Technische Universität Berlin

FACHBEREICH 3
MATHEMATIK

Scheduling Precedence-Constrained Jobs with Stochastic Processing Times on Parallel Machines

by

Martin Skutella Marc Uetz

No. 691/2000

Scheduling Precedence-Constrained Jobs with Stochastic Processing Times on Parallel Machines

Martin Skutella* Marc Uetz*

(Extended Abstract)

Abstract

We consider parallel machine scheduling problems where the jobs are subject to precedence constraints, and the processing times of jobs are governed by independent probability distributions. The objective is to minimize the weighted sum of job completion times $\sum_j w_j C_j$ in expectation, where $w_j \geq 0$. Building upon an LP-relaxation by Möhring, Schulz, and Uetz (J. ACM 46 (1999), pp. 924–942) and an idle time charging scheme by Chekuri, Motwani, Natarajan, and Stein (SIAM J. Comp., to appear) we derive the first approximation algorithms for this model.

1 Preliminaries

Denote by $V = \{1, \ldots, n\}$ a set of jobs which must be scheduled on m parallel machines. Precedence constraints are given by an acyclic digraph $G = (V, A)$. In the stochastic model, a job processing time p_j is known only upon completion of the job, however, the distribution of the corresponding random variable $\boldsymbol{p}_j$ is given beforehand. Let $\boldsymbol{p} = (\boldsymbol{p}_1, \ldots, \boldsymbol{p}_n)$, and denote by $p = (p_1, \ldots, p_n)$ a particular realization of the processing times. A scheduling policy consists of an online process of decisions which must not anticipate future information; we refer to [2] for details. A given policy eventually yields a feasible m-machine schedule for each realization p. Let $S_j(p)$ and $C_j(p)$ denote the start and completion times of job j for a given realization p, and let $S_j(\boldsymbol{p})$ and $C_j(\boldsymbol{p})$ denote the associated random variables.

2 List scheduling with idle time charging

Given a priority list L, Graham's classical list scheduling algorithm schedules the first *available* job(s) from the list whenever a machine falls idle. Hence, jobs may be scheduled 'out of order' w.r.t. the given list L. For the makespan

* Technische Universität Berlin, Fachbereich Mathematik, Sekr. MA 6–1, Straße des 17. Juni 136, 10623 Berlin, Germany.
Email:{skutella,uetz}@math.tu-berlin.de. The second author is supported by the German-Israeli Foundation for Scientific Research and Development (GIF) under grant I 246-304.02/97.

objective, Graham's list scheduling achieves a performance ratio of $2 - 1/m$, which is also true for the stochastic setting. The weighted completion time objective, however, turns out to be more difficult to approximate. In particular, no approximation result is known for the parallel machine setting with precedence-constraints and stochastic processing times. In order to derive approximation bounds for the weighted completion time objective in deterministic scheduling, a refined list scheduling algorithm has been suggested by Chekuri et al. [1]. The idea is to extend Graham's list scheduling in such a way that a job may be scheduled out of order (w.r.t. the given list L) only if 'enough' idle time has accumulated. The analysis of the algorithm relies on a charging scheme for idle time. We show that an appropriate adaption of the list scheduling algorithm from [1], based on an optimal solution to a generalized LP-relaxation from [3], leads to constant worst-case performance guarantees also for the stochastic model. The algorithm proceeds over time, starting at time $t = 0$, until all jobs have been planed. As usual, a job is called available at time t if all predecessors have already been completed by t, and for a given realization p, the earliest point in time when job j is available in the schedule constructed by Algorithm 1 is denoted by $r_j(p)$.

Algorithm 1 (Delay List [1]).
while *there are unscheduled jobs* **do***: let t be the earliest point in time when a machine falls idle, or the next tentative decision time* (*see case* (c))*, whatever occurs first; let j be the first unscheduled job and i the first unscheduled and available job* (*if any*) *in the list L;*
(a) if *j is available,* **then** *start j at time t and charge all uncharged idle time in the interval $[r_j(p), t]$ to j;*
(b) else, if *there is at least $\beta\, E[\boldsymbol{p}_i]$ uncharged idle time in the interval $[r_i(p), t]$,* **then** *start i at time t and charge all uncharged idle time in $[r_i(p), t]$ to i;*
(c) else *define the next tentative decision time as the first point in time when* (b) *applies to i.*

3 Analysis

We analyze the outcome of Algorithm 1 point-wise, that is, for every realization p of processing times; the analysis is job-by-job. Like in [1], denote by B_j and A_j the set of jobs that come before and after job j in the list L, respectively (by convention, B_j also includes j). For a given realization p, let $O_j(p) \subseteq A_j$ be the set of jobs in A_j that Algorithm 1 starts before job j. The basic idea is to partition the time interval $[0, C_j(p)]$ into two disjoint categories: time intervals I_1 where a chain $j_1, j_2, \ldots, j_h = j$ of predecessors of j is in process (as in Graham's analysis); the total length of this chain is denoted by $\ell_j(p)$. The total processing in the remaining time intervals $[r_k(p), S_k(p)]$, $k = j_1, \ldots, j_h$, which are denoted by I_2, can be partitioned into three categories: processing of jobs in B_j, processing of jobs in $O_j(p)$, and idle time. It follows from the analysis in [1] that any job k is charged no more than $\beta E[\boldsymbol{p}_k]$ idle time. Moreover, there is no uncharged idle time in $[r_k(p), S_k(p)]$, and the idle time in $[r_k(p), S_k(p)]$ is charged only to jobs in B_k. This holds in particular for $k = j_1, \ldots, j_h$. Since $B_{j_1} \subseteq \cdots \subseteq B_{j_h} = B_j$, the total amount of idle time in I_2 is bounded from above by $\beta \sum_{i \in B_j} E[\boldsymbol{p}_i]$. Hence, we obtain for every realization p of processing times

$$C_j(p) \le \ell_j(p) + \tfrac{1}{m}\Big(\sum_{i\in B_j} (p_i + \beta E[\boldsymbol{p}_i]) + \sum_{i\in O_j(p)} p_i\Big). \tag{1}$$

Before we take expectations in (1), we concentrate on the term $\sum_{i\in O_j(p)} p_i$. First, we require:

Lemma 1. $E\left[\sum_{i\in O_j(\boldsymbol{p})} \boldsymbol{p}_i\right] = E\left[\sum_{i\in O_j(\boldsymbol{p})} E[\boldsymbol{p}_i]\right]$.

Proof. We can write $\sum_{i\in O_j(\boldsymbol{p})} \boldsymbol{p}_i$ equivalently as $\sum_{i\in A_j} \delta_i(\boldsymbol{p})\,\boldsymbol{p}_i$, where $\delta_i(\boldsymbol{p})$ is a binary random variable which is 1 if and only if $i \in O_j(p)$. Linearity of expectation yields $E[\sum_{i\in O_j(\boldsymbol{p})} \boldsymbol{p}_i] = \sum_{i\in A_j} E[\delta_i(\boldsymbol{p})\,\boldsymbol{p}_i]$. But $\delta_i(\boldsymbol{p})$ is stochastically independent of the processing time $\boldsymbol{p}_i$ — when job i is started, it is already decided whether $i \in O_j(p)$. In particular, this decision is independent of the actual processing time of job i (processing times are independent). Hence, $\sum_{i\in A_j} E[\delta_i(\boldsymbol{p})\,\boldsymbol{p}_i] = \sum_{i\in A_j} Pr\,(i \in O_j(\boldsymbol{p}))\; E[\boldsymbol{p}_i] = E\big[\sum_{i\in O_j(\boldsymbol{p})} E[\boldsymbol{p}_i]\big]$. □

Next, as in [1], it can be shown that the amount of idle time in $[0, S_j(p)]$ charged to jobs in A_j is bounded by $(m-1)\ell_j(p)$. If a job i is scheduled out of order w.r.t. j (that is, $i \in O_j(p)$), then $\beta E[\boldsymbol{p}_i]$ idle time is charged to i. Hence, we obtain $\beta \sum_{i\in O_j(p)} E[\boldsymbol{p}_i] \le (m-1)\,\ell_j(p)$. Taking expectations in (1), Lemma 1 now yields

$$E[C_j(\boldsymbol{p})] \le \big(1 + \tfrac{m-1}{m\beta}\big) E[\ell_j(\boldsymbol{p})] + \tfrac{1+\beta}{m} \sum_{i\in B_j} E[\boldsymbol{p}_i]. \tag{2}$$

4 Linear programming relaxation

To obtain a priority list L as input for Algorithm 1, Chekuri et al. [1] use a single machine relaxation. This approach does not help in the stochastic setting, since the single machine problem does not necessarily provide a lower bound for the parallel machine problem (see [3] for an example). Instead, we use an LP-relaxation which extends the one proposed in [3] by adding inequalities which represent the precedence constraints. Define $f : 2^V \to \mathbb{R}$ by

$$\begin{aligned} f(W) = \tfrac{1}{2m}\Big(\Big(\sum_{j\in W} E[\boldsymbol{p}_j]\Big)^2 \;+\; \sum_{j\in W} E[\boldsymbol{p}_j]^2\Big) \\ -\; \tfrac{(m-1)(\Delta-1)}{2m}\Big(\sum_{j\in W} E[\boldsymbol{p}_j]^2\Big), \quad W \subseteq V. \end{aligned}$$

Here, $\Delta \ge 0$ is an upper bound on $Var[\boldsymbol{p}_j]/E[\boldsymbol{p}_j]^2$ for all j, where $Var[\boldsymbol{p}_j] = E[\boldsymbol{p}_j^2] - E[\boldsymbol{p}_j]^2$ is the variance of $\boldsymbol{p}_j$. It follows from [3] that the inequalities $\sum_{j\in W} E[\boldsymbol{p}_j]\, E[C_j(\boldsymbol{p})] \ge f(W)$ are valid for all $W \subseteq V$ and any scheduling policy. Hence, the following is a valid LP-relaxation for the problem at hand

$$\begin{aligned} \min \quad & \sum_{j\in V} w_j\, C_j^{\mathrm{LP}} & & \\ \text{s.t.} \quad & \sum_{j\in W} E[\boldsymbol{p}_j]\, C_j^{\mathrm{LP}} \;\ge\; f(W), & W \subseteq V, & \quad (3) \\ & C_j^{\mathrm{LP}} \;\ge\; C_i^{\mathrm{LP}} + E[\boldsymbol{p}_j], & (i,j) \in A. & \quad (4) \end{aligned}$$

We assume that job 1 is artificial with $\boldsymbol{p}_1 \equiv 0$, predecessor of all other jobs, and fixed at time 0, then (4) yields $C_j^{\mathrm{LP}} \geq E[\boldsymbol{p}_j]$ for all jobs j. Since inequalities (3) can be separated in time $\mathrm{O}(n \log n)$, see [3], this LP-relaxation can be solved in polynomial time. From an optimum solution to the LP-relaxation, we define a priority list L according to non-decreasing 'LP completion times' C_j^{LP}. Using Algorithm 1 with input L, we obtain:

Theorem 1. *LP-based list scheduling using algorithm* DELAY LIST *is an α-approximation, where*

$$\alpha = 1 + \tfrac{m-1}{m\beta} + (1+\beta)\big(1 + \max\{1, \tfrac{m-1}{m}\Delta\}\big).$$

Proof. Use (2) and [3, Lemma 4.2] in order to obtain an upper bound on $1/m \sum_{i \in B_j} E[\boldsymbol{p}_i]$ in terms of the LP-completion time C_j^{LP}. The fact that $E[\ell_j(\boldsymbol{p})]$ is a lower bound on the expected completion time of job j for any scheduling policy then yields the desired result. $\square$

Using $\beta = 1/\sqrt{2}$, this yields a constant worst-case performance bound of $\alpha < 5.83$ if $\Delta \leq 1$, which is the case, e.g., for exponentially distributed processing times, or more generally for so-called NBUE distributions (that is, the expected remaining processing time of a job never exceeds its total expected processing time).

5 Final Remarks

The presented results can be slightly improved by a more involved analysis, which also allows to recover the single machine results mentioned in [3]. With minor modifications our results also carry over to problems with release dates. Finally, improved results can be obtained for in-tree precedence constraints using similar ideas as in [1, Sec. 4.4].

References

[1] C. Chekuri, R. Motwani, B. Natarajan, and C. Stein, *Approximation techniques for average completion time scheduling*, to appear in SIAM J. Comp.

[2] R. H. Möhring, F. J. Radermacher, and G. Weiss, *Stochastic scheduling problems I: General strategies*, ZOR – Zeitschrift für Oper. Res. **28** (1984), 193–260.

[3] R. H. Möhring, A. S. Schulz, and M. Uetz, *Approximation in stochastic scheduling: The power of LP-based priority policies*, J. ACM **46** (1999), 924–942.

Reports from the group

“Combinatorial Optimization and Graph Algorithms”

of the Department of Mathematics, TU Berlin

691/2000 *Martin Skutella and Marc Uetz:* Scheduling Precedence Constrained Jobs with Stochastic Processing Times on Parallel Machines

689/2000 *Rolf H. Möhring, Martin Skutella, and Frederik Stork:* Scheduling with AND/OR precedence constraints

685/2000 *Martin Skutella:* Approximating the single source unsplittable min-cost flow problem

684/2000 *Han Hoogeveen, Martin Skutella, and Gerhard J. Woeginger:* Preemptive scheduling with rejection

683/2000 *Martin Skutella:* Convex quadratic and semidefinite programming relaxations in Scheduling

682/2000 *Rolf H. Möhring and Marc Uetz:* Scheduling Scarce Resources in Chemical Engineering

681/2000 *Rolf H. Möhring:* Scheduling under Uncertainty: Optimizing Against a Randomizing Adversary

680/2000 *Rolf H. Möhring, Andreas S. Schulz, Frederik Stork, and Marc Uetz:* Solving Project Scheduling Problems by Minimum Cut Computations (Journal version for the previous Reports 620 and 661)

674/2000 *Esther M. Arkin, Michael A. Bender, Erik D. Demaine, Sándor P. Fekete, Joseph S. B. Mitchell, and Saurabh Sethia:* Optimal Covering Tours with Turn Costs

669/2000 *Michael Naatz:* A Note On a Question of C. D. Savage

667/2000 *Sándor P. Fekete and Henk Meijer:* On Geometric Maximum Weight Cliques

666/2000 *Sándor P. Fekete, Joseph S. B. Mitchell, and Karin Weinbrecht:* On the Continuous Weber and k-Median Problems

664/2000 *Rolf H. Möhring, Andreas S. Schulz, Frederik Stork, and Marc Uetz:* A Note on Scheduling Problems with Irregular Starting Time Costs

661/2000 *Frederik Stork and Marc Uetz:* Resource-Constrained Project Scheduling: From a Lagrangian Relaxation to Competitive Solutions

658/1999 *Olaf Jahn, Rolf H. Möhring, and Andreas S. Schulz:* Optimal Routing of Traffic Flows with Length Restrictions in Networks with Congestion

655/1999 *Michel X. Goemans and Martin Skutella:* Cooperative facility location games

654/1999 *Michel X. Goemans, Maurice Queyranne, Andreas S. Schulz, Martin Skutella, and Yaoguang Wang:* Single Machine Scheduling with Release Dates

653/1999 *Andreas S. Schulz and Martin Skutella:* Scheduling unrelated machines by randomized rounding

646/1999 *Rolf H. Möhring, Martin Skutella, and Frederik Stork:* Forcing Relations for AND/OR Precedence Constraints

640/1999 *Foto Afrati, Evripidis Bampis, Chandra Chekuri, David Karger, Claire Kenyon, Sanjeev Khanna, Ioannis Milis, Maurice Queyranne, Martin Skutella, Cliff Stein, and Maxim Sviridenko:* Approximation Schemes for Minimizing Average Weighted Completion Time with Release Dates

639/1999 *Andreas S. Schulz and Martin Skutella:* The Power of α-Points in Preemptive Single Machine Scheduling

634/1999 *Karsten Weihe, Ulrik Brandes, Annegret Liebers, Matthias Müller–Hannemann, Dorothea Wagner and Thomas Willhalm:* Empirical Design of Geometric Algorithms

633/1999 *Matthias Müller–Hannemann and Karsten Weihe:* On the Discrete Core of Quadrilateral Mesh Refinement

632/1999 *Matthias Müller–Hannemann:* Shelling Hexahedral Complexes for Mesh Generation in CAD

631/1999 *Matthias Müller–Hannemann and Alexander Schwartz:* Implementing Weighted b-Matching Algorithms: Insights from a Computational Study

629/1999 *Martin Skutella:* Convex Quadratic Programming Relaxations for Network Scheduling Problems

628/1999 *Martin Skutella and Gerhard J. Woeginger:* A PTAS for minimizing the total weighted completion time on identical parallel machines

627/1998 *Jens Gustedt:* Specifying Characteristics of Digital Filters with FilterPro

620/1998 *Rolf H. Möhring, Andreas S. Schulz, Frederik Stork, and Marc Uetz:* Resource Constrained Project Scheduling: Computing Lower Bounds by Solving Minimum Cut Problems

619/1998 *Rolf H. Möhring, Martin Oellrich, and Andreas S. Schulz:* Efficient Algorithms for the Minimum-Cost Embedding of Reliable Virtual Private Networks into Telecommunication Networks

618/1998 *Friedrich Eisenbrand and Andreas S. Schulz:* Bounds on the Chvátal Rank of Polytopes in the 0/1-Cube

617/1998 *Andreas S. Schulz and Robert Weismantel:* An Oracle-Polynomial Time Augmentation Algorithm for Integer Proramming

616/1998 *Alexander Bockmayr, Friedrich Eisenbrand, Mark Hartmann, and Andreas S. Schulz:* On the Chvátal Rank of Polytopes in the 0/1 Cube

615/1998 *Ekkehard Köhler and Matthias Kriesell:* Edge-Dominating Trails in AT-free Graphs

613/1998 *Frederik Stork:* A branch and bound algorithm for minimizing expected makespan in stochastic project networks with resource constraints

612/1998 *Rolf H. Möhring and Frederik Stork:* Linear preselective policies for stochastic project scheduling

611/1998 *Rolf H. Möhring and Markus W. Schäffter:* Scheduling series-parallel orders subject to 0/1-communication delays

609/1998 *Arfst Ludwig, Rolf H. Möhring, and Frederik Stork:* A computational study on bounding the makespan distribution in stochastic project networks

605/1998 *Friedrich Eisenbrand:* A Note on the Membership Problem for the Elementary Closure of a Polyhedron

596/1998 *Andreas Fest, Rolf H. Möhring, Frederik Stork, and Marc Uetz:* Resource Constrained Project Scheduling with Time Windows: A Branching Scheme Based on Dynamic Release Dates

595/1998 *Rolf H. Möhring Andreas S. Schulz, and Marc Uetz:* Approximation in Stochastic Scheduling: The Power of LP-based Priority Policies

591/1998 *Matthias Müller–Hannemann and Alexander Schwartz:* Implementing Weighted b-Matching Algorithms: Towards a Flexible Software Design

590/1998 *Stefan Felsner and Jens Gustedt and Michel Morvan:* Interval Reductions and Extensions of Orders: Bijections to Chains in Lattices

584/1998 *Alix Munier, Maurice Queyranne, and Andreas S. Schulz:* Approximation Bounds for a General Class of Precedence Constrained Parallel Machine Scheduling Problems

577/1998 *Martin Skutella:* Semidefinite Relaxations for Parallel Machine Scheduling

Reports may be requested from:

Hannelore Vogt-Möller
Fachbereich Mathematik, MA 6–1
TU Berlin
Straße des 17. Juni 136
D-10623 Berlin – Germany

e-mail: moeller@math.TU-Berlin.DE

Reports are also available in various formats from

`http://www.math.tu-berlin.de/coga/publications/techreports/`

and via anonymous ftp as

`ftp://ftp.math.tu-berlin.de/pub/Preprints/combi/Report-`*number*`-`*year*`.ps`