let  $\mathbf{P} = (X, \leq)$  be a semi-order and  $L$  be a linear extension of  $\mathbf{P}$ . For two incomparable points  $y, z \in X$ , assume that  $y$  is high in  $L$  with respect to*


 FIGURE 6.  $y$  is not high in  $L$ ,  $y$  is high in  $L$  with respect to  $z$ 

$z$ . If  $\mathbf{P}$  is extended to  $\mathbf{P}' = (X \cup \{x\}, \leq)$  and the linear extension  $L'_{\text{down}}$  is defined as in Section 4 then

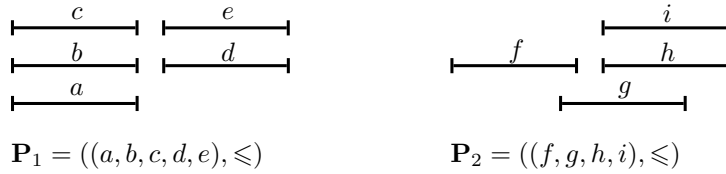
- (i) if  $z \prec x$  then  $y$  is high with respect to  $z$  in any  $L'$  obtained from  $L$ ,
- (ii) if  $x \parallel y$  and  $x \parallel z$  then  $x < y$  in  $L'_{\text{down}}$ .

*Proof.* Condition  $z \prec x$  guarantees that  $x$  will not destroy the property of  $y$  being high with respect to  $z$  in any extension  $L'$  obtained from  $L$ , exactly as (i) states.

Now let  $d$  be the point preceding  $x$  in  $L'_{\text{down}}$ . Then  $d < x$ . Together with the assumption  $x \parallel y$  we get  $d \neq y$ . Moreover, if  $x \parallel z$  then  $d \prec z$  as otherwise  $z \preceq d < x$  would give  $z < x$ . Point  $y$  is high in  $L$  with respect to  $z$ , hence  $d < y$  in  $L$ . Since  $x$  is just above  $d$  in  $L'_{\text{down}}$ , this implies  $x < y$  in  $L'_{\text{down}}$  and proves (ii).  $\square$

Let  $\mathbf{P} = (X, \leq)$  be an up-growing semi-order of width  $w$ . We say that  $y \in X$  is *active* in  $\mathbf{P}$  if there is an extension  $\mathbf{P}'$  of  $\mathbf{P}$  such that  $X' = X \cup \{x\}$ ,  $\text{width}(\mathbf{P}') = w$ ,  $x \parallel y$  and  $x$  is maximal in  $\mathbf{P}'$ . In other words, it is possible to add to  $\mathbf{P}$  a maximal element incomparable with  $y$  without increasing the width of  $\mathbf{P}$ . The set of active points in  $\mathbf{P}$  is denoted by  $\text{act}(\mathbf{P})$ .

**Example 15.** Let up-growing semi-orders  $\mathbf{P}_1$  and  $\mathbf{P}_2$  of width 3 be defined by correlating interval representations as in Figure 7. In poset  $\mathbf{P}_1$  all five points are active, in poset  $\mathbf{P}_2$  the only active points are  $h$  and  $i$ . (Recall, that in the currently considered on-line dimension game the interval representation of the poset is not revealed to Algorithm. Posets  $\mathbf{P}_1$  and  $\mathbf{P}_2$  are presented as intervals since it is more intuitive to think about interval orders in the terms of their geometry).


 FIGURE 7. Up-growing semi-orders  $\mathbf{P}_1$  and  $\mathbf{P}_2$ 

The concept of active points turns out to be extremely handy on the class of up-growing semi-orders. The intuition is that they are the only points an on-line algorithm needs to keep track of. As we shall see below, their number is also very limited.

**Observation 16.** Let  $\mathbf{P} = (X, \leq)$  be a semi-order of width  $w$ . If  $\mathbf{P}' = (X \cup \{x\}, \leq)$  is an up-growing extension of  $\mathbf{P}$  such that  $\text{width}(\mathbf{P}') = w$ , then

- (i) if  $y \notin \text{act}(\mathbf{P})$  then  $x > y$  and  $y \notin \text{act}(\mathbf{P}')$ ,
- (ii)  $\text{Inc}(x) \subseteq \text{act}(\mathbf{P})$ ,
- (iii) if  $y \in \text{act}(\mathbf{P})$  and  $z \succ y$  then  $z \in \text{act}(\mathbf{P})$ ,
- (iv) if  $y \in \text{act}(\mathbf{P})$  then  $y \uparrow$  is an antichain.

*Proof.* Properties (i) and (ii) follow directly from the definition of being an active point. To see that (iii) holds observe that  $x \parallel y$  implies  $x \parallel z$ , for any maximal element  $x$  in  $\mathbf{P}'$ . Finally, suppose that (iv) fails, i.e., there exist in  $\mathbf{P}$  three active points  $y, z$  and  $w$  with  $y < z < w$ . Hence  $\mathbf{P}$  can be extended to  $\mathbf{P}' = (X \cup \{x\}, \leq)$  so that  $x \parallel y$ . Since  $x$  is maximal in  $\mathbf{P}'$ , we get  $x \parallel z, w$ . This produces a  $(\mathbf{3} + \mathbf{1})$  configuration  $y, z, w \parallel x$  forbidden in semi-orders.  $\square$

Let  $c : X \rightarrow \{1, \dots, w\}$  be a coloring function of a semi-order  $\mathbf{P} = (X, \leq)$  with a realizer  $\mathcal{R} = \{L_1, \dots, L_w\}$ . Observe that for two points  $y$  and  $z$  with  $c(y) = c(z) = \gamma$  we have  $y < z$  or  $y > z$  as both  $y$  and  $z$  are high  $L_\gamma$ . Thus, for  $\gamma \in c(X)$  we may define  $\text{top}(\gamma)$  to be the highest (with respect to  $\leq$ ) element in  $\mathbf{P}$  colored with  $\gamma$ . Function  $\text{top} : c(X) \rightarrow X$  induces a linear quasi-ordering of colors actually used by  $c$ . For  $\gamma, \delta \in c(X)$  we say that  $\gamma$  is a (weakly) higher color than  $\delta$  and write  $\gamma \sqsupseteq \delta$  if  $\text{top}(\gamma) \succcurlyeq \text{top}(\delta)$ .

Recall the poset  $\mathbf{P}$  and linear extensions  $L_1, L_2, L_3$  from Example 12. If a coloring function for  $(\mathbf{P}, \{L_1, L_2, L_3\})$  is defined as follows:  $c(\{a, d, e\}) = 1$ ,  $c(b) = 2$ ,  $c(\{c, f\}) = 3$ , then  $\text{top}(1) = e$ ,  $\text{top}(2) = b$ ,  $\text{top}(3) = f$  and hence  $2 \sqsubset 1, 3$ .

We describe our algorithm in three steps. First, we introduce a data structure used by the algorithm. Second, we define a set of invariants which are to be kept during each run of the algorithm. Finally, we present a pseudo-code of the algorithm.

*Data structure.* We introduce variables similar to those used in Algorithm 1.

$\mathbf{P} = (X, \leq)$	an up-growing semi-order of width $w$
$\mathcal{R}$	a set of $w$ linear extensions of $\mathbf{P}$ , $\mathcal{R} = \{L_1, \dots, L_w\}$
$x$	a new (maximal) element extending $\mathbf{P}$ to $\mathbf{P}' = (X \cup \{x\}, \leq)$
$\Gamma = \{1, \dots, w\}$	the set of numbers of $w$ linear extensions in $\mathcal{R}$
$\star \in \Gamma$	the number indicating a <i>special</i> linear extension $L_\star \in \mathcal{R}$
$\Gamma_0 = \Gamma - \{\star\}$	the set of numbers of $w - 1$ non-special linear extensions in $\mathcal{R}$
$c : X \rightarrow \Gamma$	a (high) coloring function for $(\mathbf{P}, \mathcal{R})$ , i.e., $y$ high in $L_{c(y)}$

Note that  $\Gamma$  is now, in contrary to Algorithm 1, of size  $w$  (not  $w + 1$ ) and the value of  $\star$  may be changed by each run of the algorithm. Note also that the coloring  $c : X \rightarrow \Gamma$  enforces an additional structure on  $\Gamma$ , namely the quasi-ordering  $\sqsubseteq$ . These facts will help to substantially improve the old algorithm.

Following our notation from previous sections we distinguish input and output variables of the algorithm by appending  $'$  to the output variables, i.e.,  $\mathbf{P}$  becomes  $\mathbf{P}'$ ,  $L_2$  becomes  $L_2'$ ,  $\Gamma$  becomes  $\Gamma'$ ,  $\star$  becomes  $\star'$  etc.

*Invariants.*

- (I<sub>0</sub>)  $\mathcal{R}$  is a realizer of  $\mathbf{P}$  and  $c : X \rightarrow \Gamma$  is a coloring function for  $(\mathbf{P}, \mathcal{R})$
- (I<sub>1</sub>) among points colored by  $\star$  there is at most one active point in  $\mathbf{P}$ , denoted by  $\mathbf{k}$  and called the *anchor*. In other words, for every  $y \in \text{act}(\mathbf{P})$ , if  $y \neq \mathbf{k}$  then  $c(y) \in \Gamma_0$

The conditions (I<sub>0</sub>) and (I<sub>1</sub>) are the only invariants if the anchor does not exist. However, if the anchor does exist, we require four more invariants to be kept:

- (I<sub>2</sub>)  $|\text{Inc}_0(\mathbf{k})| = w - 1$
- (I<sub>3</sub>) if  $z \in \text{Inc}_0(\mathbf{k})$  then  $c(z) \in \Gamma_0$  and  $\mathbf{k}$  is high in  $L_{c(z)}$  with respect to  $z$

Define two antichains  $A$  and  $B$  by:  $A = \text{Inc}_1(\mathbf{k})$ ,  $B = \mathbf{k}\uparrow$ . Note that  $A$  is an antichain, since  $\mathbf{P}$  is  $(\mathbf{3} + \mathbf{1})$ -free. Moreover,  $B$  is an antichain by Observation 16(iv) applied to the active point  $\mathbf{k}$ . The behavior of  $A$  and  $B$  is further restricted by the invariants:

- (I<sub>4</sub>)  $|c(A \cup B)| \leq \text{width}(A \cup B)$
- (I<sub>5</sub>) if  $\text{width}(A) < \text{width}(A \cup B) = |c(A \cup B)|$  then there exists  $\mathbf{b} \in B$  such that

- (i)  $c(\mathbf{b}) \notin c(A \cup B - \{\mathbf{b}\})$
- (ii)  $\mathbf{b}$  is high in  $L_\star$
- (iii) there is an antichain  $I \subseteq A \cup B$  such that  $\mathbf{b} \in I$  and  $|I| = \text{width}(A \cup B)$

Although our algorithm works also for  $w = 1$ , there is obviously a much simpler one if the poset produced by Spoiler is a chain. Therefore, we may assume that  $w > 1$ . Now we easily check that invariants  $(I_0)$ – $(I_5)$  are satisfied by any one-element poset  $\mathbf{P}$ , its realizer  $\mathcal{R}$  and any coloring that does not use  $\star$ . This proves the base of the induction step.

*Algorithm.* See the following page.

---

**Algorithm 2:** Dimension of up-growing semi-orders:  $\text{val}(w) \leq w$ 


---

```

1   $\star' \leftarrow \star$ 
2  foreach  $y \in X$  do  $c'(y) \leftarrow c(y)$ 
3  if  $c(\text{Inc}_0(x)) \subsetneq \Gamma_0$  then /* Case 1 */
4     $\gamma \leftarrow$  a maximal element, with respect to  $\sqsubseteq$ , in  $\Gamma_0 - c(\text{Inc}_0(x))$ 
5    put  $x$  high in  $L_\gamma$ 
6    put  $x$  on the top of  $L_\star$ 
7    foreach  $j \in \Gamma_0 - \{\gamma\}$  do put  $x$  down in  $L_j$ 
8     $c'(x) \leftarrow \gamma$ 
9  else if  $c(\text{Inc}_0(x)) = \Gamma_0$  then /* Case 2 */
10   put  $x$  on the top of  $L_\star$ 
11   foreach  $y \in \text{Inc}_0(x)$  do put  $x$  right below  $y$  in  $L_{c(y)}$ 
12    $c'(x) \leftarrow \star$ 
13 else if  $\star \in c(\text{Inc}_0(x))$  then /* Case 3 */
14   let  $\mathbf{k}$  denote the anchor defined in  $(I_1)$ 
15   if  $x \parallel m$  for some  $m \in \text{Inc}_0(\mathbf{k})$  then /* Case 3.1 */
16      $\gamma \leftarrow$  any element from  $\Gamma_0 - c(\text{Inc}_0(x))$ 
17      $c'(x) \leftarrow \gamma$ 
18     if  $\gamma \in c(A \cup B)$  then /* Case 3.1.1 */
19       put  $x$  high in  $L_\gamma$ 
20       put  $x$  on the top of  $L_\star$ 
21       foreach  $j \in \Gamma_0 - \{\gamma\}$  do put  $x$  down in  $L_j$ 
22     else if  $\gamma \notin c(A \cup B)$  then /* Case 3.1.2 */
23       put  $x$  on the top of  $L_\gamma$ 
24       foreach  $j \in \Gamma - \{\gamma\}$  do put  $x$  down in  $L_j$ 
25     else if  $(\text{Inc}_0(\mathbf{k}) < x)$  and  $(c(A \cup B) \subsetneq \Gamma_0)$  then /* Case 3.2 */
26        $\gamma \leftarrow$  any element from  $\Gamma_0 - c(A \cup B)$ 
27        $c'(x) \leftarrow \gamma$ 
28       put  $x$  on the top of  $L_\gamma$ 
29       foreach  $j \in \Gamma - \{\gamma\}$  do put  $x$  down in  $L_j$ 
30     else if  $(\text{Inc}_0(\mathbf{k}) < x)$  and  $(c(A \cup B) = \Gamma_0)$  then /* Case 3.3 */
31       choose  $\mathbf{b}$ , an active point witnessing  $(I_5)$ 
32        $\star' \leftarrow c(\mathbf{b})$ ,  $c'(\mathbf{b}) \leftarrow \star$ 
33     if  $c'(\text{Inc}_0(x)) \subsetneq \Gamma'_0$  then /* Case 3.3.1 */
34        $\gamma \leftarrow$  any element from  $\Gamma'_0 - c'(\text{Inc}_0(x))$ 
35       put  $x$  high in  $L_\gamma$ 
36       put  $x$  on the top of  $L_{\star'}$ 
37       foreach  $j \in \Gamma'_0 - \{\gamma\}$  do put  $x$  down in  $L_j$ 
38        $c'(x) \leftarrow \gamma$ 
39     else if  $c'(\text{Inc}_0(x)) = \Gamma'_0$  then /* Case 3.3.2 */
40       put  $x$  on the top of  $L_{\star'}$ 
41       foreach  $y \in \text{Inc}_0(x)$  do put  $x$  right below  $y$  in  $L_{c(y)}$ 
42        $c'(x) \leftarrow \star'$ 

```

---

As it can be easily seen our Algorithm proceeds the incoming point  $x$  depending on the mutual behavior of  $\text{Inc}_0(x)$  and  $\text{Inc}_0(\mathbf{k})$ . This produces 7 cases. Before considering each case separately and proving that invariants (I<sub>0</sub>)–(I<sub>5</sub>) are satisfied by the output of the algorithm, we collect a few useful facts in the observation below.

**Observation 17.** *Assume that  $\text{width}(\mathbf{P}) = w$  and the anchor  $\mathbf{k}$  defined in (I<sub>1</sub>) exists in  $\mathbf{P}$ . Let  $x$  be a maximal element which extends  $\mathbf{P}$  to  $\mathbf{P}'$  so that  $\text{width}(\mathbf{P}') = w$ . Then*

- (i) *if  $Y$  is an antichain of size  $w$  in  $\mathbf{P}$  then  $y \prec x$  in  $\mathbf{P}'$  for all  $y \in Y$ ,*
- (ii)  *$\mathbf{k} \prec x$ ,*
- (iii)  *$\text{Inc}_0(\mathbf{k}) \prec x$ ,*
- (iv)  *$\text{Inc}(x) \subseteq \text{Inc}(\mathbf{k}) \cup \{\mathbf{k}\} \cup \mathbf{k}\uparrow$ ,*
- (v) *if  $c(\text{Inc}_0(x)) \subseteq \Gamma_0$  then  $\mathbf{k} < x$ .*

*Proof.* Suppose that (i) fails and  $x \preceq y$  for some  $y \in Y$ . Then  $x \parallel Y$  (see Figure 8). The set  $Y \cup \{x\}$  would be a  $(w + 1)$ -element antichain in  $\mathbf{P}'$ , a contradiction. By

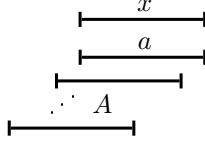


FIGURE 8.  $x \preceq y \in Y$  implies  $x \parallel Y$

(I<sub>2</sub>) we have  $|\text{Inc}_0(\mathbf{k})| = w - 1$ , so the set  $\{\mathbf{k}\} \cup \text{Inc}_0(\mathbf{k})$  is an antichain of size  $w$  in  $\mathbf{P}$ . Now, (ii) and (iii) follow from (i).

For  $z \parallel x$ , by (ii) we have  $z \not\prec \mathbf{k} \prec x$ , which gives (iv). Finally, to prove (v), assume that  $c(\text{Inc}_0(x)) \subseteq \Gamma_0$ , i.e.,  $\mathbf{k} \notin \text{Inc}_0(x)$ . Then  $\mathbf{k} \prec x \prec \text{Inc}_1(x)$  gives  $\mathbf{k} \notin \text{Inc}_1(x)$ , so indeed,  $\mathbf{k} \notin \text{Inc}(x) = \text{Inc}_0(x) \cup \text{Inc}_1(x)$ .  $\square$

We are ready to prove that each run of Algorithm 2 keeps invariants (I<sub>0</sub>)–(I<sub>5</sub>).

*Case 1.* In this case there are spare colors for  $x$  in the set  $\Gamma_0 - c(\text{Inc}_0(x))$ . Therefore, Algorithm behaves pretty much as Algorithm 1, but we will need some effort to prove that it still keeps our involving invariants. Note that if the anchor  $\mathbf{k}$  existed in  $\mathbf{P}$  then assumption  $c(\text{Inc}_0(x)) \subseteq \Gamma_0$  applied to Observation 17(v) gives

$$\mathbf{k} < x. \quad (1)$$

Furthermore, the color  $\gamma$  in line 4 is chosen so that  $\text{top}(\gamma) = \max_{\preceq} \{\text{top}(\delta) : \delta \in \Gamma_0 - c(\text{Inc}_0(x))\}$ . Now we analyze the invariants.

(I<sub>0</sub>) By Observation 5(iii) we get  $x < y$  in  $L'_{c(y)}$  for  $y \in \text{Inc}_0(x)$ . On the other hand,  $x$  is put on the top of  $L_\gamma$ . This proves that  $\mathcal{R}'$  is a realizer of  $\mathbf{P}'$ , by Observation 10.

To prove that  $c'$  is a high-coloring function for  $(\mathbf{P}', \mathcal{R}')$  we show that  $y$  is high in  $L'_{c'(y)}$  for every  $y \in X \cup \{x\}$ . Clearly,  $x$  is high in  $L'_\gamma$ . Fix  $y \in X$ . First, assume that  $c(y) = \star$ . If  $y \in \text{act}(\mathbf{P})$  then (I<sub>1</sub>) gives  $y = \mathbf{k}$  and together with (1) we get  $y < x$ . If  $y \notin \text{act}(\mathbf{P})$  then  $y < x$  as well, by Observation 16(i). In both cases,  $y$  remains high in  $L'_\star$ . Now, let  $c(y) \neq \star$  and  $y \in \text{Inc}_0(x)$ . Point  $x$  is put down in  $L_{c(y)}$  and hence  $y$  remains high in  $L'_{c'(y)}$ , by Observation 5(ii). Finally, let  $c(y) \neq \star$  and  $y \in X - \text{Inc}_0(x)$ . Then  $x$  and  $y$  are not twins. Again, from Observation 5(i) it follows that  $y$  is high in  $L'_{c'(y)}$ .

(I<sub>1</sub>) By Observation 16(i) we have  $\text{act}(\mathbf{P}') \subseteq \text{act}(\mathbf{P}) \cup \{x\}$ , but the value of  $c'(x)$  is defined so that  $c'(x) \neq \star$ .

Invariants (I<sub>2</sub>)–(I<sub>5</sub>) assume that in the poset  $\mathbf{P}'$  there is an anchor, i.e.,  $\mathbf{k}' \in \text{act}(\mathbf{P}')$  with  $c'(\mathbf{k}') = \star'$ . Note that the anchor  $\mathbf{k}'$  may exist in  $\mathbf{P}'$  only if  $\mathbf{P}$  had the anchor  $\mathbf{k}$  which remains active in  $\mathbf{P}'$ . Hence  $\mathbf{k}' = \mathbf{k}$ .

(I<sub>2</sub>) From (1) it clearly follows that the set  $\text{Inc}_0(\mathbf{k})$  does not change in  $\mathbf{P}'$ .

(I<sub>3</sub>) By Observation 17(iii), for  $z \in \text{Inc}_0(\mathbf{k})$  we have  $z \prec x$ . Now we make use of Observation 14(i).

(I<sub>4</sub>) From (1) we have  $x > \mathbf{k} = \mathbf{k}'$ . Now, Observation 16(iv) applied to  $\mathbf{k}$  gives  $x \parallel B$ , so  $A' = A$  and  $B' = B \cup \{x\}$ . Obviously, (I<sub>4</sub>) holds if  $\text{width}(A \cup B) < \text{width}(A \cup B')$  or  $|c(A \cup B)| < \text{width}(A \cup B)$ . Otherwise,

$$\text{width}(A \cup B') = \text{width}(A \cup B) = |c(A \cup B)| \quad (2)$$

and the first equality means in particular that  $x$  must be comparable with some point from  $A \cup B$ . This combined with  $x \parallel B$  yields a point  $a \in A$  with  $a < x$ . However,  $\text{Inc}_0(\mathbf{k}) \prec \text{Inc}_1(\mathbf{k}) = A \ni a < x$  so that  $\text{Inc}_0(\mathbf{k}) < x$ . Hence  $\text{Inc}_0(\mathbf{k}) \cap \text{Inc}(x) = \emptyset$ , which together with Observation 17(iv) gives  $\text{Inc}_0(x) \subseteq \text{Inc}_1(\mathbf{k}) \cup \mathbf{k} \uparrow = A \cup B$  (see Figure 9). Now, note that  $\{x\} \cup \text{Inc}_0(x)$  is an antichain in  $A \cup B'$  and so  $|\text{Inc}_0(x)| + 1 \leq \text{width}(A \cup B')$ . By (2) we obtain  $c(A \cup B) - c(\text{Inc}_0(x)) \neq \emptyset$ .

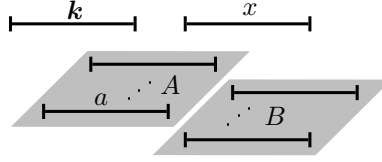


FIGURE 9. Algorithm 2—Case 1,  $\text{Inc}_0(x) \subseteq A \cup B$

According to line 4, point  $x$  gets a maximal available color with respect to  $\sqsubseteq$ . Observe that colors used on points from  $A \cup B$  are  $\sqsubseteq$ -greater than the remaining ones. Thus,  $x$  gets a color from a non-empty set  $c(A \cup B) - c(\text{Inc}_0(x))$ . This proves that  $|c'(A \cup B')| = \text{width}(A \cup B')$ .

(I<sub>5</sub>) As above, we have  $A' = A$ ,  $B' = B \cup \{x\}$  and  $x \parallel B$ . The assumptions of (I<sub>5</sub>) for  $\mathbf{P}'$  are:  $\text{width}(A) < \text{width}(A \cup B')$  and  $|c'(A \cup B')| = \text{width}(A \cup B')$ .

If  $\text{width}(A \cup B) < \text{width}(A \cup B')$  then the assumption  $|c'(A \cup B')| = \text{width}(A \cup B')$  implies  $c'(x) \notin c(A \cup B)$ , as  $|c(A \cup B)| = |c'(A \cup B')| = \text{width}(A \cup B') > \text{width}(A \cup B)$  contradicts (I<sub>4</sub>) for  $\mathbf{P}$ . This shows that  $\mathbf{b}' := x$  satisfies (I<sub>5</sub>-i), while the other conditions saying that  $x$  becomes a valid witness for  $\mathbf{b}'$  in  $\mathbf{P}'$  are obvious.

If  $\text{width}(A \cup B) = \text{width}(A \cup B')$  then as in (I<sub>4</sub>) we argue that  $\text{Inc}_0(x) \subseteq A \cup B$ . First, consider the case  $c'(x) \notin c(A \cup B)$ . Since Algorithm has to use for  $x$  colors from  $c(A \cup B)$  whenever possible (as they are greater than other colors in the  $\sqsubseteq$ -ordering), we deduce that  $c(A \cup B) \subseteq c(\text{Inc}_0(x))$  and therefore

$$|\text{Inc}_0(x)| = |c(A \cup B)| = |c(A \cup B')| - 1 = \text{width}(A \cup B') - 1.$$

This proves that  $x \cup \text{Inc}_0(x)$  is an antichain of size  $\text{width}(A \cup B')$  in  $A \cup B'$ . Thus  $\mathbf{b}' := x$  satisfies (I<sub>5</sub>-iii). The other conditions saying that  $x$  becomes a valid witness for  $\mathbf{b}'$  in  $\mathbf{P}'$  are obvious. Finally, consider the case  $c'(x) \in c(A \cup B)$ . Then  $|c(A \cup B)| = |c'(A \cup B')| = \text{width}(A \cup B') = \text{width}(A \cup B)$  and we also know that  $\text{width}(A) < \text{width}(A \cup B') = \text{width}(A \cup B)$ . Hence  $\mathbf{b}$  witnessing (I<sub>5</sub>) has already existed in  $\mathbf{P}$ . Condition (I<sub>5</sub>-iii) supplies us with an antichain  $I \subseteq A \cup B$  of size  $\text{width}(A \cup B) = \text{width}(A \cup B')$ . Therefore  $\mathbf{b} \prec x$ , as otherwise  $I \cup \{x\}$  would be an antichain of size  $\text{width}(A \cup B) + 1$ , fully contained in  $A \cup B'$ . Thus,  $\mathbf{b}$  remains high in  $L'_\star$ , by Observation 5(i). All we need to show that  $\mathbf{b}$  remains a valid witness of (I<sub>5</sub>) in  $\mathbf{P}'$  is that the color of  $\mathbf{b}$  is still unique in  $A \cup B \cup \{x\}$ , i.e.,  $c'(x) \neq c'(\mathbf{b})$ .



However,  $x \parallel \mathbf{b}$  and  $x \succ \mathbf{b}$  gives  $\mathbf{b} \in \text{Inc}_0(x)$ , while the color for  $x$  was chosen to be outside  $\text{Inc}_0(x)$ .

*Case 2.* In this case there is no spare color for  $x$  in  $\Gamma_0$  (see Figure 10). We will see that point  $x$  is the only candidate to be the anchor, as long as it would be active in  $\mathbf{P}'$ . Assumption  $c(\text{Inc}_0(x)) = \Gamma_0$  gives  $w - 1 \geq |\text{Inc}_0(x)| \geq |c(\text{Inc}_0(x))| = |\Gamma_0| = w - 1$ , so that we actually have

$$|\text{Inc}_0(x)| = w - 1. \quad (3)$$

This shows that  $c(y) \neq c(z)$  for two different points  $y, z \in \text{Inc}_0(x)$  and proves that line 11 can be properly executed. Also, if anchor  $\mathbf{k}$  existed in  $\mathbf{P}$  then (as in Case 1) we have

$$\mathbf{k} < x. \quad (4)$$

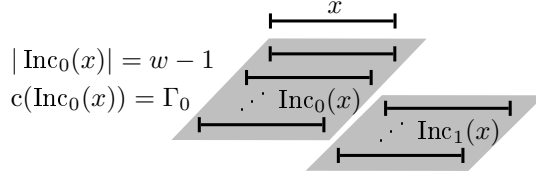


FIGURE 10. Algorithm 2—Case 2

(I<sub>0</sub>) The proof of the fact that  $\mathcal{R}'$  is a realizer of  $\mathbf{P}'$  is the same as in Case 1. It remains to show that  $c'$  is a coloring function for  $(\mathbf{P}', \mathcal{R}')$ . Clearly,  $x$  is high in  $L'_*$ . Fix  $y \in X$ . If  $c(y) = \star$  then, as in Case 1, we obtain that  $y$  remains high in  $L'_*$ . Now assume that  $c(y) = \gamma \in \Gamma_0$ . Since  $c(\text{Inc}_0(x)) = \Gamma_0$ , there is  $z \in \text{Inc}_0(x)$  with  $c(z) = \gamma$ . Note that  $z \preccurlyeq x$  as  $z \in \text{Inc}_0(x)$  and  $x$  is maximal in  $\mathbf{P}'$ . If  $y \prec z$  then  $y \prec z \preccurlyeq x$  gives  $y \prec x$ , so  $x$  cannot destroy the property of  $y$  being high in  $L'_\gamma$ . Otherwise,  $y \succcurlyeq z$ . (Actually,  $y \succ z$  as  $y \approx z$  contradicts the fact that both  $y$  and  $z$  are high in the same extension  $L_\gamma$ ). Since  $y$  is high in  $L_\gamma$ , we know that  $y > z$  in  $L_\gamma$ . However,  $x$  is put (right) below  $z$  in  $L'_\gamma$  and therefore  $y > x$  in  $L'_\gamma$  as well. This proves that  $y$  is high in  $L'_\gamma$ .

(I<sub>1</sub>) Since  $c'(x) = \star$ , to prove that (I<sub>1</sub>) holds in  $\mathbf{P}'$  we need to show that the possible old anchor  $\mathbf{k}$  is no longer active in  $\mathbf{P}'$ . From (3) we know that  $x \cup \text{Inc}_0(x)$  is an antichain of size  $w$  in  $\mathbf{P}'$ . Now, Observation 17(i) guarantees that any maximal point  $x'$  extending  $\mathbf{P}'$  to  $\mathbf{P}'' = (X' \cup \{x'\})$  satisfies  $x \prec x'$ . By (4),  $\mathbf{k} < x \prec x'$  and hence  $\mathbf{k} < x'$ , so indeed,  $\mathbf{k}$  is no longer active in  $\mathbf{P}'$ .

The rest of invariants are to be shown only if  $x \in \text{act}(\mathbf{P}')$ , i.e.,  $x$  is actually an anchor  $\mathbf{k}'$ .

(I<sub>2</sub>) is already obtained by (3).

(I<sub>3</sub>) Fix  $z \in \text{Inc}_0(x)$  and let  $\gamma = c(z)$ . According to line 11, we know that  $x$  is put right below  $z$  in  $L'_\gamma$ . Since  $z$  was high in  $L_\gamma$  we know that  $z < w$  in  $L_\gamma$  implies  $z \prec w$  for all  $w$ , and consequently,  $x < w$  in  $L'_\gamma$  implies  $z \prec w$  for all  $w \neq z$ .

(I<sub>4</sub>)–(I<sub>5</sub>) Since  $x = \mathbf{k}'$  is maximal in  $\mathbf{P}'$ , we have  $B' = \emptyset$  and (I<sub>4</sub>)–(I<sub>5</sub>) are trivially fulfilled.

*Case 3.* In this case we have  $\star \in c(\text{Inc}_0(x))$ . First of all we need to show that line 14 makes sense. As, by Observation 16(ii), we have  $\text{Inc}_0(x) \subseteq \text{Inc}(x) \subseteq \text{act}(\mathbf{P})$ , the color  $\star$  was used on some active point of  $\mathbf{P}$  (namely, the anchor), and this anchor  $\mathbf{k}$  is incomparable with  $x$ . The ideas from Algorithm 1 or Cases 1 and 2 do not work any longer. Indeed, putting  $x$  on the top of  $L_*$ , putting  $x$  down in  $L_j$  for

$j \in c(\text{Inc}_0(x))$  would require  $x$  to be on the top of  $L_\star$  and, at the same time, to be down in  $L_\star$ .

*Case 3.1.* In Case 3.1 we have not only  $x \parallel \mathbf{k}$  but additionally  $x \parallel m$ . More precisely, by Observation 17(ii–iii) we deduce that

$$\mathbf{k}, m \in \text{Inc}_0(x). \quad (5)$$

This gives  $|c(\text{Inc}_0(x)) - \{\star\}| \leq w - 2 < |\Gamma_0|$  and proves that line 16 makes sense. Note that although  $m \in \text{Inc}_0(\mathbf{k})$  it may happen that  $\mathbf{k} \prec m$  in  $\mathbf{P}$  (see Figure 11). A careful reader might ask for a reason to distinguish subcases 3.1.1 and 3.1.2.

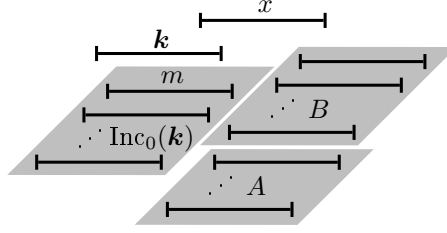


FIGURE 11. Algorithm 2—Case 3.1

Indeed, the difference between the two subcases is very subtle. It all boils down to the proofs of (I<sub>0</sub>) and (I<sub>5</sub>).

*Case 3.1.1.* (I<sub>0</sub>) Both  $c'$  and  $\mathcal{R}'$  are defined nearly like in Case 1. The one big difference is that  $x$  is not put down in  $L_{c(\mathbf{k})}$  although  $\mathbf{k} \in \text{Inc}_0(x)$ . We need an extra argument to show that

$$x < \mathbf{k} \text{ in some } L' \in \mathcal{R}'. \quad (6)$$

To see that  $x$  is put down in  $L_{c(m)}$  we need to know that  $c(m) \neq \gamma, \star$ . Obviously,  $c(m) \neq \gamma$  as  $\gamma \notin c(\text{Inc}_0(x))$  and  $m \in \text{Inc}_0(x)$  (see (5)). Also  $c(m) \neq \star$ , as by (I<sub>1</sub>) the only active point colored by  $\star$  is the anchor  $\mathbf{k}$ . Hence  $c(m) \neq \gamma, \star$  and according to line 21 point  $x$  is put down in  $L_{c(m)}$ . Applying (I<sub>3</sub>) to  $m \in \text{Inc}_0(\mathbf{k})$  we also get that  $\mathbf{k}$  is high in  $L_{c(m)}$  with respect to  $m$ . Now, (5) together with Observation 14(ii) give  $x < \mathbf{k}$  in  $L'_{c(m)}$ , just as required in (6).

(I<sub>1</sub>) See Case 1.

(I<sub>2</sub>) The set  $\text{Inc}_0(\mathbf{k})$  does not change in  $\mathbf{P}'$ . Otherwise, if  $x \in \text{Inc}_0'(\mathbf{k})$ , the set  $\{\mathbf{k}\} \cup \text{Inc}_0'(\mathbf{k})$  would be a  $(w + 1)$ -element antichain in  $\mathbf{P}'$ .

(I<sub>3</sub>) See Case 1.

(I<sub>4</sub>) Since  $x$  is maximal in  $\mathbf{P}'$ ,  $x \parallel \mathbf{k}$  and  $\mathbf{k} \prec \text{Inc}_1(\mathbf{k}) = A$  implies  $x \parallel A$ . Clearly,  $A' = A \cup \{x\}$  and  $B' = B$ . Moreover,  $x \parallel B$  as otherwise  $x > b \in B > \mathbf{k}$  would contradict  $x \parallel \mathbf{k}$ . Now,  $\text{width}(A' \cup B) > \text{width}(A \cup B) \geq |c(A \cup B)|$  so trivially  $\text{width}(A' \cup B) \geq |c(A \cup B)| + 1 \geq |c'(A' \cup B)|$ .

(I<sub>5</sub>) Assumption  $c'(x) \in c(A \cup B)$  from line 18 gives  $|c'(A' \cup B)| = |c(A \cup B)| \leq \text{width}(A \cup B) < \text{width}(A' \cup B)$  and the second assumption of (I<sub>5</sub>) fails in  $\mathbf{P}'$ .

*Case 3.1.2.* (I<sub>0</sub>) Observation 10 and  $\gamma \notin c(\text{Inc}_0(x))$  trivially give that  $\mathcal{R}'$  is a realizer of  $\mathbf{P}'$ . A more sophisticated argument is needed to show that  $c'$  is actually a coloring function. Again, fix  $y \in X$ . If  $c(y) \neq \gamma$  then  $y$  remains high in  $L'_{c(y)}$ , by Observation 5(ii). Similarly, if  $y < x$  then  $y$  remains high in any linear extension  $L'$  obtained from  $L$ , in particular, in  $L'_{c(y)}$ . Otherwise,  $y \parallel x$  and  $c(y) = \gamma \notin c(A \cup B)$  (see line 22), which together with Observation 17(iv) gives  $y \in \text{Inc}_0(\mathbf{k})$ . Now, by Observation 17(iii) we have  $y \prec x$  and therefore  $y$  remains high in  $L'_\gamma$ .

(I<sub>1</sub>)–(I<sub>4</sub>) See Case 3.1.1.

(I<sub>5</sub>) Recall from the proof of (I<sub>4</sub>) in Case 3.1.1 that  $x \parallel A \cup B$ ,  $A' = A \cup \{x\}$  and  $B' = B$ . The assumptions for (I<sub>5</sub>) in  $\mathbf{P}'$  are  $\text{width}(A') < \text{width}(A' \cup B)$  and  $|c'(A' \cup B)| = \text{width}(A' \cup B)$ . Together with  $\gamma \notin c(A \cup B)$  we get  $\text{width}(A) = \text{width}(A') - 1 < \text{width}(A' \cup B) - 1 = \text{width}(A \cup B)$  and  $|c(A \cup B)| = |c(A' \cup B)| - 1 = \text{width}(A' \cup B) - 1 = \text{width}(A \cup B)$ . Hence  $\mathbf{b}$  witnessing (I<sub>5</sub>) already existed in  $\mathbf{P}$ . Now, (I<sub>5</sub>-i) holds in  $\mathbf{P}'$  as  $c(x) \notin c(A \cup B)$ . Property (I<sub>5</sub>-ii) is fulfilled in  $\mathbf{P}'$  because  $x$  is put down in  $L_\star$  and  $\mathbf{b}$  remains high in  $L'_\star$  by Observation 5(ii). Finally, (I<sub>5</sub>-iii) holds as  $x \parallel A \cup B$  makes it possible to extend  $I \subseteq A \cup B$  to  $I' = I \cup \{x\} \subseteq A' \cup B$ .

Once more we stress the subtle difference between Cases 3.1.1 and 3.1.2. In Case 3.1.2, if  $\gamma \in c(A \cup B)$  then for  $y \succ x$  with  $c(y) = \gamma$  point  $y$  would no longer be high in  $L'_\gamma$  and the value  $c'(y)$  could not be defined. On the other hand, if  $\gamma \notin c(A \cup B)$  in Case 3.1.1, the witness  $\mathbf{b}$  for (I<sub>5</sub>) would not be high in  $L'_\star$ .

*Case 3.2.* In this case  $x \parallel \mathbf{k}$ ,  $\text{Inc}_0(\mathbf{k}) < x$  and  $c(A \cup B) \subsetneq \Gamma_0$ . As in (I<sub>4</sub>) for Case 3.1.1,  $x \parallel \mathbf{k}$  and  $\mathbf{k} \prec \text{Inc}_1(\mathbf{k}) = A$  imply  $x \parallel A$ ,  $A' = A \cup \{x\}$  and  $B' = B$ . Observation 17(iv) together with  $\text{Inc}_0(\mathbf{k}) < x$  gives  $\text{Inc}(x) \subseteq \text{Inc}_1(\mathbf{k}) \cup \{\mathbf{k}\} \cup \mathbf{k}\uparrow = A \cup \{\mathbf{k}\} \cup B$ . Clearly,  $B \subseteq \text{Inc}_1(x)$ , as otherwise  $x \succ b \in B > \mathbf{k}$  would contradict  $x \parallel \mathbf{k}$ . On the other hand, if  $A \ni a \in \text{Inc}_1(x)$  then there would exist  $a' \in \text{Inc}(x) = A \cup B \cup \{\mathbf{k}\}$  such that  $a' < a$ . Obviously, this is not possible. This proves  $\text{Inc}_0(x) = A \cup \{\mathbf{k}\}$  and  $\text{Inc}_1(x) = B$  (see Figure 12). Since  $\gamma \notin c(A \cup B)$ , we conclude that  $\gamma \notin c(\text{Inc}_0(x))$ . Now, we are almost in the setting from Case 3.1.2. The proofs of invariants (I<sub>0</sub>)–(I<sub>5</sub>) can be repeated from Case 3.1.2.

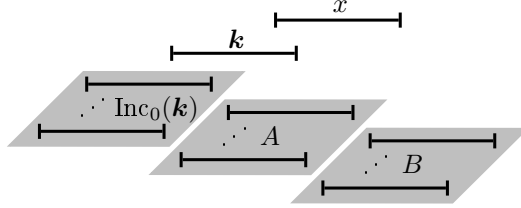


FIGURE 12. Algorithm 2—Cases 3.2 and 3.3

*Case 3.3.* In this case  $x \parallel \mathbf{k}$  and  $\text{Inc}_0(\mathbf{k}) < x$  but now  $c(A \cup B) = \Gamma_0$ . Like in Case 3.2, we argue that  $\text{Inc}_0(x) = A \cup \{\mathbf{k}\}$  and  $\text{Inc}_1(x) = B$  (see Figure 12).

First, we show that point  $\mathbf{b}$  from line 31 exists. Indeed, the set  $A \cup \{x, \mathbf{k}\}$  is an antichain in  $\mathbf{P}'$ , so  $\text{width}(A) \leq w - 2$ . On the other hand, we have  $w - 1 = |\Gamma_0| = |c(A \cup B)| \leq \text{width}(A \cup B) \leq w - 1$ , where the first inequality follows from (I<sub>4</sub>) and the second inequality follows from  $x \parallel A \cup B$ . Hence  $\text{width}(A) < \text{width}(A \cup B)$  and  $|c(A \cup B)| = \text{width}(A \cup B)$ , so a witness  $\mathbf{b}$  from (I<sub>5</sub>) existed in  $\mathbf{P}$ . Note that  $x \parallel \mathbf{k} < \mathbf{b}$  and thus  $x \prec \mathbf{b}$ . Since  $x$  and  $\mathbf{b}$  are maximal in  $\mathbf{P}'$ , the latter gives

$$\text{Inc}'(\mathbf{b}) \subseteq A \cup B \cup \{x\}. \quad (7)$$

So far, we have made no essential use of the mysterious invariant (I<sub>5</sub>). Now comes the moment when (I<sub>5</sub>) plays its tricky role. Line 32 defines the values of  $\star'$ ,  $c'(\mathbf{b})$ , and also, (indirectly) the new set  $\Gamma'_0 := \Gamma - \{\star'\}$ . There are a few issues that need to be discussed.

First of all, by (I<sub>5</sub>-ii), point  $\mathbf{b}$  is high in  $L_{c'(\mathbf{b})}$ , so that the modified function  $c' : X \rightarrow \Gamma$  is still a coloring function for  $(\mathbf{P}, \mathcal{R})$ . By (I<sub>5</sub>-i) we have  $\star' = c(\mathbf{b}) \notin c(A)$  and therefore  $\star' \notin c(A) \cup \{\star\} = c(A \cup \{\mathbf{k}\})$ . Hence

$$c'(\text{Inc}_0(x)) = c'(A \cup \{\mathbf{k}\}) = c(A \cup \{\mathbf{k}\}) \subseteq \Gamma - \{\star'\} = \Gamma'_0. \quad (8)$$

The obtained inclusion  $c'(\text{Inc}_0(x)) \subseteq \Gamma'_0$  proves that Cases 3.3.1 and 3.3.2 are indeed the only possibilities that we are left with. Before considering these cases separately, we make some preparations. First, we will show that

$$\text{act}(\mathbf{P}') \subseteq A \cup B \cup \{x\}. \quad (9)$$

Invariant (I<sub>5</sub>-iii) for  $\mathbf{P}$  supplies us with an antichain  $I \subseteq A \cup B$  of size  $\text{width}(A \cup B) = |c(A \cup B)| = |\Gamma_0| = w - 1$  and such that  $\mathbf{b} \in I$ . This means that  $I \cup \{x\}$  is an antichain witnessing the width  $w$  in  $\mathbf{P}'$ . Now, by our basic Observation 17(i) we get that any future presented maximal point  $x'$  extending  $\mathbf{P}'$  to  $\mathbf{P}'' = (X' \cup \{x'\}, \leq)$  must fulfill  $\mathbf{b} \prec x'$  and hence  $\text{Inc}''(x') \subseteq \text{Inc}'(\mathbf{b}) \cup \{\mathbf{b}\}$ . The latter together with (7) proves (9).

Again,  $c(A \cup B) = \Gamma_0$  together with (I<sub>5</sub>-i) in  $\mathbf{P}$  gives

$$c'(A \cup B - \{\mathbf{b}\}) = c(A \cup B - \{\mathbf{b}\}) = \Gamma_0 - \{c(\mathbf{b})\} \subseteq \Gamma'_0, \quad (10)$$

so that with (9) we get

$$c'(\text{act}(\mathbf{P}') - \{x\}) \subseteq c'(A \cup B) = c'(\mathbf{b}) \cup c'(A \cup B - \{\mathbf{b}\}) \subseteq c'(\mathbf{b}) \cup \Gamma'_0 = \Gamma'_0. \quad (11)$$

We are now ready to analyze the proofs of Cases 3.3.1 and 3.3.2. A quick glance at lines 3–12 and 33–42 brings us to a conclusion that our Algorithm behaves in these cases pretty much the same as in Cases 1 and 2, only with  $c'$  and  $\star'$  in place of  $c$  and  $\star$ , respectively.

*Case 3.3.1.* (I<sub>0</sub>) First, recall that  $c'|_X$  is a coloring function for  $(\mathbf{P}, \mathcal{R})$ . Now, the proof of the fact that  $\mathcal{R}'$  is a realizer of  $\mathbf{P}'$  is pretty much the same as in Case 1.

To prove that  $c'$  is a high-coloring function for  $(\mathbf{P}', \mathcal{R}')$  we will show that  $y$  is high in  $L'_{c'(y)}$  for every  $y \in X \cup \{x\}$ . Clearly,  $x$  is high in  $L'_\gamma$ . Recall that  $\text{Inc}(x) = A \cup B \cup \{\mathbf{k}\}$ . Fix  $y \in X$ . If  $c'(y) = \star' = c(\mathbf{b})$  then invariant (I<sub>5</sub>-i) for  $\mathbf{P}$  together with  $\star' \neq c'(\mathbf{b}), c'(\mathbf{k})$  gives  $y \notin A \cup B \cup \{\mathbf{k}\} = \text{Inc}(x)$ . Hence  $y < x$  and  $y$  remains high in  $L'_{\star'}$ . Now, let  $c'(y) \neq \star'$  and  $y \in \text{Inc}_0(x)$ . Point  $x$  is put down in  $L_{c'(y)}$  and hence  $y$  remains high in  $L'_{c'(y)}$ , by Observation 5(ii). Finally, let  $c'(y) \neq \star'$  and  $y \in X - \text{Inc}_0(x)$ . Then  $x$  and  $y$  are not twins. Again, from Observation 5(i) it follows that  $y$  is high in  $L'_{c'(y)}$ .

(I<sub>1</sub>) By (11) we get  $c'(\text{act}(\mathbf{P}')) \subseteq \Gamma'_0 \cup \{c'(x)\} = \Gamma'_0$ , since  $\gamma \in \Gamma'_0$ .

(I<sub>2</sub>)–(I<sub>5</sub>) Since  $c'(\text{act}(\mathbf{P}')) \subseteq \Gamma'_0$ , the anchor  $\mathbf{k}'$  does not exist in  $\mathbf{P}'$  and the invariants do not apply.

*Case 3.3.2.* (I<sub>0</sub>) The proof of the fact that  $\mathcal{R}'$  is a realizer of  $\mathbf{P}'$  is pretty much the same as in Case 1. To prove that  $c'$  is a high-coloring function for  $(\mathbf{P}', \mathcal{R}')$  we show that  $y$  is high in  $L'_{c'(y)}$  for every  $y \in X \cup \{x\}$ . If  $c'(y) = \star'$  then we repeat the argument from the proof of (I<sub>0</sub>) in Case 3.3.1 to see that  $y$  is high in  $L'_{\star'}$ . Otherwise,  $c'(y) \neq \star'$  and we may follow the proof of (I<sub>0</sub>) in Case 2, only with the function  $c$  replaced by  $c'|_X$ .

(I<sub>1</sub>) follows directly from (11) as the only candidate for an anchor is the point  $x$ .

Invariants (I<sub>2</sub>)–(I<sub>5</sub>) assume there exists an anchor  $\mathbf{k}'$  in  $\mathbf{P}'$ . In our case, it means that  $\mathbf{k}' = x$ . Now, it is easy to see that proofs of (I<sub>2</sub>)–(I<sub>5</sub>) can be obtained by repeating the proofs from Case 2.

This ends the proof of 7 cases of the algorithm. It is easy to check that apart from those 7 cases there are no other possibilities. Thus, Algorithm 2 witnesses Theorem 18.

**Theorem 18.** *Let  $\text{val}(w)$  denote the value of the on-line dimension game on the class of up-growing semi-orders of width at most  $w$ , presented without representation. Then*

$$\text{val}(w) = w.$$

## REFERENCES

- [BBM07] Patrick Baier, Bartłomiej Bosek, and Piotr Micek. On-line chain partitioning of up-growing interval orders. *Order*, 24(1):1–13, 2007.
- [BKKM] Bartłomiej Bosek, Kamil Kloch, Tomasz Krawczyk, and Piotr Micek. On-line version of Rabinovitch theorem for proper intervals. manuscript.
- [Bos08] Bartłomiej Bosek. *On-line chain partitioning approach to scheduling*. PhD thesis, Jagiellonian University, 2008.
- [Fel97] Stefan Felsner. On-line chain partitions of orders. *Theoret. Comput. Sci.*, 175(2):283–292, 1997. Orders, algorithms and applications (Lyon, 1994).
- [FHRT92] Zoltán Füredi, Péter Hajnal, Vojtěch Rödl, and William T. Trotter. Interval orders and shift graphs. In *Sets, graphs and numbers (Budapest, 1991)*, volume 60 of *Colloq. Math. Soc. János Bolyai*, pages 297–313. Amsterdam, 1992.
- [FKMM] Stefan Felsner, Kamil Kloch, Grzegorz Matecki, and Piotr Micek. On-line chain partitioning of up-growing semi-orders. submitted.
- [GL88] András Gyárfás and Jenő Lehel. On-line and first fit colorings of graphs. *J. Graph Theory*, 12(2):217–227, 1988.
- [Hop81] L. Hopkins. *Some problems involving combinatorial structures determined by intersections of intervals and arcs*. PhD thesis, University of South Carolina, 1981.
- [Kie81] Henry A. Kierstead. An effective version of Dilworth’s theorem. *Trans. Amer. Math. Soc.*, 268(1):63–77, 1981.
- [Kie86] Henry A. Kierstead. Recursive ordered sets. In *Combinatorics and ordered sets (Arcata, Calif., 1985)*, volume 57 of *Contemp. Math.*, pages 75–102. Amer. Math. Soc., Providence, RI, 1986.
- [KMT84] Henry A. Kierstead, George F. McNulty, and William T. Trotter, Jr. A theory of recursive dimension for ordered sets. *Order*, 1(1):67–82, 1984.
- [KQ95] Henry A. Kierstead and Jun Qin. Coloring interval graphs with First-Fit. *Discrete Math.*, 144(1-3):47–57, 1995. Combinatorics of ordered sets (Oberwolfach, 1991).
- [KT81] Henry A. Kierstead and William T. Trotter. An extremal problem in recursive combinatorics. In *Proceedings of the Twelfth Southeastern Conference on Combinatorics, Graph Theory and Computing, Vol. II (Baton Rouge, La., 1981)*, volume 33, pages 143–153, 1981.
- [Rab78] Issie Rabinovitch. The dimension of semiorders. *J. Combin. Theory Ser. A*, 25(1):50–61, 1978.
- [SS58] Dana Scott and Patrick Suppes. Foundational aspects of theories of measurement. *J. Symb. Logic*, 23:113–128, 1958.
- [Tro92] William T. Trotter. *Combinatorics and partially ordered sets*. Johns Hopkins Series in the Mathematical Sciences. Johns Hopkins University Press, Baltimore, MD, 1992. Dimension theory.
- [Tro95] William T. Trotter. Partially ordered sets. In *Handbook of combinatorics, Vol. 1, 2*, pages 433–480. Elsevier, Amsterdam, 1995.

ALGORITHMIC RESEARCH GROUP, FACULTY OF MATHEMATICS AND COMPUTER SCIENCE,  
JAGIELLONIAN UNIVERSITY, GRONOSTAJOWA 3, 30-387 KRAKÓW, POLAND

*E-mail address:* bartlomiej.bosek@tcs.uj.edu.pl

*URL:* <http://tcs.uj.edu.pl/Bosek>

*E-mail address:* tomasz.krawczyk@tcs.uj.edu.pl

*URL:* <http://tcs.uj.edu.pl/Krawczyk>

*E-mail address:* piotr.micek@tcs.uj.edu.pl

*URL:* <http://tcs.uj.edu.pl/Micek>

EMBEDDED SYSTEMS LAB, UNIVERSITY OF PASSAU, INNSTRASSE 43, 94032 PASSAU, GERMANY

*E-mail address:* kamil.kloch@uni-passau.de

*URL:* <http://www.esl.fim.uni-passau.de/~kloch/>