Scheduling with Precedence Constraints of Low Fractional Dimension

Christoph Ambühl[1], Monaldo Mastrolilli[2], Nikolaus Mutsanas[2], and Ola Svensson[2]

[1] University of Liverpool - Great Britain
christoph@csc.liv.ac.uk
[2] IDSIA- Switzerland
{monaldo,nikolaus,ola}@idsia.ch

Abstract. We consider the single machine scheduling problem to minimize the average weighted completion time under precedence constrains. Improving on the various 2-approximation algorithms is considered one of the ten most prominent open problems in scheduling theory. Recently, research has focused on special cases of the problem, mostly by restricting the set of precedence constraints to special classes such as convex bipartite, two-dimensional, and interval orders.

In this paper we extend our previous results by presenting a framework for obtaining $(2-2/d)$-approximation algorithms provided that the set of precedence constraints has fractional dimension d. Our generalized approach yields the best known approximation ratios for all previously considered classes of precedence constraints, and it provides the first results for bounded degree and interval dimension 2 orders.

As a negative result we show that the addressed problem remains NP-hard even when restricted to the special case of interval orders.

1 Introduction

The problem we consider in this paper is a classical problem in scheduling theory, known as $1|prec|\sum_j w_jC_j$ in standard scheduling notation (see e.g. Graham et al. [12]). It is defined as the problem of scheduling a set $N = \{1, \ldots, n\}$ of n jobs on a single machine, which can process at most one job at a time. Each job j has a processing time p_j and a weight w_j, where p_j and w_j are nonnegative integers. Jobs also have precedence constraints between them that are specified in the form of a *partially ordered set* (*poset*) $\mathbf{P} = (N, P)$, consisting of the set of jobs N and a partial order i.e. a reflexive, antisymmetric, and transitive binary relation P on N, where $(i, j) \in P$ $(i \neq j)$ implies that job i must be completed before job j can be started. The goal is to find a non-preemptive schedule which minimizes $\sum_{j=1}^{n} w_jC_j$, where C_j is the time at which job j completes in the given schedule.

The described problem was shown to be strongly NP-hard already in 1978 by Lawler [17] and Lenstra & Rinnooy Kan [18]. While currently no inapproximability result is known (other than that the problem does not admit a fully

M. Fischetti and D.P. Williamson (Eds.): IPCO 2007, LNCS 4513, pp. 130–144, 2007.

polynomial time approximation scheme), there are several 2-approximation algorithms [26,29,13,6,5,20,2]. Closing this approximability gap is a longstanding open problem in scheduling theory (see e.g. [30]).

Due to the difficulty to obtain better than 2-approximation algorithms, much attention has recently been given to special cases which manifests itself in recent approximation and exact algorithms [16,33,7,2,3].

On the negative side, Woeginger [33] proved that many quite severe restrictions on the weights and processing times do not influence approximability. For example, the special case in which all jobs either have $p_j = 1$ and $w_j = 0$, or $p_j = 0$ and $w_j = 1$, is as hard to approximate as the general case. This suggests that in order to identify classes of instances which allow a better than 2-approximation one has to focus on the precedence constraints rather than the weights and processing times.

Indeed, Lawler [17] gave an exact algorithm for series-parallel orders already in 1978. For interval orders and convex bipartite precedence constraints, Woeginger [33] gave approximation algorithms with approximation ratio arbitrarily close to the golden ratio $\frac{1}{2}(1+\sqrt{5}) \approx 1.61803$.

Recently, Ambühl & Mastrolilli [2] settled an open problem first raised by Chudak & Hochbaum [6] and whose answer was subsequently conjectured by Correa & Schulz [7]. The results in [2,7] imply that $1|prec|\sum w_j C_j$ is a special case of the weighted vertex cover problem. More precisely, they proved that every instance S of $1|prec|\sum w_j C_j$ can be translated in polynomial time into a weighted graph $G_{\mathbf{P}}$, such that finding the optimum of S can be reduced to finding an optimum vertex cover in $G_{\mathbf{P}}$. This result even holds for approximate solutions: Finding an α-approximate solution for S can be reduced to finding an α-approximate vertex cover in $G_{\mathbf{P}}$.

Based on these results, three of the authors [3] discovered an interesting connection between $1|prec|\sum w_j C_j$ and the dimension theory of posets [32], by observing that the graph $G_{\mathbf{P}}$ is well known in dimension theory as the graph of incomparable pairs of a poset $\mathbf{P}$. Applying results from dimension theory allowed to describe a framework for obtaining simple and efficient approximation algorithms for $1|prec|\sum w_j C_j$ with precedence constraints of low dimension, such as convex bipartite and semi-orders. In both cases, the new 4/3-approximation algorithms outperform the previously known results. The approach even yields a polynomial algorithm for 2-dimensional precedence constraints, based on the fact that the minimum weighted vertex cover on $G_{\mathbf{P}}$ can be solved in polynomial time since $G_{\mathbf{P}}$ is bipartite for a 2-dimensional poset $\mathbf{P}$ [32,7]. This considerably extends Lawler's result [17] for series-parallel orders. Unfortunately, the framework in [3] fails in the case of interval orders (in this case the dimension can be of the order of $\log \log n$ [32]).

The work in this paper originated from the study of $1|prec|\sum w_j C_j$ under interval orders (abbreviated $1|interval\text{-}order|\sum_j w_j C_j$). Interval orders appear in many natural contexts [10]. We provide both positive and negative results.

In the first part of the paper, we further generalize our previous framework [3] such that it can be applied to precedence constraints of low *fractional* dimension [4] (Section 3). The extended framework yields $(2-2/d)$-approximation algorithms whenever precedence constraints have fractional dimension bounded by a constant d and satisfy a mild condition (see Section 3). Since the fractional dimension of interval orders is bounded by 4 (see Section 4.1), this gives a 1.5-approximation algorithm and improves the previous result in [33]. The extended framework can also be applied to interval dimension two posets (Section 4.2), bounded degree posets (Section 4.3), and posets obtained by the lexicographic sums (Section 4.4).

In the second part of the paper, we show that $1|\textit{interval-order}|\sum_j w_j C_j$ remains NP-hard (Section 5). This result is rather unexpected as many problems can be solved in polynomial time when restricted to interval orders (see e.g. [25]). The reduction heavily relies on the connection between $1|\textit{prec}\,|\sum w_j C_j$ and weighted vertex cover described in [2].

In summary, our results indicate a strong relationship between the approximability of $1|\textit{prec}|\sum_j w_j C_j$ and the fractional dimension d of the precedence constraints. In particular, it is polynomial for $d = 2$, but NP-hard already for $d \geq 3$. The latter stems from the facts that problem $1|\textit{prec}|\sum_j w_j C_j$ is strongly NP-hard even for posets with in-degree 2 [17], and the fractional dimension of these posets is bounded by 3 [8]. This leaves the complexity for $2 < d < 3$ as an open question.

2 Definitions and Preliminaries

2.1 Posets and Fractional Dimension

Let $\mathbf{P} = (N, P)$ be a poset. For $x, y \in N$, we write $x \leq y$ when $(x, y) \in P$, and $x < y$ when $(x, y) \in P$ and $x \neq y$. When neither $(x, y) \in P$ nor $(y, x) \in P$, we say that x and y are incomparable, denoted by $x||y$. We call $\text{inc}(\mathbf{P}) = \{(x, y) \in N \times N : x||y \text{ in } P\}$ the set of *incomparable pairs* of $\mathbf{P}$. A poset $\mathbf{P}$ is a *linear order* (or a *total order*) if for any $x, y \in N$ either $(x, y) \in P$ or $(y, x) \in P$, i.e. $\text{inc}(\mathbf{P}) = \emptyset$. A partial order P' on N is an *extension* of a partial order P on the same set N, if $P \subseteq P'$. An extension that is a linear order is called a *linear extension*. Mirroring the definition of the fractional chromatic number of a graph, Brightwell & Scheinerman [4] introduce the notion of fractional dimension of a poset. Let $\mathcal{F} = \{L_1, L_2, \ldots, L_t\}$ be a nonempty multiset of linear extensions of $\mathbf{P}$. The authors in [4] call $\mathcal{F}$ a *k-fold realizer* of $\mathbf{P}$ if for each incomparable pair (x, y), there are at least k linear extensions in $\mathcal{F}$ which reverse the pair (x, y), i.e., $|\{i = 1, \ldots, t : y < x \text{ in } L_i\}| \geq k$. We call a k-fold realizer of size t a *k:t-realizer*. The *fractional dimension of* $\mathbf{P}$ is then the least rational number $\text{fdim}(\mathbf{P}) \geq 1$ for which there exists a *k:t*-realizer of $\mathbf{P}$ so that $k/t \geq 1/\text{fdim}(\mathbf{P})$. Using this terminology, the *dimension* of $\mathbf{P}$, denoted by $\dim(\mathbf{P})$, is the least t for which there exists a 1-fold realizer of P. It is immediate that $\text{fdim}(\mathbf{P}) \leq \dim(\mathbf{P})$ for any poset $\mathbf{P}$. Furthermore [4], $\text{fdim}(\mathbf{P}) = 1$, or $\text{fdim}(\mathbf{P}) \geq 2$.

2.2 Scheduling, Vertex Cover, and Dimension Theory

In [7,2,3] a relationship between $1|\textit{prec}|\sum_j w_j C_j$, weighted vertex cover, and the dimension theory of posets is shown. This relationship will turn out to be useful for both improving the approximation ratio for several classes of precedence constraints and establishing the NP-hardness of $1|interval\text{-}order|\sum_j w_j C_j$.

Let $\mathbf{P} = (N, P)$ be any poset, that is not a linear order. Felsner and Trotter [9] associate with $\mathbf{P}$ a hypergraph $\mathcal{H}_\mathbf{P}$, called the *hypergraph of incomparable pairs*, defined as follows. The vertices of $\mathcal{H}_\mathbf{P}$ are the incomparable pairs in $\mathbf{P}$. The edge set consists of those sets U of incomparable pairs such that no linear extension of $\mathbf{P}$ reverses all incomparable pairs in U. Let $G_\mathbf{P}$ denote the ordinary graph, called the *graph of incomparable pairs*, determined by all edges of size 2 in $\mathcal{H}_\mathbf{P}$. In [9,32] it is shown that the dimension of $\mathbf{P}$ is equal to the chromatic number of $\mathcal{H}_\mathbf{P}$, i.e., $dim(\mathbf{P}) = \chi(\mathcal{H}_\mathbf{P}) \geq \chi(G_\mathbf{P})$. In [4], it was noted that the same relationship holds for the fractional versions, i.e., $\text{fdim}(\mathbf{P}) = \chi_f(\mathcal{H}_\mathbf{P}) \geq \chi_f(G_\mathbf{P})$. We refer the reader to [28] for an introduction to fractional graph coloring.

Given an instance S of $1|prec|\sum_j w_j C_j$, we associate with S a weighted vertex cover instance VC_S on $G_\mathbf{P}$, where $G_\mathbf{P}$ is the graph of incomparable pairs of the poset $\mathbf{P}$ representing the precedence constraints and each vertex $(i,j) \in \text{inc}(\mathbf{P})$ has weight $p_i \cdot w_j$. We denote the *value* of a solution s by $\text{val}(s)$.

Theorem 1 ([2,3,7]). *Let S be an instance of $1|prec|\sum_j w_j C_j$ where precedence constraints are given by the poset $\mathbf{P} = (N, P)$. Then the following transformations can be performed in polynomial time.*

1. *Any feasible solution s' of S can be turned into a feasible solution c' of VC_S, such that*
$$\text{val}(c') \leq \text{val}(s') - \sum_{(i,j)\in P} p_i \cdot w_j.$$
2. *Any feasible solution c' to VC_S can be turned into a feasible solution s' of S, such that*
$$\text{val}(s') \leq \text{val}(c') + \sum_{(i,j)\in P} p_i \cdot w_j.$$

In particular, if c^* and s^* are optimal solutions to VC_S and S, respectively, we have $\text{val}(c^*) = \text{val}(s^*) - \sum_{(i,j)\in P} p_i \cdot w_j$.

We remark that the term $\sum_{(i,j)\in P} p_i \cdot w_j$ is a *fixed cost* and it is present in all feasible schedules of S. This follows from the facts that a job's processing time is always included in its completion time, and any feasible schedule of S must schedule job i before job j if $i < j$ in P.

3 Scheduling and Fractional Dimension

In this section, we present an algorithmic framework that can be used to obtain better than 2-approximation algorithms provided that the set of precedence constraints has low fractional dimension. Applications that follow this pattern are given in Section 4.

We say that a poset $\mathbf{P}$ admits an *efficiently samplable k:t*-realizer if there exists a randomized algorithm that, in polynomial time, returns any linear extension from a k-fold realizer $\mathcal{F} = \{L_1, L_2, \ldots, L_t\}$ with probability $1/t$.

Let S be an instance of $1|prec|\sum_j w_j C_j$ where precedence constraints are given by a poset $\mathbf{P} = (N, P)$. Assuming that $\mathbf{P}$ admits an efficiently samplable $k{:}t$-realizer $\mathcal{F} = \{L_1, \ldots, L_t\}$, we proceed as follows.

Let $V_{\mathbf{P}}$ and $E_{\mathbf{P}}$ be the vertex set and edge set, respectively, of the graph of incomparable pairs $G_{\mathbf{P}}$. Consider the following integer program formulation of the weighted vertex cover VC_S:

$$\begin{array}{lll} \min & \sum_{i \in V_{\mathbf{P}}} w_i x_i & \\ \text{s.t.} & x_i + x_j \geq 1 & \{i,j\} \in E_{\mathbf{P}} \\ & x_i \in \{0,1\} & i \in V_{\mathbf{P}} \end{array}$$

where w_i denotes the weight of vertex $v_i \in V_{\mathbf{p}}$, as specified in the definition of VC_S (see Section 2.2). Let [VC-LP] denote the linear relaxation of the integer program above.

Nemhauser & Trotter [23,24] proved that any basic feasible solution to [VC-LP] is *half-integral*, that is $x_i \in \{0, \frac{1}{2}, 1\}$ for all $i \in V$. Let V_i be the set of nodes whose corresponding variables took value $i \in \{0, \frac{1}{2}, 1\}$ in the optimal solution of [VC-LP].

Observe that for any linear extension L, the set of all incomparable pairs that are reversed in L is an independent set in the graph of incomparable pairs $G_{\mathbf{P}}$. Now, pick uniformly at random a linear extension L of $\mathcal{F}$ in polynomial time. Note that $V_0 \cup (V_{1/2} \setminus L)$ defines an independent set of $G_{\mathbf{P}}$. Generalizing a result by Hochbaum in [14], we prove that the complement of $V_0 \cup (V_{1/2} \setminus L)$ is a vertex cover whose expected value is within $(2 - 2\frac{k}{t})$ times the weight of an optimum cover. By Theorem 1, we can transform (in polynomial time) the solution of VC_S into a feasible solution of S of expected value at most $(2 - 2\frac{k}{t})$ times the value of an optimum schedule. We summarize the above arguments in the following theorem.

Theorem 2. *The problem $1|prec|\sum_j w_j C_j$, whenever precedence constraints admit an efficiently samplable k:t-realizer, has a randomized $(2-2\frac{k}{t})$-approximation algorithm.*

For a proof of this theorem, see Appendix A.1. Following a similar argumentation, Hochbaum's approach [14] for approximating the vertex cover problem can be extended to fractional coloring, yielding the same approximation result.

A natural question is for which posets one can have an efficiently samplable $k{:}t$-realizer. In the general case, Jain & Hedge [15] recently proved that it is hard to approximate the dimension of a poset with n elements within a factor $n^{0.5-\epsilon}$, and the same hardness of approximation holds for the fractional dimension. However, for several special cases, including interval orders (Section 4.1) and bounded degree posets (Section 4.3), efficiently samplable $k{:}t$-realizers exist.

4 Precedence Constraints with Low Fractional Dimension

4.1 Interval Orders

A poset $\mathbf{P} = (N, P)$ is an *interval order* if there is a function F, which assigns to each $x \in N$ a closed interval $F(x) = [a_x, b_x]$ of the real line $\mathbb{R}$, so that $x < y$ in P if and only if $b_x < a_y$ in $\mathbb{R}$. Interval orders can be recognized in $O(n^2)$ time [21,25]. The dimension of interval orders can be of the order of $\log \log n$ [32], whereas the fractional dimension is known to be less than 4 [4], and this bound is asymptotically tight [8]. In the following we show how to obtain a 1.5-approximation algorithm for $1|\textit{interval-order}|\sum_j w_j C_j$. By Theorem 2, it is sufficient to prove that interval orders admit an efficiently samplable k:t-realizer with $t/k = 4$.

Given a poset $\mathbf{P} = (N, P)$, disjoint subsets A and B of the ground set N, and a linear extension L of P, we say that B *is over* A *in* L if, for every incomparable pair of elements (a, b) with $a \in A$ and $b \in B$, one has $b > a$ in L. The following property of interval orders is fundamental.

Theorem 3 (Rabinovitch [27,10]). *A poset $\mathbf{P} = (N, P)$ is an interval order if and only if for every pair (A, B) of disjoint subsets of N there is a linear extension L of P with B over A.*

By using this property we can easily obtain a k-fold realizer $\mathcal{F} = \{L_1, \ldots, L_t\}$ with $k = 2^{n-2}$ and $t = 2^n$, where $n = |N|$. Indeed, consider every subset A of N and let L_A be a linear extension of P in which $B = N \setminus A$ is over A. Now let $\mathcal{F}$ be the multiset of all the L_A's. Note that $|\mathcal{F}| = 2^n$. Moreover, for any incomparable pair (x, y) there are at least $k = 2^{n-2}$ linear extensions in $\mathcal{F}$ for which $x \in B$ and $y \in A$. Finally, observe that we can efficiently pick uniformly at random one linear extension from $\mathcal{F}$: for every job $j \in N$ put j either in A or in B with the same probability 1/2.

By the previous observations and Theorem 2, we have a randomized polynomial time 1.5-approximation for $1|\textit{interval-order}|\sum_j w_j C_j$. The described algorithm can easily be derandomized by using the classical method of conditional probabilities.

Theorem 4. *Problem* $1|\text{interval-order}|\sum_j w_j C_j$ *has a deterministic polynomial time 1.5-approximation algorithm.*

4.2 Interval Dimension Two

The *interval dimension* of a poset $\mathbf{P} = (N, P)$, denoted by $\dim_I(\mathbf{P})$, is defined [32] as the least t for which there exist t extensions $Q_1, Q_2, \ldots, Q_t$, so that:

- $P = Q_1 \cap Q_2 \cap \cdots \cap Q_t$ and
- (N, Q_i) is an interval order for $i = 1, 2, \ldots, t$.

Generally $\dim_I(\mathbf{P}) \leq \dim(\mathbf{P})$. Obviously, if $\mathbf{P}$ is an interval order, $\dim_I(\mathbf{P}) = 1$.

The class of posets of interval dimension 2 forms a proper superclass of the class of interval orders. Posets of interval dimension two can be recognized in

$O(n^2)$ time due to Ma & Spinrad [19]. Given a poset $\mathbf{P}$ with $\dim_I(\mathbf{P}) = 2$, their algorithm also yields an interval realizer $\{Q_1, Q_2\}$. As described in Section 4.1, we obtain k-fold realizers $\mathcal{F}_1 = \{L_1, L_2, \ldots, L_t\}$ and $\mathcal{F}_2 = \{L'_1, L'_2, \ldots, L'_t\}$ of Q_1 and Q_2, respectively, with $k = 2^{n-2}$ and $t = 2^n$. It is immediate that $\mathcal{F} = \mathcal{F}_1 \cup \mathcal{F}_2$ is a k-fold realizer of $\mathbf{P}$ of size $2t = 2^{n+1}$. Furthermore, we can efficiently pick uniformly at random one linear extension from $\mathcal{F}$: pick uniformly at random a linear extension from either $\mathcal{F}_1$ or $\mathcal{F}_2$ with the same probability 1/2. Again by using conditional probabilities we have the following.

Theorem 5. *Problem* $1|prec|\sum_j w_j C_j$*, whenever precedence constraints have interval dimension at most* 2*, has a polynomial time* 1.75*-approximation algorithm.*

4.3 Posets of Bounded Degree

In the following we will see how to obtain, using Theorem 2, an approximation algorithm for $1|prec|\sum w_j C_j$ when the precedence constraints form a poset of bounded degree. Before we proceed, we need to introduce some definitions.

Let $\mathbf{P} = (N, P)$ be a poset. For any job $j \in N$, define the *degree of* j, denoted $\deg(j)$, as the number of jobs comparable (but not equal) to j in $\mathbf{P}$. Let $\Delta(\mathbf{P}) = \max\{\deg(j) : j \in N\}$. Given a job j, let $D(j)$ denote the set of all jobs which are less than j, and $U(j)$ those which are greater than j in P. Define $\deg_D(j) = |D(j)|$ and $\Delta_D(\mathbf{P}) = \max\{\deg_D(j) : j \in N\}$. The quantities $\deg_U(j)$ and $\Delta_U(\mathbf{P})$ are defined dually.

We observe that the NP-completeness proof for $1|prec|\sum w_j C_j$ given by Lawler [17] was actually provided for posets $\mathbf{P}$ with $\Delta_D(\mathbf{P}) = 2$. By using fractional dimension we show that these posets (with bounded $\min\{\Delta_D, \Delta_U\}$) allow for better than 2-approximation.

Theorem 6. *Problem* $1|prec|\sum w_j C_j$ *has a polynomial time* $(2 - 2/f)$*-approximation algorithm, where* $f = 1 + \min\{\Delta_D, \Delta_U, 1\}$*.*

Proof. Let $\mathbf{P} = (N, P)$ be the poset representing the precedence constraints with bounded $\min\{\Delta_D, \Delta_U\}$. Assume, without loss of generality, that $\mathbf{P}$ is *not* decomposable with respect to lexicographic sums (see Section 4.4). Otherwise, a decomposition with respect to lexicographic sums can be done in $O(n^2)$ time (see e.g. [22]), and each component can be considered separately. We call an incomparable pair $(x, y) \in \text{inc}(\mathbf{P})$ a *critical pair* if for all $z, w \in N \setminus \{x, y\}$

1. $z < x$ in P implies $z < y$ in P, and
2. $y < w$ in P implies $x < w$ in P.

Critical pairs play an important role in dimension theory: if for each critical pair (x, y), there are at least k linear extensions in $\mathcal{F}$ which reverse the pair (x, y) then $\mathcal{F}$ is a k-fold realizer of P and vice versa [4].

For any permutation M of N, consider the set $C(M)$ of critical pairs (x, y) that satisfy the following two conditions:

1. $x > (D(y) \cup \{y\})$ in M if $|D(y)| < \Delta_D$
2. $x > D(y)$ in M if $|D(y)| = \Delta_D$

In [8], Felsner & Trotter present an algorithm that converts in polynomial time a permutation M of N to a linear extension L of P so that L reverses all critical pairs in the set $C(M)$. Now set $t = |N|!$ and consider the set $\mathcal{M} = \{M_1, M_2, \ldots, M_t\}$ of all permutations of the ground set N. Observe that for any critical pair (x, y) there are at least $n!/(\Delta_D + 1)$ different permutations $M_i \in \mathcal{M}$, where the critical pair is reversed, i.e., $(y, x) \in C(M_i)$. Applying the algorithm in [8] we obtain a k-fold realizer $\mathcal{F} = \{L_1, \ldots, L_t\}$ of P with $t = n!$ and $k = n!/(\Delta_D + 1)$. Moreover, we can efficiently pick uniformly at random one linear extension from $\mathcal{F}$: generate uniformly at random one permutation of jobs (e.g. by using Knuth's shuffle algorithm) and transform it into a linear extension with the described properties by using the algorithm in [8]. The described algorithm can be derandomized by using the classical method of conditional probabilities. Finally observe that we can repeat a similar analysis by using Δ_U instead of Δ_D. □

In fact, this result is stronger than the same statement with $d = \Delta(\mathbf{P})$. To see this, consider the *graph poset* $\mathbf{P}(G) = (N, P)$ defined as follows: given an undirected graph $G(V, E)$, let $N = V \cup E$ and for every $v \in V$ and $e = \{v_1, v_2\} \in E$, put $(v, e) \in P$ if and only if $v \in \{v_1, v_2\}$. If $\Delta(G)$ is unbounded, this also holds for $\Delta(\mathbf{P})$. However, since every edge is adjacent to only two vertices, Δ_D is bounded by 2, thus the value $1 + \min\{\Delta_U, \Delta_D\}$ is also bounded. On the other hand, for the complete graph on n nodes, K_n, Spencer [31] showed that $\dim(\mathbf{P}(K_n)) = \Theta(\log \log n)$. Therefore, the poset $\mathbf{P}(K_n)$ is an example where the dimension of the poset is unbounded, while $\min\{\Delta_D, \Delta_U\}$ (and thus also the fractional dimension) is bounded. This means that the fractional dimension approach can yield a substantially better result than the dimension approach used in [3].

4.4 Lexicographic Sums

In this section we show how to use previous results to obtain approximation algorithms for new ordered sets. The construction we use here, *lexicographic sums*, comes from a very simple pictorial idea (see [32] for a more comprehensive discussion). Take a poset $\mathbf{P} = (N, P)$ and replace each of its points $x \in N$ with a partially ordered set $\mathbf{Q_x}$, the *module*, such that the points in the module have the same relation to points outside it. A more formal definition follows. For a poset $\mathbf{P} = (N, P)$ and a family of posets $\mathcal{S} = \{(Y_x, Q_x) \mid x \in N\}$ indexed by the elements in N, the lexicographic sum of $\mathcal{S}$ over (N, P), denoted $\sum_{x \in (N,P)} (Y_x, Q_x)$ is the poset (Z, R) where $Z = \{(x, y) \mid x \in N, y \in Y_x\}$ and $(x_1, y_1) \le (x_2, y_2)$ in R if and only if one of the following two statements holds:

1. $x_1 < x_2$ in P.
2. $x_1 = x_2$ and $y_1 \le y_2$ in Q_{x_1}.

We call $\mathcal{P} = P \cup \mathcal{F}$ the *components* of the lexicographic sum. A lexicographic sum is *trivial* if $|N| = 1$ or if $|Y_x| = 1$ for all $x \in N$. A poset is *decomposable with respect to lexicographic sums* if it is isomorphic to a non-trivial lexicographic sum.

In case the precedence constraints of every component admit an efficiently samplable realizer, we observe that this translates into a randomized approximation algorithm:

Theorem 7. *Problem* $1|prec|\sum_j w_j C_j$*, whenever precedence constraints form a lexicographic sum whose components* $i \in \mathcal{P}$ *admit efficiently samplable realizers, has a polynomial time randomized* $(2 - \frac{2t}{k})$*−approximation algorithm, where* $t/k = \max_{i\in\mathcal{P}}(t_i/k_i)$*.*

Finally, we point out that, if the approximation algorithm for each component can be derandomized, this yields a derandomized approximation algorithm for the lexicographic sum.

5 NP-Completeness for Interval Orders

In this section we show that $1|prec|\sum_j w_j C_j$ remains NP-complete even in the special case of interval order precedence constraints. To prove this we exploit the vertex cover nature of problem $1|prec|\sum w_j C_j$.

Theorem 8. *Problem* $1|\text{interval-order}|\sum_j w_j C_j$ *is* NP*-complete.*

Proof. A graph G is said to have bounded degree d if every vertex v in G is adjacent to at most d other vertices. The problem of deciding if a graph G with bounded degree 3 has a (unweighted) vertex cover of size at most m is NP-complete [11]. We provide a reduction from the minimum vertex cover on graphs with bounded degree 3 to $1|\textit{interval-order}|\sum_j w_j C_j$.

Given a connected graph $G = (V, E)$ with bounded degree 3, we construct an instance S of $1|\textit{interval-order}|\sum_j w_j C_j$ so that S has a schedule with value less than $m + c + 1$ if and only if G has a vertex cover of size at most m, where c is a fixed value defined later (see Equation (1)). We present the construction of S in two stages.

Stage 1 (Tree-layout of the graph). Starting from any vertex $s \in V$, consider the tree $T = (V, E_T)$, with $E_T \subseteq E$, rooted at s on the set of nodes reachable from s by using, for example, breadth-first search. Furthermore, we number the vertices of T top-down and left-right. Figure 1 shows the breadth-first search tree T for K_4.

Define $G' = (V', E')$ to be the graph obtained from T in the following way. For each vertex v_i in T we add two new vertices u_2^i, u_1^i and edges $\{u_2^i, u_1^i\}, \{u_1^i, v_i\}$. Furthermore, for each edge $\{v_i, v_j\} \in E \setminus E_T$ with $i < j$ we add vertices e_1^{ij}, e_2^{ij} and edges $\{v_i, e_1^{ij}\}, \{e_1^{ij}, e_2^{ij}\}, \{e_2^{ij}, u_2^j\}$.

The following claim relates the optimum unweighted vertex covers of G and G'.

Claim 1. Let $C_* \subseteq V$ and $C'_* \subseteq V'$ be optimum vertex cover solutions to G and G', respectively, then $|C_*| = |C'_*| - |V| - |E \setminus E_T|$. (For a proof, see Appendix A.2).

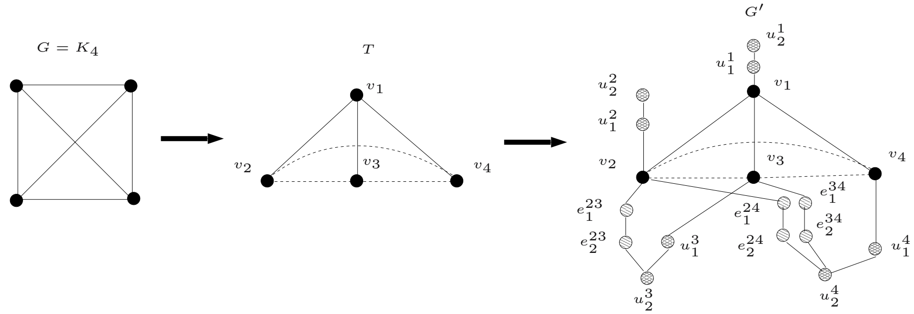

Fig. 1. The breadth first search tree $T = (V, E_T)$ for the graph $G = K_4$, and the graph G'. The solid edges belong to E_T.

Stage 2 (Construction of scheduling instance). Given the vertex cover graph $G = (V, E)$ and its corresponding tree $T = (V, E_T)$, we construct the scheduling instance S with processing times, weights, and precedence constraints to form an interval order I as defined below (see Figure 2 for an example), where k is a value to be determined later.

Job	**Interval Repr.**	**Proc. Time**	**Weight**
s_0	[-1,0]	1	0
s_1	[0, 1]	$1/k$	1
$s_j, j = 2, \ldots, \lvert V\rvert$	$[i, j]$, where $\{v_i, v_j\} \in E_T,\ i < j$	$1/k^j$	k^i
$m_i, i = 1, \ldots, \lvert V\rvert$	$[i - \frac{1}{2}, \lvert V\rvert + i]$	$1/k^{(\lvert V\rvert+i)}$	k^i
$e_i, i = 1, \ldots, \lvert V\rvert$	$[\lvert V\rvert + i, \lvert V\rvert + i + 1]$	0	$k^{(\lvert V\rvert+i)}$
b_{ij}, where $\{v_i, v_j\} \in E \setminus E_T, i < j$	$[i, j - \frac{1}{2}]$	$1/k^j$	k^i

Remark 1. Let i and j be two jobs in S with interval representations $[a, b]$ and $[c, d]$ respectively, where $a \leq d$. By the construction of the scheduling instance S we have $p_i \leq 1/k^{\lceil b \rceil}$ and $w_j \leq k^{\lceil c \rceil}$. It follows that $p_i \cdot w_j = 1$ or $p_i \cdot w_j \leq 1/k$ if i and j are incomparable, since $p_i \cdot w_j \geq k$ implies that $b < c$, i.e., i's interval representation is completely to the left of j's interval representation. Furthermore, if $p_i \cdot w_j = 1$ then $\lceil b \rceil = \lceil c \rceil$.

$$
\begin{aligned}
\text{Let } D = \; & \{(s_0, s_1)\} \\
& \cup \{(s_i, s_j) : v_i \text{ is the parent of } v_j \text{ in } T\} \\
& \cup \{(s_i, m_i), (m_i, e_i) : i = 1, 2, \ldots, |V|\} \\
& \cup \{(s_i, b_{ij}), (b_{ij}, m_j) : \{v_i, v_j\} \in E \setminus E_T, i < j\}
\end{aligned}
$$

By the interval representation of the jobs and the remark above, we have the following:

Claim 2. A pair of incomparable jobs (i,j) has $p_i \cdot w_j = 1$ if $(i, j) \in D$; otherwise if $(i, j) \notin D$ then $p_i \cdot w_j \leq 1/k$.

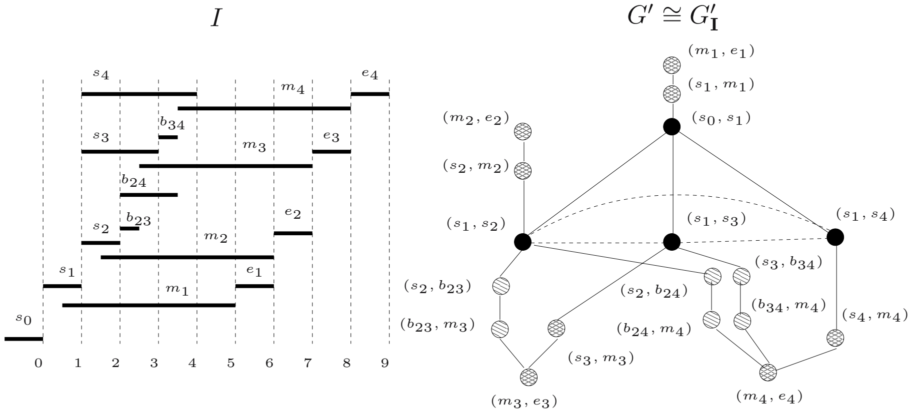

Fig. 2. The interval order I obtained from K_4; $G'_{\mathbf{I}}$ is the subgraph induced on the graph of incomparable pairs $G_{\mathbf{I}}$ by the vertex subset D (the vertices with weight 1)

Claim 3. Let $G'_{\mathbf{I}} = (D, E_I)$ be the subgraph induced on the graph of incomparable pairs $G_{\mathbf{I}}$ by the vertex subset D. Then G' and $G'_{\mathbf{I}}$ are isomorphic. (For a proof, see Appendix A.3).

By Claim 2, each incomparable pair of jobs $(i, j) \notin D$ satisfies $p(i) \cdot w(j) \leq 1/k$. Let n be the number of jobs in the scheduling instance S and select k to be n^2+1. Let C,C_I, and C'_I be optimal vertex cover solutions to G, $G_{\mathbf{I}}$ and $G'_{\mathbf{I}}$ (defined as in Claim 3), respectively. Then, by the selection of k and Claim 2, we have $|C'_I| \leq |C_I| \leq |C'_I| + \sum_{(i,j) \in \mathrm{inc}(I) \setminus D} p_i w_j < |C'_I| + 1$. Furthermore, Claims 3 and 1 give us that $|C| + |V| + |E \setminus E_T| \leq |C_I| < |C| + |V| + |E \setminus E_T| + 1$. This, together with Theorem 1, implies that $|C| \leq m$ if and only if there is a schedule of S with value less than $m + c + 1$, where

$$c = |V| + |E \setminus E_T| + \sum_{(i,j) \in I} p_i \cdot w_j . \tag{1}$$

□

Acknowledgements

We are grateful to Andreas Schulz for many helpful discussions we had with him during his visit at IDSIA. This research is supported by Swiss National Science Foundation project 200021-104017/1, "Power Aware Computing", and by the Swiss National Science Foundation project 200020-109854, "Approximation Algorithms for Machine scheduling Through Theory and Experiments II". The first author is supported by Nuffield Foundation Grant NAL32608.

References

1. P. Alimonti and V. Kann. Some APX-completeness results for cubic graphs. *Theor. Comput. Sci.*, 237(1-2):123–134, 2000.
2. C. Ambühl and M. Mastrolilli. Single machine precedence constrained scheduling is a vertex cover problem. In *Proceedings of the 14th Annual European Symposium on Algorithms (ESA)*, volume 4168 of *Lecture Notes in Computer Science*, pages 28–39. Springer, 2006.
3. C. Ambühl, M. Mastrolilli, and O. Svensson. Approximating precedence-constrained single machine scheduling by coloring. In *Proceedings of APPROX + RANDOM*, volume 4110 of *Lecture Notes in Computer Science*, pages 15–26. Springer, 2006.
4. G. R. Brightwell and E. R. Scheinerman. Fractional dimension of partial orders. *Order*, 9:139–158, 1992.
5. C. Chekuri and R. Motwani. Precedence constrained scheduling to minimize sum of weighted completion times on a single machine. *Discrete Applied Mathematics*, 98(1-2):29–38, 1999.
6. F. A. Chudak and D. S. Hochbaum. A half-integral linear programming relaxation for scheduling precedence-constrained jobs on a single machine. *Operations Research Letters*, 25:199–204, 1999.
7. J. R. Correa and A. S. Schulz. Single machine scheduling with precedence constraints. *Mathematics of Operations Research*, 30(4):1005–1021, 2005. Extended abstract in Proceedings of the 10th Conference on Integer Programming and Combinatorial Optimization (IPCO 2004), pages 283–297.
8. S. Felsner and W. T. Trotter. On the fractional dimension of partially ordered sets. *DMATH: Discrete Mathematics*, 136:101–117, 1994.
9. S. Felsner and W. T. Trotter. Dimension, graph and hypergraph coloring. *Order*, 17(2):167–177, 2000.
10. P. C. Fishburn. *Interval Orders and Interval Graphs*. John Wiley and Sons, 1985.
11. M. R. Garey, D. S. Johnson, and L. J. Stockmeyer. Some simplified NP-complete graph problems. *Theor. Comput. Sci.*, 1(3):237–267, 1976.
12. R. Graham, E. Lawler, J. K. Lenstra, and A. H. G. Rinnooy Kan. Optimization and approximation in deterministic sequencing and scheduling: A survey. In *Annals of Discrete Mathematics*, volume 5, pages 287–326. North–Holland, 1979.
13. L. A. Hall, A. S. Schulz, D. B. Shmoys, and J. Wein. Scheduling to minimize average completion time: off-line and on-line algorithms. *Mathematics of Operations Research*, 22:513–544, 1997.
14. D. S. Hochbaum. Efficient bounds for the stable set, vertex cover and set packing problems. *Discrete Applied Mathematics*, 6:243–254, 1983.
15. K. Jain, R. Hedge. Some inapproximability results for the (fractional) poset dimension *Personal communication*, 2006
16. S. G. Kolliopoulos and G. Steiner. Partially-ordered knapsack and applications to scheduling. In *Proceedings of the 10th Annual European Symposium on Algorithms (ESA)*, pages 612–624, 2002.
17. E. L. Lawler. Sequencing jobs to minimize total weighted completion time subject to precedence constraints. *Annals of Discrete Mathematics*, 2:75–90, 1978.
18. J. K. Lenstra and A. H. G. Rinnooy Kan. The complexity of scheduling under precedence constraints. *Operations Research*, 26:22–35, 1978.
19. T.-H. Ma and J. P. Spinrad. On the 2-chain subgraph cover and related problems. *J. Algorithms*, 17(2):251–268, 1994.

20. F. Margot, M. Queyranne, and Y. Wang. Decompositions, network flows and a precedence constrained single machine scheduling problem. *Operations Research*, 51(6):981–992, 2003.
21. R. H. Möhring. Computationally tractable classes of ordered sets. In I. Rival, editor, *Algorithms and Order*, pages 105–193. Kluwer Academic, 1989.
22. R. H. Möhring. Computationally tractable classes of ordered sets. *Algorithms and Order*, pages 105–194, 1989.
23. G. L. Nemhauser and L. E. Trotter. Properties of vertex packing and independence system polyhedra. *Mathematical Programming*, 6:48–61, 1973.
24. G. L. Nemhauser and L. E. Trotter. Vertex packings: Structural properties and algorithms. *Mathematical Programming*, 8:232–248, 1975.
25. C. H. Papadimitriou and M. Yannakakis. Scheduling interval-ordered tasks. *SIAM Journal of Computing*, 8:405–409, 1979.
26. N. N. Pisaruk. A fully combinatorial 2-approximation algorithm for precedence-constrained scheduling a single machine to minimize average weighted completion time. *Discrete Applied Mathematics*, 131(3):655–663, 2003.
27. I. Rabinovitch. The dimension of semiorders. *J. Comb. Theory, Ser. A*, 25:50–61, 1978.
28. E. R. Scheinerman and D. H. Ullman. *Fractional Graph Theory*. John Wiley and Sons Inc., 1997.
29. A. S. Schulz. Scheduling to minimize total weighted completion time: Performance guarantees of LP-based heuristics and lower bounds. In *Proceedings of the 5th Conference on Integer Programming and Combinatorial Optimization (IPCO)*, pages 301–315, 1996.
30. P. Schuurman and G. J. Woeginger. Polynomial time approximation algorithms for machine scheduling: ten open problems. *Journal of Scheduling*, 2(5):203–213, 1999.
31. J. Spencer. On minimum scrambling sets of simple orders. *Acta Mathematica*, 22:349–353, 1971.
32. W. T. Trotter. *Combinatorics and Partially Ordered Sets: Dimension Theory*. Johns Hopkins Series in the Mathematical Sciences. The Johns Hopkins University Press, 1992.
33. G. J. Woeginger. On the approximability of average completion time scheduling under precedence constraints. *Discrete Applied Mathematics*, 131(1):237–252, 2003.

A Omitted Proofs

A.1 Proof of Theorem 2

Proof. Let S be an instance of $1|prec|\sum_j w_jC_j$ where precedence constraints are given by a poset $\mathbf{P} = (N, P)$ that admits an efficiently samplable $k{:}t$-realizer $\mathcal{F} = \{L_1, L_2, \ldots, L_t\}$. Furthermore, we assume that $\mathrm{fdim}(\mathbf{P}) \geq 2$. The case when $\mathrm{fdim}(\mathbf{P}) = 1$, i.e., $\mathbf{P}$ is a linear order, is trivial.

Let $V_{\mathbf{P}}$ and $E_{\mathbf{P}}$ be the vertex set and edge set, respectively, of the graph of incomparable pairs $G_{\mathbf{P}}$. Consider the weighted vertex cover VC_S on $G_{\mathbf{P}}$ where each vertex (incomparable pair) $(i, j) \in V_{\mathbf{P}}$ has weight $w_{(i,j)} = p_i \cdot w_j$, as specified in the definition of VC_S (see Section 2.2). Solve the [VC-LP] formulation of VC_S (see Section 3) and let V_i be the set of vertices with value i ($i = 0, \frac{1}{2}, 1$) in the optimum solution. Denote by $G_{\mathbf{P}}[V_{1/2}]$ the subgraph of $G_{\mathbf{P}}$ induced by the vertex set $V_{1/2}$.

We consider the linear extensions of $\mathcal{F}$ as outcomes in a uniform sample space. For an incomparable pair (x, y), the *probability that* y *is over* x *in* $\mathcal{F}$ is given by

$$Prob_{\mathcal{F}}[y > x] = \frac{1}{t}|\{i = 1, \ldots, t : y > x \in L_i\}| \geq \frac{k}{t} \tag{2}$$

The last inequality holds because every incomparable pair is reversed in at least k linear extensions of $\mathcal{F}$.

Let us pick one linear extension L uniformly at random from $\mathcal{F} = \{L_1, \ldots, L_t\}$. Then, by linearity of expectation, the expected value of the independent set $I_{1/2}$, obtained by taking the incomparable pairs in $V_{1/2}$ that are reversed in L, is

$$E[w(I_{1/2})] = \sum_{(i,j) \in V_{1/2}} Prob_{\mathcal{F}}[j > i] \cdot w_{(i,j)} \geq \frac{k}{t} \cdot w(V_{1/2}) \tag{3}$$

A vertex cover solution C for the graph $G_{\mathbf{P}}[V_{1/2}]$ can be obtained by picking the nodes that are not in $I_{1/2}$, namely $C = V_{1/2} \setminus I_{1/2}$. The expected value of this solution is

$$E[w(C)] = w(V_{1/2}) - E[w(I_{1/2})] \leq \left(1 - \frac{k}{t}\right) w(V_{1/2})$$

As observed in [14], $V_1 \cup C$ gives a valid vertex cover for graph $G_{\mathbf{P}}$. Moreover, the expected value of the cover is bounded as follows

$$\begin{aligned} E[w(V_1 \cup C)] &\leq w(V_1) + \left(1 - \frac{k}{t}\right) w(V_{1/2}) && (4)\\ &\leq 2\left(1 - \frac{k}{t}\right)\left(w(V_1) + \frac{1}{2}w(V_{1/2})\right) && (5)\\ &\leq \left(2 - \frac{2k}{t}\right) OPT && (6) \end{aligned}$$

where the last inequality holds since $w(V_1) + \frac{1}{2}w(V_{1/2})$ is the optimal value of [VC-LP]. Note that $t/k \geq \text{fdim}(\mathbf{P}) \geq 2$ was used for the second inequality. Theorem 1 implies that any α-approximation algorithm for VC_S also gives an α-approximation algorithm for S. Thus we obtain a randomized $(2 - 2\frac{k}{t})$-approximation algorithm for S. □

A.2 Proof of Claim 1

This proof is similar to the proof in [1] for proving APX-completeness of vertex cover on cubic graphs.

Proof of Claim. It is easy to see that from every vertex cover $C \subseteq V$ of G we can construct a vertex cover $C' \subseteq V'$ of G' of size exactly $|C| + |V| + |E \setminus E_T|$. In C' we include u_1^i for all $i \in \{i : v_i \in V \setminus C\}$; u_2^i for all $i \in \{i : v_i \in C\}$; e_1^{ij} for each $(v_i, v_j) \in E \setminus E_T$ with $v_i \in V \setminus C$; e_2^{ij} for each $(v_i, v_j) \in E \setminus E_T$ with $v_i \in C$; and every vertex in C.

Given a vertex cover $C' \subseteq V'$ of G' we transform it into a vertex cover $C \subseteq V$ of G in the following manner. Suppose there exists $v_i, v_j \in V$ with $i < j$ such that $\{v_i, v_j\} \in E$ and $v_i \notin C', v_j \notin C'$. Since C' is a feasible vertex cover of G' we have that $\{v_i, v_j\} \in E \setminus E_T$ and either $\{e_1^{ij}, e_2^{ij}, u_1^j\} \subseteq C'$ or $\{e_1^{ij}, u_2^j, u_1^j\} \subseteq C'$. Thus we can obtain a vertex cover $C'' \subseteq V'$ of G' with $|C''| \leq |C'|$ by letting $C'' = (C' \setminus \{u_1^j, e_2^{ij}\}) \cup \{v_j, u_2^j\}$. Repeating this procedure will result in a vertex cover $C''' \subseteq V'$ of G' with $|C'''| \leq |C'|$ such that $C = C''' \cap V$ is a feasible vertex cover of G. Furthermore it is easy to see that $|C| \leq |C'''| - |V| - |E \setminus E_T|$. □

A.3 Proof of Claim 3

Proof of Claim. We relate the two graphs G'_I and G' by the bijection $f : D \to V'$, defined as follows.

$$f((a,b)) = \begin{cases} v_j, & \text{if } (a,b) = (s_i, s_j), \\ u_1^i, & \text{if } (a,b) = (s_i, m_i), \\ u_2^i, & \text{if } (a,b) = (m_i, e_i), \\ e_1^{ij}, & \text{if } (a,b) = (s_i, b_{ij}), \\ e_2^{ij}, & \text{if } (a,b) = (b_{ij}, m_j). \end{cases}$$

Suppose $\{(a,b),(c,d)\} \in E_I$. Since I is an interval order (does not contain any $\mathbf{2}+\mathbf{2}$ structures as induced posets [21,32]) and by the definition of D we have that $b = c$. Now consider the possible cases of $\{(a,b),(b,d)\}$.

($a = s_i, b = s_j, d = s_k, i < j < k$) By construction of I, v_j is the parent of v_k, i.e., $(f((s_i, s_j)), f((s_j, s_k)) = (v_j, v_k) \in E_T \subseteq E'$.

($a = s_i, b = s_j, d = b_{jk}, i < j < k$) Then $f((s_i, s_j)) = v_j$ and $f((s_j, b_{jk})) = e_1^{ij}$ and by definition of G' we have $(v_j, e_1^{jk}) \in E'$.

The remaining cases ($a = s_i, b = s_j, d = m_j, i < j$), ($a = s_i, b = b_{ij}, d = m_j, i < j$), ($a = s_i, b = m_i, d = e_i$), and ($a = b_{ij}, b = m_j, d = e_j, i < j$) are similar to the two above and it is straightforward to check the implication $\{(a,b),(b,d)\} \in E_I \Rightarrow \{f((a,b)), f((b,c))\} \in E'$.

On the other hand, suppose $(a,b) \in E'$ and again consider the different possible cases.

($a = v_i, b = v_j, i < j$) Then v_i is the parent of v_j in T and $f^{-1}(v_i) = (s_k, s_i)$ and $f^{-1}(v_j) = (s_i, s_j)$ for some $k < i < j$. Since s_k's interval representation is completely to the left of s_j's interval representation in I the incomparable pairs (s_k, s_i) and (s_i, s_j) cannot be reversed in the same linear extension, i.e., $\{(s_k, s_i), (s_i, s_j)\} \in E_I$.

($a = v_i, b = e_1^{ij}, i < j$) Then $f^{-1}(v_i) = (s_k, s_i)$ and $f^{-1}(e_1^{ij}) = (s_i, b_{ij})$ for some $k < i < j$. Since s_k's interval representation is completely to the left of b_{ij}'s interval representation in I the incomparable pairs (s_k, s_i) and (s_i, b_{ij}) cannot be reversed in the same linear extension, i.e., $\{(s_k, s_i), (s_i, b_{ij})\} \in E_I$.

The remaining cases ($a = e_1^{ij}, b = e_2^{ij}, i < j$), ($a = e_2^{ij}, b = u_2^j, i < j$), ($a = u_1^j, b = u_2^j, i < j$), and ($a = v_j, b = u_1^j, i < j$) are similar to the two above and omitted.

We have thus proved that $\{(a,b),(b,d)\} \in E_I \Leftrightarrow \{f((a,b)), f((b,c))\} \in E'$, i.e., the function f defines an isomorphism between G'_I and G'. □