Min-Sum Scheduling Under Precedence Constraints*

Andreas S. Schulz[1] and José Verschae[2]

1 **Departments of Mathematics and Economics, TU Munich, Munich, Germany**
andreas.s.schulz@tum.de

2 **Department of Mathematics & School of Engineering, Pontifical Catholic University of Chile, Santiago, Chile**
jverschae@uc.cl

Abstract

In many scheduling situations, it is important to consider non-linear functions of job completions times in the objective. This was already recognized by Smith (1956). Recently, the theory community has begun a thorough study of the resulting problems, mostly on single-machine instances for which all permutations of jobs are feasible. However, a typical feature of many scheduling problems is that some jobs can only be processed after others. In this paper, we give the first approximation algorithms for min-sum scheduling with (nonnegative, non-decreasing) non-linear functions and general precedence constraints. In particular, for $1|\text{prec}|\sum w_j f(C_j)$, we propose a polynomial-time universal algorithm that performs well for all functions f simultaneously. Its approximation guarantee is 2 for all concave functions, at worst. We also provide a (non-universal) polynomial-time algorithm for the more general case $1|\text{prec}|\sum f_j(C_j)$. The performance guarantee is no worse than $2+\epsilon$ for all concave functions. Our results match the best bounds known for the case of linear functions, a widely studied problem, and considerably extend the results for minimizing $\sum w_j f(C_j)$ without precedence constraints.

1998 ACM Subject Classification F.2.2 Sequencing and scheduling

Keywords and phrases scheduling; approximation algorithms; linear programming relaxations; precedence constraints

Digital Object Identifier 10.4230/LIPIcs.ESA.2016.74

1 Introduction

We consider a single-machine scheduling problem with non-linear objective function under precedence constraints. Let $f : \mathbb{R}_+ \to \mathbb{R}_+$ be a non-decreasing cost function. Given a set J of n jobs, where each job $j \in J$ is characterized by a weight $w_j \geq 0$ and an integral processing time $p_j \geq 0$, our goal is to find a sequence of the jobs that minimizes $\sum_j w_j f(C_j)$. Here, C_j denotes the completion time of job j in the corresponding nonpreemptive schedule. Additionally, we consider a set of precedence constraints $P \subseteq J \times J$. Now, $(i, j) \in P$ implies that job i must be completed before j can start. Keeping to the standard three-field notation [11], this problem can be denoted by $1|\text{prec}|\sum w_j f(C_j)$.

Our main result is that there exists a *universal* schedule, i.e., a feasible sequence that depends only on (p_j), (w_j) and P (but not on f), which performs well for all f together.

* This work was partially supported by FONDECYT project 3130407, by Núcleo Milenio Información y Coordinación en Redes ICM/FIC RC130003, and by the Alexander von Humboldt-Foundation.

24th Annual European Symposium on Algorithms (ESA 2016).
Editors: Piotr Sankowski and Christos Zaroliagis; Article No. 74; pp. 74:1–74:13
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

This sequence can be computed in polynomial time, and its performance guarantee is

$$\sup_{C \geq 0} \frac{C \cdot f(C)}{\int_0^C f(t)dt} \ .$$

For concave f, the approximation guarantee can be bounded by 2 with a simple geometric argument. It is worth mentioning that for a vast class of concave functions the approximation guarantee is strictly better than 2, which improves upon the best possible[1] 2-approximation for the linear case.

One somewhat surprising lesson of our study is that an important concept from classic work on the strongly NP-hard problem $1|\text{prec}|\sum w_j C_j$ still holds considerable value for the more general problem $1|\text{prec}|\sum w_j f(C_j)$. In his PhD thesis, Sidney introduced a (polynomial-time computable) decomposition of J into mutually disjoint subsets $J_1, J_2, \ldots, J_k$ and showed that there always exists an optimal solution that follows this order [21]. Chekuri and Motwani [4] as well as Margot, Queyranne and Wang [16] later realized that any feasible sequence that is consistent with Sidney's decomposition is a 2-approximation. This observation was preceded by a number of linear programming based 2-approximation algorithms [19, 12, 6]. Correa and Schulz subsequently proved that virtually all known 2-approximation algorithms are of the Chekuri and Motwani/Margot, Queyranne and Wang type; in fact, several common linear programming relaxations follow Sidney's decomposition [7].

We analyze the same sequence as Chekuri and Motwani/Margot, Queyranne and Wang, but for $1|\text{prec}|\sum w_j f(C_j)$. In contrast to the case of linear f, it is not true that one of the sequences following Sidney's decomposition is optimal, which requires us to devise a very different, more complicated analysis. Still, the algorithm is the same, and our analysis implies that its performance depends on a simple geometric ratio defined by the shape of f. For all concave f, this ratio is at most 2 – the same bound that was earlier observed for linear functions f.

Our main technique relies on analyzing a time-indexed LP relaxation that is intimately related to the *partially ordered knapsack* problem (POK) and its fractional relaxation. In POK we are given a set of items J with weights and values, and a knapsack with a given capacity t. The items have precedence constraints $P \subseteq J \times J$, such that if $(i, j) \in P$ and we pack j into the knapsack, also i must be packed. The total weight of the packing cannot exceed t, and the objective is to maximize the total value. In its fractional version, we are allowed to take fractions of the jobs. If a fraction $x_j \in [0, 1]$ is packed into the knapsack and $(i, j) \in P$, then a fraction of i at least as large as x_j has to be packed. POK and its relaxation was previously studied by Kolliopoulos and Steiner [15], who derived an FPTAS for 2-dimensional precedence constraints and characterized cases where the natural LP relaxation has bounded integrality gap. Our main technical contribution is to show that Sidney's decomposition implies an optimal solution for fractional POK for each t. This implies that the solution is optimal for the time-indexed relaxation, *independently* of the cost function f, which in turn allow us to derive the approximation ratio.

Sidney's decomposition and corresponding algorithm can be viewed as an extension of Smith's optimal WSPT rule for $1||\sum w_j C_j$, which sequences jobs in order of non-increasing ratios of weight to processing time, to $1|\text{prec}|\sum w_j C_j$. In this sense, our work also generalizes earlier contributions by Stiller and Wiese [23] and Höhn and Jacobs [13] for $1||\sum w_j f(C_j)$, to instances with precedence constraints. For arbitrary concave f, Stiller and Wiese showed that Smith's rule guarantees an approximation factor of $(\sqrt{3}+1)/2 \approx 1.366$. Höhn and

[1] Assuming a stronger version of the Unique Games Conjecture [1].

Jacobs built on their analysis to obtain refined and tight bounds of Smith's rule for any specific concave or convex function f. Unlike our LP-based analysis, their techniques rely on identifying a worst-case instance, which can be shown to only contain jobs with the same weight to processing time ratio. The argument then exploits the fact that for concave or convex functions it is easy to identify the worst WSPT solution and the optimal value. For general functions f, Im, Moseley and Pruhs [14] proved that Smith's rule is a $(2+\epsilon)$-speed O(1)-approximate algorithm. Epstein et al. [8] also gave a universal algorithm (quite different from Smith's rule, based on earlier work by Hall et al. [12]), which has performance guarantee $4+\epsilon$, for any cost function f. Additionally, they derived a randomized version of their algorithm with performance guarantee $e+\epsilon$, also for any f. As for non-universal algorithms, Megow and Verschae designed a PTAS for $1| \,|\sum w_j f(C_j)$ [17] for any given function f.

The entire area of min-sum scheduling with non-linear functions of the completion times was arguably revived by Bansal and Pruhs ([2], see also [3]). They considered the more general objective $\sum_j f_j(F_j)$, where f_j is a (nonnegative, non-decreasing) job-dependent cost function, $F_j - r_j$ is the flow time of job j, and r_j its release date. In case $r_j = 0$ for all jobs j, i.e., the setting considered here, their algorithm has performance guarantee 16. Subsequently, a better primal-dual algorithm was given with approximation ratio $4+\epsilon$ [5, 18].

Here, we also give the first polynomial-time approximation algorithm for $1|\text{prec}|\sum f_j(C_j)$, i.e., the case of job-dependent non-linear functions and general precedence constraints; see Section 3. Its approximation guarantee is at most $2+\epsilon$ for concave functions. The algorithm relies on solving a time-indexed LP relaxation and then rounding the fractional solution using randomized α-points. This type of rounding has been extensively used for the sum of weighted completion times objective (see, e.g., [22] for a summary); to the best of our knowledge this is the first time it is applied to a non-linear objective function.

2 A universal algorithm

In this section, we show that the 2-approximation algorithm for the linear case by Margot et al. [16] and Chekuri and Motwani [4] yields the following theorem.

▶ **Theorem 1.** *For any non-decreasing function $f : \mathbb{R}_+ \to \mathbb{R}_+$, the problem $1|prec|\sum w_j f(C_j)$ admits a polynomial-time, purely combinatorial algorithm with approximation guarantee*

$$\Gamma_f := \sup_{C \geq 0} \frac{C \cdot f(C)}{\int_0^C f(t)dt} \;.$$

Moreover, the solution is universal*, that is, it is independent of the cost function f.*

We say that a set of jobs $I \subset J$ is an *ideal* or *initial set* if it is a feasible prefix of a schedule; that is, for any $j \in I$, if $(i,j) \in P$, then $i \in I$. Also, if an ideal I maximizes $w(I)/p(I)$ over all ideals, we say that I is a *density maximal* ideal. Here, $w(I) = \sum_{j \in I} w_j$, and $p(I)$ is defined similarly. We assume that precedence constraints are given by a precedence digraph $G = (J, P)$, with jobs as nodes and arcs given by P. For a given set of jobs K, we denote by G_K the graph induced by set K. A Sidney decomposition is a collection of sets $J_1, J_2, \ldots, J_\ell$ such that J_i is a density maximal ideal in $G_{(J \setminus \cup_{k<i} J_k)}$. The problem of computing a Sidney decomposition can be seen as a parametric network flow problem, and thus it can be computed efficiently [16].

The algorithm computes a Sidney decomposition $J_1, J_2, \ldots, J_\ell$ of set J, and creates a schedule that processes all jobs in the order given by the decomposition; that is, if $j \in J_i, k \in J_s$ and $i < s$, then j is processed before k. Within each set J_i we pick an arbitrary linear ordering of the jobs that is consistent with the precedence constraints. To analyze this algorithm we use a time-indexed linear programming relaxation and find an explicit optimal solution that follows a Sidney decomposition. This is the core of our analysis.

A preemptive relaxation. Unlike in the classic setting, i.e., $1|\text{prec}|\sum w_j C_j$, for non-linear cost functions we cannot necessarily find an optimal solution that follows a Sidney decomposition. Indeed, in the absence of precedence constraints a Sidney decomposition corresponds to a solution given by Smith's rule, and such a solution is not necessarily optimal for non-linear f [13]. However, we will show that for a relaxed version of our problem a Sidney decomposition indeed gives an optimal solution. The relaxation considers preemptive solutions. To model precedence constraints in this relaxation we use the concept of *fractional precedence constraints*; that is, for $(i,j) \in P$ we impose that for any time t the fraction of job i processed in $[0,t]$ is as least as large as that of j.

Let $T = \sum_j p_j$ be the time horizon (assuming, w.l.o.g., that there is no idle-time). In order to avoid discretizing time, or making unnecessary assumptions on the structure of optimal solutions, we describe an arbitrary preemptive solution as a family of non-decreasing piecewise linear functions $x_j : [0,T] \to [0,1]$ for each $j \in J$, where $x_j(t)$ represents the fraction of job j that is processed up to time t. In our relaxation, a job can be split in small pieces, each with the same density $\rho_j = w_j/p_j$ as the original job j. If a small piece of (infinitesimal) processing time δ finishes at time t, in our relaxation this incurs a cost of $\delta \rho_j f(t)$. Equivalently, function x_j increases by $x_j(t+\delta) - x_j(t) = \delta/p_j$ and we incur a cost of $w_j \cdot f(t) \cdot (x_j(t+\delta) - x_j(t))$. With this in mind, we define the cost of a preemptive solution as

$$\sum_{j \in J} w_j \cdot \int_0^T f(t) x_j'(t) dt,$$

where x_j' is the derivative of x_j, which is defined almost everywhere, except for the breakpoints of the function. More formally, our relaxation is given by the following optimization problem.

$$\text{[Rel]} \quad \textsc{Opt}_f = \inf \sum_{j \in J} w_j \cdot \int_0^T f(t) x_j'(t) dt$$

$$\text{s.t. } x_j(T) = 1 \qquad \text{for all } j \in J, \quad (1)$$

$$\sum_{j \in J} x_j(t) p_j \le t \qquad \text{for all } t \in [0,T], \quad (2)$$

$$x_j(t) \ge x_k(t) \qquad \text{for all } (j,k) \in P, t \in [0,T] \quad (3)$$

$$x_j : [0,T] \to \mathbb{R}_+ \qquad \text{non-decreas., piecew. lin. for all } j \in J. \quad (4)$$

We call any solution $(x_j)_{j \in J}$ that satisfies all conditions of this problem a *preemptive* schedule.

Let Opt be the optimal value of our original, non-preemptive, problem. It is not hard to see that the optimal value of this program is a lower bound on Opt. Indeed, let $\mathcal{S}$ be an arbitrary non-preemptive schedule, and let S_j and $C_j = S_j + p_j$ be the starting time and completion time of job j in that solution, respectively. Naturally, we define x_j as a piecewise linear function

$$x_j(t) = \begin{cases} 0 & \text{if } t \le S_j, \\ \frac{t - S_j}{p_j} & \text{if } S_j < t < C_j, \\ 1 & \text{if } t \ge C_j. \end{cases} \quad (5)$$

It is easy to see that this definition yields a feasible solution to [Rel]. Moreover, the objective function can be upper bounded as follows,

$$\sum_{j\in J} w_j \cdot \int_0^T f(t)x_j'(t)dt = \sum_{j\in J} w_j \cdot \int_{S_j}^{C_j} \frac{1}{p_j} f(t)dt \leq \sum_{j\in J} w_j \cdot f(C_j), \tag{6}$$

where the equality holds since x_j has non-zero slope only in (S_j, C_j), where its derivative equals $1/p_j$. The last inequality follows since f is non-decreasing. We conclude that $\textsc{Opt}_f \leq \textsc{Opt}$.

An explicit optimal fractional solution. In what follows we explicitly find an optimal solution to [Rel]. Indeed, let $J_1, \ldots, J_\ell$ be a Sidney decomposition of the instance. For any set J_i we define its *starting time* by $S(J_i) = \sum_{\ell<i} p(J_\ell)$ and its *completion time* as $C(J_i) = S(J_i) + p(J_i)$. For any given $j \in J_i$, we define

$$x_j^*(t) = \begin{cases} 0 & \text{if } t \leq S(J_i), \\ \frac{t-S(J_i)}{p(J_i)} & \text{if } S(J_i) < t < C(J_i), \\ 1 & \text{if } t \geq C(J_i). \end{cases}$$

We will show that this solution is optimal for [Rel]. Notice that x^* does not depend on f, which is why we can derive a universal schedule. Our solution can be interpreted as a schedule in which we first process all jobs in J_1, and within J_1 we process all jobs in a round robin fashion: we first schedule an infinitesimal fraction δ of each job in J_1, the same fraction for each job, ordering the fractions within J_1 according to any feasible linear ordering. With this we are able to process a set of jobs with density $w(J_1)/p(J_1)$, which gives us the most bang for the buck since J_1 is a density maximal ideal. This operation is repeated until J_1 is completely processed. Then we continue for each J_i for $i = 2, \ldots, k$ iteratively with the same strategy. It is straightforward to check that the constructed solution is feasible to [Rel].

To establish that x^* is optimal, we show that it simultaneously solves a family of *fractional partially ordered knapsack problems*. For a given $t \in [0, T]$ we consider the linear program,

$$\begin{aligned} &[\text{FK}(t)] \qquad && \max \quad \sum_{j\in J} x_j w_j && && (7)\\ & && \text{s.t.} \quad \sum_{j\in J} x_j p_j \leq t && && (8)\\ & && x_j \geq x_k && \text{for all } (j,k) \in P, && (9)\\ & && 0 \leq x_j \leq 1 && \text{for all } j \in J. && (10) \end{aligned}$$

We show that, for a fixed t, the vector $(x_j^*(t))_{j\in J}$ yields an optimal solution to [FK(t)]. To do so we first start by characterizing the structure of feasible solutions to this LP. For a given set $S \subseteq J$ we denote by χ^S the indicator vector of set S, that is, $\chi_j^S = 1$ if $j \in S$ and $\chi_j^S = 0$ otherwise.

▶ **Lemma 2** (Fractional Decomposition). *Let z be any feasible solution to [FK(t)]. Then there exist r sets $A_1, \ldots, A_r$ and numbers $1 \geq \gamma_1 > \ldots > \gamma_r > 0$ such that*

1. $z = \sum_{i=1}^r \gamma_i \cdot \chi^{A_i}$,
2. *for all $s \in \{1, \ldots, r\}$ the set $\cup_{i\leq s} A_i$ is an ideal,*
3. $\sum_{i=1}^r \gamma_i p(A_i) \leq t$, *and*
4. $A_i \cap A_\ell = \emptyset$ *for all $i, \ell \in J$ with $i \neq \ell$.*

Proof. Consider the set $Z = \{z_j \neq 0 : j \in J\}$ and assume that $Z = \{\gamma_1, \ldots, \gamma_r\}$ with $\gamma_1 > \gamma_2 > \ldots > \gamma_r > 0$. Define $A_i = \{j : z_j = \gamma_i\}$. Property 4 is obvious. Property 1 follows by 4 and the definition of A_i. Property 3 follows from Property 1 and Inequality (8) of [FK(t)]. To show Property 2, notice that $\cup_{i \leq s} A_i = \{j \in J : z_j \geq \gamma_s\}$. Consider an arbitrary job $k \in \cup_{i \leq s} A_i$, so that $z_k \geq \gamma_s$. We need to show that for any $j \in J$ with $(j,k) \in P$, it holds that $z_j \geq \gamma_s$. This follows because by Inequality (9) we have that $z_j \geq z_k \geq \gamma_s$. ◀

For a given feasible solution z, we call the sets $A_1, \ldots, A_r$ and numbers $\gamma_1, \ldots, \gamma_r$ given by the lemma a *fractional decomposition* of z.

▶ **Lemma 3** (Extremal Fractional Decomposition)**.** *For any feasible solution z to [FK(t)] there exists another feasible solution $z' = \chi^{A_1} + \gamma \cdot \chi^{A_2}$ such that*
1. $\sum_j z_j w_j \leq \sum_j z'_j w_j$,
2. *A_1 and $A_1 \cup A_2$ are ideals,*
3. $p(A_1) + \gamma \cdot p(A_2) \leq t$,
4. $A_1 \cap A_2 = \emptyset$.

Proof. For a given feasible solution z, consider its fractional decomposition given by sets $A'_1, \ldots, A'_r$ and numbers $\gamma_1 > \ldots > \gamma_r$. To show the lemma we write an LP to optimize over the weights γ. We use β_i for the variables representing the weights γ_i.

$$\begin{aligned} \max \quad & \sum_{i=1}^{r} \beta_i w(A'_i) \\ \text{s.t.} \quad & \sum_{i=1}^{r} \beta_i p(A'_i) \leq t, \\ & 1 \geq \beta_1 \geq \ldots \geq \beta_r \geq 0. \end{aligned}$$

Let β^* be an extreme point optimal solution to this LP. Since there are $r+2$ inequalities and r variables, at most 2 inequalities in $1 \geq \beta_1 \geq \ldots \geq \beta_r \geq 0$ are not satisfied with equality. Let us first assume that β^* is not an integral solution, and let ℓ be the smallest index such that $\beta^*_\ell \in (0,1)$. This already induces one strict inequality $\beta^*_\ell < \beta^*_{\ell-1} = 1$. Let $\gamma = \beta^*_\ell$. There must exists an index $u \in \{\ell, \ldots, r\}$ such that $\beta^*_i = \gamma$ for all $i \in \{\ell, \ldots, u\}$ and $\beta^*_i = 0$ for all $i > u$, because otherwise there would be in total 3 strict inequalities. We get the following property: there exist a number $\gamma \in [0,1]$ and two indices $1 \leq \ell \leq u \leq r$ such that $\beta^*_i = 1$ if $i < \ell$, $\beta^*_i = \gamma$ if $i \in \{\ell, \ldots, u\}$, and $\beta^*_i = 0$ if $i > u$. This property holds also if β^* is integral by taking $\gamma = 0$.

Now we can define $A_1 = \cup_{i=1}^{\ell-1} A'_i$, $A_2 = \cup_{i=\ell}^{u} A'_i$ and $\gamma = \beta^*_\ell$. Property 1 follows since β^* is optimal for the LP and setting $\beta_i = \gamma_i$ for all i yields a feasible solution. Properties 2 and 4 hold because of the Fractional Decomposition Lemma. Finally, Property 3 is implied by the first inequality of the LP and since the sets A'_i are pairwise disjoint. ◀

The next lemma shows that, if we are given a density maximal ideal $I \subset J$ such that $t \leq p(I)$, then an optimal solution of [FK(t)] can be constructed by taking each job in I with a fraction of $\frac{t}{p(I)}$. A similar statement was given by Kolliopoulos and Steiner [15], although they used a different proof technique.

▶ **Lemma 4.** *Given any density maximal ideal $I \subseteq J$, then for any $t \leq p(I)$ the vector $x = \frac{t}{p(I)} \chi^I$ is an optimal solution to [FK(t)].*

Proof. Let x be as defined in the statement of the lemma. It is clear that x is a feasible solution to [$FK(t)$]. Consider any optimal solution to [FK(t)] which, by the Extremal

Fractional Decompostion Lemma, can be taken as $\chi^{A_1} + \gamma \cdot \chi^{A_2}$. Recalling that A_1 and $A_1 \cup A_2$ are ideals, and that $A_1 \cap A_2 = \emptyset$, the difference in objective values of the two solutions is given by

$$\begin{aligned}
\frac{t}{p(I)} w(I) - (w(A_1) + \gamma w(A_2)) &= \frac{t}{p(I)} w(I) - (\gamma w(A_1 \cup A_2) + (1-\gamma) w(A_1)) \\
&= \gamma \left(\frac{t}{p(I)} w(I) - w(A_1 \cup A_2) \right) + (1-\gamma) \left(\frac{t}{p(I)} w(I) - w(A_1) \right) \\
&= \gamma \left(\frac{w(I)}{p(I)} t - \frac{w(A_1 \cup A_2)}{p(A_1 \cup A_2)} p(A_1 \cup A_2) \right) + (1-\gamma) \left(\frac{w(I)}{p(I)} t - \frac{w(A_1)}{p(A_1)} p(A_1) \right) \\
&\geq \frac{w(I)}{p(I)} \left\{ \gamma \cdot (t - p(A_1 \cup A_2)) + (1-\gamma) \cdot (t - p(A_1)) \right\} \\
&= \frac{w(I)}{p(I)} \left\{ t - p(A_1) - \gamma p(A_2) \right\} \geq 0,
\end{aligned}$$

where the first inequality follows since I is density maximal, and the last one because $\chi^{A_1} + \gamma \cdot \chi^{A_2}$ is feasible for [FK(t)]. ◀

▶ **Lemma 5.** *For any $t \in [0, T]$, solution $(x_j^*(t))_{j \in J}$ is an optimal solution to [FK(t)].*

Proof. By Lemma 3, there exists an optimal solution z to [FK(t)] such that $z = \chi^{A_1} + \gamma \chi^{A_2}$ where sets A_1 and A_2 satisfy properties 2, 3, and 4 of Lemma 3. Without loss of generality, we assume that $\sum_j z_j p_j = p(A_1) + \gamma p(A_2) = t$, since otherwise we can add to A_2 a dummy job (which does not participate in any precedence constraint) of zero weight and processing time $\frac{1}{\gamma}(t - p(A_1)) - p(A_2)$. We split the proof in two cases.

Case 1: $p(J_1) \geq t$. In this case $x^*(t) = \frac{t}{p(J_1)} \chi^{J_1}$ and thus it is optimal by Lemma 4.

Case 2: $p(J_1) < t$. We modify z to obtain a solution that takes set J_1 integrally. For some number $\gamma' \in [0, 1]$, we will subtract from z the following vector in order to leave space for J_1

$$y = \chi^{A_1 \cap J_1} + \gamma \chi^{A_2 \cap J_1} + \gamma' \chi^{A_1 \setminus J_1} + \gamma' \gamma \chi^{A_2 \setminus J_1}.$$

Notice that if we choose $\gamma' = 0$ then $\sum_{j \in J} p_j y_j = p(A_1 \cap J_1) + \gamma p(A_2 \cap J_1) \leq p(J_1)$, and setting $\gamma' = 1$ we obtain that $\sum_{j \in J} p_j y_j = p(A_1) + \gamma p(A_2) = t > p(J_1)$. Therefore, by continuity we can choose $\gamma' \in [0, 1]$ so that $\sum_{j \in J} p_j y_j = p(J_1)$, that is,

$$\gamma' = \frac{p(J_1 \setminus A_1) - \gamma p(A_2 \cap J_1)}{p(A_1 \setminus J_1) + \gamma p(A_2 \setminus J_1)} \in [0, 1].$$

We first notice that the total weight of y is larger than the weight of χ^{J_1}. To this end, we first show that y is a feasible solution to [FK($p(J_1)$)]. This suffices to conclude that $w(J_1) \geq \sum_{j \in J} w_j y_j$, since χ^{J_1} is optimal to [FK($p(J_1)$)] by Lemma 4. To show that y is feasible to [FK($p(J_1)$)], notice that Inequality (8) is satisfied by our choice of γ'. Let us check (9) for some $(j, k) \in P$. Notice that if $k \notin A_1 \cup A_2$ then $z_k = 0$ and thus the corresponding inequality in (9) is satisfied immediately. Similarly, if $j \notin A_1 \cup A_2$ then, since $A_1 \cup A_2$ is an ideal, $k \notin A_1 \cup A_2$ and thus (9) holds. Thus it suffices to consider $j, k \in A_1 \cup A_2$. Since J_1, A_1, and A_2 are ideals, the only precedence constraints that we need to check are: $(j, k) \in (A_1 \cap J_1) \times (A_2 \cup (A_1 \setminus J_1))$, $(j, k) \in (A_1 \setminus J_1) \times (A_2 \setminus J_1)$, $(j, k) \in (A_2 \cup J_1) \times (A_2 \setminus J_1)$.

For each of these cases, it holds that $y_j \geq y_k$ since $1 \geq \gamma \geq \gamma\gamma'$ and $1 \geq \gamma' \geq \gamma\gamma'$. We conclude that y is feasible for [FK($p(J_1)$)] and thus $w(J_1) \geq \sum_{j\in J} w_j y_j$.

We are now ready to construct a new solution for [FK(t)],

$$z' = z - y + \chi^{J_1} = (1-\gamma')\left(\chi^{A_1\setminus J_1} + \gamma\chi^{A_2\setminus J_1}\right) + \chi^{J_1},$$

has a weight that is less or equal to z. Moreover, we can check that this solution is also feasible for [FK(t)]. Indeed, by construction it holds that $\sum_j p_j z'_j = \sum_j p_j z_j \leq t$. Similarly as before, the only precedence constraints $(j,k) \in P$ that we need to check are when $(j,k) \in J_1 \times ((A_1 \setminus J_1) \cup (A_2 \setminus J_1))$, and $(j,k) \in (A_1 \setminus J_1) \times (A_2 \setminus J_1)$. All of these constraints hold since $1 \geq \gamma \geq \gamma\gamma'$.

We conclude that there exists an optimal solution z' to [FK(t)] that assigns integrally J_1, that is, $z'_j = 1$ for all $j \in J_1$. As well we have that $x^*_j(t) = 1$ for all $j \in J_1$. We can then remove J_1 from our instance and consider a residual problem with precedence graph $G_{J\setminus J_1}$ for a remaining fractional knapsack problem with capacity $t - p(J_1)$. The lemma follows by recursing on this argument. ◀

▶ **Lemma 6.** *The solution $(x^*_j)_{j\in J}$ is an optimal solution for [Rel].*

Proof. Integrating by parts, the objective function of [Rel] can be rewritten as[2]

$$\sum_{j\in J} w_j \cdot \int_0^T f(t)x'_j(t)dt = w(J)\cdot f(T) - \int_0^T \sum_{j\in J} w_j x_j(t) df(t),$$

and hence we can obtain a problem equivalent to [Rel] if we change the objective to maximize $\int_0^T \sum_{j\in J} w_j x_j(t) df(t)$. Since any solution $(x_j(\cdot))_{j\in J}$ for [Rel] defines a feasible solution $(x_j(t))_{j\in J}$ to [FK(t)], and $(x^*_j(t))_{j\in J}$ optimizes [FK(t)], we obtain that $\sum_{j\in J} w_j x_j(t) \leq \sum_{j\in J} w_j x^*_j(t)$. Because f is non-decreasing, then

$$\int_0^T \sum_{j\in J} w_j x_j(t) df \leq \int_0^T \sum_{j\in J} w_j x^*_j(t) df,$$

which helps us to conclude that $(x^*_j(\cdot))_{j\in J}$ is an optimal solution to [Rel]. ◀

We are finally ready to prove Theorem 1.

Proof of Theorem 1. Consider a feasible schedule $\mathcal{S}$ that follows the Sidney decomposition $J_1,\ldots,J_k$. Since for any $j \in J_i$ we have that $C_j \leq C(J_i)$, the cost of that solution can be bounded from above by $\sum_{i=1}^k w(J_i) f(C(J_i))$. On the other hand, the optimal fractional value, attained at solution x^*, can be rewritten as

$$\sum_{j=1}^n w_j \int_0^T f(t)\frac{dx^*_j(t)}{dt}dt = \sum_{i=1}^k w(J_i)\int_{S(J_i)}^{C(J_i)} \frac{f(t)}{p(J_i)}dt,$$

[2] Here the integral taken is the Riemann-Stieltjes integral. This is well defined since f is of bounded variation (since it is non-decreasing) and x_j is continuous. Integration by parts is then valid [10, Theorem 12.14]. Notice that if f were differentiable we could simply write $\int_0^T \sum_{j\in J} w_j x_j(t) f'(t)dt$.

and thus the approximation ratio is at most

$$\frac{\sum_{i=1}^{k} w(J_i) f(C(J_i))}{\sum_{i=1}^{k} w(J_i) \int_{S(J_i)}^{C(J_i)} \frac{1}{p(J_i)} f(t)dt} \leq \max_i \frac{f(C(J_i))}{\int_{S(J_i)}^{C(J_i)} \frac{1}{p(J_i)} f(t)dt} \leq \sup_{0\leq S<C} \frac{f(C)}{\frac{1}{C-S}\int_S^C f(t)dt}$$

$$\leq \sup_{0\leq C} \frac{C\cdot f(C)}{\int_0^C f(t)dt},$$

where the last inequality follows since $\frac{1}{C-S}\int_S^C f(t)dt \geq \frac{1}{C}\int_0^C f(t)dt$, because f is non-decreasing. This shows the theorem. ◀

▶ **Corollary 7.** *There exists a universal solution that, for any concave function f, achieves an approximation guarantee of 2. This solution can be computed in polynomial time.*

Proof. It is enough to notice that if f is concave and non-negative, then $f(t) \geq tf(C)/C$ for any $t \in [0, C]$. Hence, $\int_0^C f(t)dt \geq Cf(C)/2$ and thus $\Gamma_f \leq 2$. ◀

It is worth noticing that the result of Theorem 1 is tight, in the sense that the integrality gap of [Rel] is exactly Γ_f. Indeed, note that for any $C \geq 0$ we can take an instance with one job of processing time C and weight 1. Then the optimal LP solution has a cost of $(1/C)\int_0^C f(t)dt$, whereas the optimal schedule has a cost of $f(C)$.

3 A Rounding Procedure for Job-Dependent Cost Functions

The more general case with objective function $\sum_j f_j(C_j)$ for f_j non-decreasing can also be tackled based on [Rel]. For this we simple generalize the objective function to $\sum_j \int_0^T f_j x_j'$. We call the new relaxation [G-Rel]. In this case we are not able to give an analytic optimal solution for the relaxation. Instead, we discretize the time in the relaxation in order to compute $(1 + \epsilon)$-approximate solutions. Afterwards, we round this solution to obtain a non-preemptive schedule. Notice that this does not yield a universal solution. The approximation guarantee is not the same as in Theorem 1, however it also yields a guarantee of $2 + \epsilon$ for concave functions.

We first show the rounding procedure, which is based on the concept of α-points. Consider a feasible solution $(x_j)_j$ for [G-Rel]. For a given number $\alpha \in [0, 1]$, we define the α-point of job j as $C_j^{\mathrm{LP}}(\alpha) := \min\{t \geq 0 : x_j(t) \geq \alpha\}$, that is, the first point in time in which an α fraction of j is processed. We schedule the jobs in the order of α-points, for some (random) value of α. For simplicity, relabel the jobs so that $C_1^{\mathrm{LP}}(\alpha) \leq \ldots \leq C_n^{\mathrm{LP}}(\alpha)$, and thus the completion time of job j in the algorithm is $C_j^{\mathrm{ALG}} = \sum_{k\leq j} p_j$. The next lemma relates the α-point of a job to its actual completion time. The one thereafter relates the function $C_j^{\mathrm{LP}}(\cdot)$ with the objective function of [G-Rel]. The exposition here takes cues from that in [20].

▶ **Lemma 8** (Goemans [9])**.** *For any $\alpha \in [0, 1]$ and $j \in J$ it holds that*

$$C_j^{\mathrm{ALG}} \leq \frac{1}{\alpha} C_j^{\mathrm{LP}}(\alpha).$$

Proof. Notice that for all $k \leq j$ it holds that $x_k\left(C_j^{\mathrm{LP}}(\alpha)\right) \geq x_k\left(C_k^{\mathrm{LP}}(\alpha)\right) = \alpha$ (since x_k is non-decreasing), and thus

$$C_j^{\mathrm{ALG}} = \sum_{k\leq j} p_k \leq \frac{1}{\alpha}\sum_{k\leq j} p_k x_k(C_j^{\mathrm{LP}}(\alpha)) \leq \frac{1}{\alpha}\sum_{k\in J} p_k x_k(C_j^{\mathrm{LP}}(\alpha)) \leq \frac{1}{\alpha} C_j^{\mathrm{LP}}(\alpha),$$

where the last inequality follows from (2). ◀

▶ **Lemma 9.** *For any j it holds that*

$$\int_0^1 f_j(C_j^{\mathrm{LP}}(\alpha))d\alpha = \int_0^T f_j(t)x_j'(t)dt.$$

Proof. Consider a fixed job j and let $0 = s_1 < s_2 < \ldots < s_\ell = T$ be the breakpoints of the piecewise linear function x_j. Let us also denote by δ_k the derivative of x_j' in (s_k, s_{k+1}), and denote by $\alpha_k = x_j(s_k)$ the fraction of job j processed up to time s_k. Notice that $0 = \alpha_1 \le \alpha_2 \le \ldots \le \alpha_\ell = 1$. Then

$$\int_0^1 f_j(C_j^{\mathrm{LP}}(\alpha))d\alpha = \sum_{k=1}^{\ell-1} \int_{\alpha_k}^{\alpha_{k+1}} f_j(C_j^{\mathrm{LP}}(\alpha))d\alpha$$

Notice that if $\alpha_k < \alpha_{k+1}$, within the interval $\alpha \in (\alpha_k, \alpha_{k+1})$ the function $C_j^{\mathrm{LP}}(\alpha)$ is linear and has a slope of $1/\delta_k$ (observe that $\alpha_k < \alpha_{k+1}$ iff $\delta_k \neq 0$). Hence, using the change of variable $t = C_j^{\mathrm{LP}}(\alpha)$ we obtain that

$$\begin{aligned}\sum_{k=1}^{\ell-1} \int_{\alpha_k}^{\alpha_{k+1}} f_j(C_j^{\mathrm{LP}}(\alpha))d\alpha &= \sum_{k:\alpha_k<\alpha_{k+1}} \int_{\alpha_k}^{\alpha_{k+1}} f_j(C_j^{\mathrm{LP}}(\alpha))d\alpha = \sum_{k:\alpha_k<\alpha_{k+1}} \int_{s_k}^{s_{k+1}} f_j(t)\delta_k dt \\ &= \sum_{k=1}^{\ell-1} \int_{s_k}^{s_{k+1}} f_j(t)\delta_k dt = \int_0^T f_j(t)x_j'(t)dt.\end{aligned}$$

◀

In our algorithm we take α randomly in $[0,1]$ with density 2α.

▶ **Lemma 10.** *Let $\Gamma'_f = \sup_{0\le t\le\tau} \frac{tf(\tau)}{\tau f(t)}$. Taking $\alpha \in [0,1]$ randomly with density 2α yields a solution such that*

$$\mathbb{E}\left(\sum_{j\in J} f_j(C_j^{\mathrm{ALG}})\right) \le 2(\max_{k\in J} \Gamma'_{f_k}) \cdot \sum_{j\in J} \int_0^T f_j x_j'(t)dt.$$

Proof. Due to the Lemma 8,

$$\mathbb{E}(f_j(C_j^{\mathrm{ALG}})) \le \int_0^1 f_j\left(\frac{1}{\alpha} C_j^{\mathrm{LP}}(\alpha)\right) 2\alpha d\alpha.$$

Since for any $0 \le t \le \tau$ it holds that $t \cdot f_j(\tau) \le \Gamma'_{f_j} \cdot \tau \cdot f_j(t)$, we can take $\tau = \frac{1}{\alpha} C_j^{\mathrm{LP}}(\alpha)$ and $t = C_j^{\mathrm{LP}}(\alpha) \le \tau$ and thus $f_j(\frac{1}{\alpha}C_j^{\mathrm{LP}}(\alpha)) \le \Gamma'_{f_j} \cdot \frac{1}{\alpha} \cdot f_j(C_j^{\mathrm{LP}}(\alpha))$ we obtain that

$$\mathbb{E}(f_j(C_j^{\mathrm{ALG}})) \le 2\Gamma'_{f_j} \int_0^1 \frac{1}{\alpha} f_j(C_j^{\mathrm{LP}}(\alpha))\alpha d\alpha = 2(\Gamma'_{f_j}) \cdot \int_0^T f_j(t)x_j'(t)dt.$$

The lemma then follows by summing over j and using linearity of expectations. ◀

Notice that if f_j is concave then $\Gamma'_{f_j} \le 1$, and hence taking x to be optimal for [G-Rel] we would obtain a 2-approximation algorithm.

Moreover generally, if we can compute in polynomial time a solution x that is a $(1+\epsilon)$-approximate solution to [G-Rel], then this lemma shows that the problem admits a $(2(\max_{k\in J} \Gamma'_{f_k})(1+\epsilon))$-approximation algorithm. In what follows we show how to compute such solution x. The proof relies in the classic technique of discretizing the time axis in polynomially many intervals. Similar techniques were used by Bansal and Pruhs [2] and Cheung and Shmoys [5].

In what follows we must assume that we are given an oracle that allows us to query the values $f_j(t)$ for any $t \in \mathbb{N}$ (recall that we assume the processing times to be integral, and thus we only need to query f_j at integral points). Recall also that $T = \sum_j p_j$ is our time horizon. We assume that $f_{\max} := \max_j f_j(T)$ and $f_{\min} := \min\{f_j(t) : j \in J, t \in \{1,\ldots,T\}, f_j(t) > 0\}$ are part of the input, and thus we can manipulate these values in polynomial time. Notice that in any feasible (non-preemptive) schedule the functions f_j get evaluated only at integral points. Hence, without loss of generality, we assume that $f_j(t) = f_j(\lceil t \rceil)$ for all $t \in [0,T]$.

Let us fix a job j now. We partition the time horizon $\{0,1,2,\ldots,T\}$ in consecutive sets. The first set is defined as $I_j^0 = \{t \in \mathbb{N} : f_j(t) = 0\}$, and for any integer $\ell \geq 1$ we define

$$I_j^\ell = \{t \in \mathbb{N} : f_{\min} \cdot (1+\epsilon)^{\ell-1} < f_j(t) \leq f_{\min} \cdot (1+\epsilon)^\ell\}.$$

Notice that the intervals I_j^ℓ for $\ell \in \{0,1,\ldots,\nu\}$ completely cover $\{0,1,\ldots,T\}$ if $\nu = \lceil \log_{1+\epsilon}(f_{\max}/f_{\min}) \rceil$, and thus a polynomial number of interval suffices. Let t_j^ℓ be the largest number in I_j^ℓ and consider the set $\mathcal{T} = \{0,T\} \cup \{t_j^\ell : \text{for all } j \in J, \ell \in \{0,\ldots,\nu\}, t_j^\ell \leq T\}$. Let us relabel the elements in $\mathcal{T} = \{\tau_1,\tau_2,\ldots,\tau_h\}$ such that $0 = \tau_1 < \tau_2 < \ldots < \tau_h$. We remark that $|\mathcal{T}| = h \leq (\nu+1)n + 2$.

With this we define a rounded version of the cost function $\tilde{f}_j$. For any $t \in [0,T]$ the value $\tilde{f}_j(t)$ is defined as $f(\tau_{\ell(j,t)})$, where $\ell(j,t)$ is defined such that $\tau_{\ell(j,t)-1} < t \leq \tau_{\ell(j,t)}$. We obtain that $f_j(t) \leq \tilde{f}_j(t) = f_j(\tau_{\ell(j,t)}) \leq (1+\epsilon) f_j(\lceil t \rceil) = (1+\epsilon) f_j(t)$ for all $t \in [0,T]$. Hence, obtaining an optimal solution [G-Rel] with cost functions $\tilde{f}_j$ yields a $(1+\epsilon)$-approximate solutions for the original cost functions.

▶ **Lemma 11.** *We can compute in polynomial time an optimal solution to [G-Rel] with cost functions $\tilde{f}_j$.*

Proof. Consider any solution x for [G-Rel] with cost functions $\tilde{f}_j$. Consider a new solution $\tilde{x}$ obtained by interpolating the values at $\mathcal{T}$, that is,

$$\tilde{x}_j(0) = x_j(0) \qquad \text{for all } j \in J,$$

$$\tilde{x}_j(t) = \frac{\tau_\ell - t}{\tau_\ell - \tau_{\ell-1}} x_j(\tau_{\ell-1}) + \frac{t - \tau_{\ell-1}}{\tau_\ell - \tau_{\ell-1}} x_j(\tau_\ell) \qquad \text{for all } \ell, t \in (\tau_{\ell-1}, \tau_\ell], j \in J. \tag{11}$$

Since the functions $\tilde{f}_j$ are constant within an interval $(\tau_{\ell-1}, \tau_\ell]$, it is easy to see that the new solution achieves the same objective function, and it is also feasible. Hence, we can restrict [G-Rel] to have solutions as in (11) for all $j \in J$ and $t \in [0,T]$. We can regard this LP as having variables $x_j(\tau_\ell)$ for all $j \in J, \tau_\ell \in \mathcal{T}$. This yields a problem with a polynomial number of variables:

$$|J| \cdot |\mathcal{T}| = nh \leq n((\nu+1)n+2) = O(n^2 \log_{1+\epsilon}(f_{\max}/f_{\min})) = O((n^2/\epsilon)\log(f_{\max}/f_{\min})).$$

Hence, it suffices to argue that we can impose a polynomial number of inequalities that imply the restrictions of the LP. Notice that any solution given by (11) is piecewise linear. The fact that x_j is non-decreasing is equivalent to $x_j(\tau_{\ell-1}) \leq x_j(\tau_\ell)$ for all j,ℓ. Since the value $x_j(t)$ for $t \in (\tau_\ell, \tau_{\ell+1}]$ is a convex combination of $x_j(\tau_\ell)$ and $x_j(\tau_{\ell+1})$, restriction (3) is implied by $x_j(\tau_\ell) \geq x_k(\tau_\ell)$ for all $(j,k) \in P$ and $\ell \in \{1,\ldots,h\}$. For the same reason (2) is implied by

$$\sum_{j \in J} x_j(\tau_\ell) p_j \leq \tau_\ell \qquad \text{for all } \ell \in \{1,\ldots,h\}.$$

Combining all these inequalities yields an equivalent problem of polynomial size. ◀

Collecting our results, we obtain the following theorem.

▶ **Theorem 12.** *For any $\epsilon > 0$, the problem $1|prec|\sum_j f_j(C_j)$ admits a polynomial-time approximation algorithm with approximation factor $(1+\epsilon)\cdot 2\cdot \max_{j\in J}\sup_{t,\tau}\left\{\frac{tf_j(\tau)}{\tau f_j(t)} : 0 \le t \le \tau\right\}$. This implies the existence of a $(2+\epsilon)$-approximation algorithm if f_j is concave for all $j \in J$.*

References

1. N. Bansal and S. Khot. Optimal long code test with one free bit. In *Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2009)*, pages 453–462. IEEE, 2009.
2. N. Bansal and K. Pruhs. The geometry of scheduling. In *Proceedings of the 51th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2010)*, pages 407–414. IEEE, 2010.
3. N. Bansal and K. Pruhs. The geometry of scheduling. *SIAM Journal on Computing*, 43:1684–1698, 2014.
4. C. Chekuri and R. Motwani. Precedence constrained scheduling to minimize sum of weighted completion times on a single machine. *Discrete Applied Mathematics*, 98:29–38, 1999.
5. M. Cheung and D. Shmoys. A primal-dual approximation algorithm for min-sum single-machine scheduling problems. In *Proceedings of the 14th International Workshop on Approximation Algorithms for Combinatorial Optimization Problems (APPROX 2011)*, volume 6845 of *Lecture Notes in Computer Science*, pages 135–146. Springer, 2011.
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*, 30:1005–1021, 2005.
8. L. Epstein, A. Levin, A. Marchetti-Spaccamela, N. Megow, J. Mestre, M. Skutella, and L. Stougie. Universal sequencing on a single machine. *SIAM Journal on Computing*, 41:565–586, 2012.
9. M. X. Goemans. Improved approximation algorithms for scheduling with release dates. In *Proceedings of the 8th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 1997)*, pages 591–598, 1997.
10. R. A. Gordon. *The Integrals of Lebesgue, Denjoy, Perron, and Henstock.* American Mathematical Society, 1994.
11. R. L. Graham, E. L. Lawler, J. K. Lenstra, and A. H. G. Rinnooy Kan. Optimization and approximation in deterministic sequencing and scheduling: A survey. *Annals of Discrete Mathematics*, 5:287–326, 1979.
12. L. A. Hall, A. S. Schulz, D. B. Shmoys, and J. Wein. Scheduling to minimize average completion time: Off-line and on-line approximation algorithms. *Mathematics of Operations Research*, 22:513–544, 1997.
13. W. Höhn and T. Jacobs. On the performance of Smith's rule in single-machine scheduling with nonlinear cost. *ACM Transactions on Algorithms*, 11, 2015.
14. S. Im, B. Moseley, and K. Pruhs. Online scheduling with general cost function. In *Proceedings of the 23rd Annual Symposium on Discrete Algorithms (SODA 2012)*, pages 1254–1265, 2012.
15. S. G. Kolliopoulos and G. Steiner. Partially ordered knapsack and applications to scheduling. *Discrete Applied Mathematics*, 155:889–897, 2007.
16. F. Margot, M. Queyranne, and Y. Wang. Decompositions, network flows, and a precedence constrained single-machine scheduling problem. *Operations Research*, 51:981–992, 2003.

17 N. Megow and J. Verschae. Dual techniques for scheduling on a machine with varying speed. In *Automata, Languages, and Programming (ICALP 2013)*, volume 7965 of *Lecture Notes in Computer Science*, pages 745–756, 2013.

18 J. Mestre and J. Verschae. A 4-approximation for scheduling on a single machine with general cost function. *arXiv:1403.0298*, 2014.

19 A. S. Schulz. Scheduling to minimize total weighted completion time: Performance guarantees of LP-based heuristics and lower bounds. In *Integer Programming and Combinatorial Optimization (proceedings of IPCO V)*, volume 1084 of *Lecture Notes in Computer Science*, pages 301–315, 1996.

20 A. S. Schulz and M. Skutella. Random-based scheduling: New approximations and LP lower bounds. In *Randomization and Approximation Techniques in Computer Science*, volume 1269 of *Lecture Notes in Computer Science*, pages 119–133, 1997.

21 J. B. Sidney. Decomposition algorithms for single machine scheduling with precedence relations and deferral costs. *Operations Research*, 23:283–298, 1975.

22 M. Skutella. List scheduling in order of α-points on a single machine. In E. Bampis, K. Jansen, and C. Kenyon, editors, *Efficient Approximation and Online Algorithms: Recent Progress on Classical Combinatorial Optimization Problems and New Applications*, volume 3484 of *Lecture Notes in Computer Science*, pages 250–291. Springer, 2006.

23 S. Stiller and A. Wiese. Increasing speed scheduling and flow scheduling. In *Proceedings of the 21st Symposium on Algorithms and Computation (ISAAC 2010)*, volume 6507 of *Lecture Notes in Computer Science*, pages 279–290, 2010.