Models and Algorithms for Stochastic Online Scheduling*

Nicole Megow[1], Marc Uetz[2] and Tjark Vredeveld[2]

[1] Technische Universität Berlin, Institut für Mathematik
Strasse des 17. Juni 136, 10623 Berlin, Germany
nmegow@math.tu-berlin.de
[2] Maastricht University, Department of Quantitative Economics
P.O. Box 616, 6200 MD Maastricht, The Netherlands
{m.uetz, t.vredeveld}@ke.unimaas.nl

Abstract. We introduce a model for non-preemptive scheduling under uncertainty. In this model, we combine the main characteristics of online and stochastic scheduling in a simple and natural way. Job processing times are assumed to be stochastic, but in contrast to traditional stochastic scheduling models, we assume that jobs arrive online, and there is no knowledge about the jobs that will arrive in the future. The particular setting we analyze is parallel machine scheduling, with the objective to minimize the total weighted completion times of jobs. We propose simple, combinatorial online scheduling policies for that model, and derive performance guarantees that match the currently best known performance guarantees for stochastic and online parallel machine scheduling. For processing times that follow NBUE distributions, we improve upon previously best known performance bounds from stochastic scheduling, even though we consider a more general setting.

Keywords. Stochastic scheduling, online optimization, average completion time, policies.

1 Introduction & Model

We consider the classical problem of nonpreemptive scheduling on single and identical parallel machines to minimize the sum of weighted completion times under uncertainty.

In order to cope with uncertainty about the future, there are two major frameworks in the theory of machine scheduling, one is the stochastic scheduling model, the other online scheduling model(s). The main characteristic of *stochastic scheduling*, in contrast to deterministic models, is the fact that the processing times of jobs are subject to random fluctuations, and the actual processing times

* Research partially supported by the DFG Research Center MATHEON "Mathematics for key technologies". A full version of this abstract can be found at http://www.math.tu-berlin.de/~nmegow/muv05sos.pdf.

Dagstuhl Seminar Proceedings 05031
Algorithms for Optimization with Incomplete Information
http://drops.dagstuhl.de/opus/volltexte/2005/110

become known only upon completion of the jobs. It is generally assumed, though, that the respective random variables, or at least their first moments, are known beforehand. In *online scheduling*, the assumption is that the instance is presented to the scheduler only piecewise. Depending on the precise model, jobs are arriving either one-by-one (sequence model), or over time (time-stamp model). The job characteristics such as weight and processing time are usually disclosed upon arrival of the job, and decisions must be made without any knowledge of the jobs to come.

In this paper, we suggest a model that generalizes both, the stochastic scheduling model as well as the online scheduling models. We call it the *Stochastic Online Scheduling (*SOS*)* model. Like in online scheduling, we assume that the instance is presented to the scheduler piecewise, and nothing is known about jobs that might arrive in the future. Once a job arrives, like in stochastic scheduling, we assume that its weight and first moment of its processing time are disclosed, but the actual processing time remains unknown until the job completes.

2 Policies & Performance

We derive worst case performance guarantees for the expected performance of simple, combinatorial online scheduling policies for the stochastic online scheduling model on a single machine and on identical parallel machines, respectively. For the analysis of policies respecting release dates, we restrict ourselves to random variables that we call δ-NBUE. This is a generalization of NBUE random variables.

Definition 1 (δ-NBUE). *A non-negative random variable X is δ-NBUE if, for $\delta \geq 1$,*

$$\mathbb{E}\left[X - t \,|\, X > t\,\right] \;\leq\; \delta\, \mathbb{E}\left[X\right] \qquad \text{for all } t \geq 0.$$

Ordinary NBUE distributions are by definition 1-NBUE.

In our performance analysis, we crucially exploit the fact that lower bounds on the expected value of an optimal policy known from stochastic scheduling carry over to the SOS setting. We utilize the following lower bound by Möhring, Schulz and Uetz [1] on the expected performance $\mathbb{E}\left[Z^{\mathrm{OPT}}\right]$ of an optimal stochastic scheduling policy.

Lemma 1 (Möhring et al. [1]). *For any instance of* $\mathrm{P}|r_j|\mathbb{E}\left[\sum w_j\, C_j\right]$*, we have that*

$$\mathbb{E}\left[Z^{\text{OPT}}\right] \geq \sum_j w_j \sum_{k \in H(j)} \frac{\mathbb{E}\left[P_k\right]}{m} \;-\; \frac{(m-1)(\Delta-1)}{2m} \sum_j w_j \mathbb{E}\left[P_j\right] \,,$$

where Δ bounds the squared coefficient of variation of the processing times, that is, $\mathrm{Var}[P_j]/\mathbb{E}\left[P_j\right]^2 \leq \Delta$ *for all jobs $j = 1, \ldots, n$ and some $\Delta \geq 0$.*

2.1 Stochastic online scheduling on a single machine

For the single machine problem $1|r_j|\mathbb{E}\left[\sum w_j C_j\right]$ in the stochastic online scheduling model we consider the following policy which was proposed for parallel machines by Megow and Schulz in [2] in the deterministic online setting.

> **α-Shift-WSEPT**
> Modify the release date r_j of each job j such that $r'_j = \max\{r_j, \alpha\,\mathbb{E}\left[P_j\right]\}$, for some fixed $\alpha > 0$. At any time t, when the machine is idle, start the job with highest priority in the WSEPT order (i.e., the job with highest ratio of weight to expected processing time) among all available jobs, respecting the modified release dates.

Theorem 1. *The α-*Shift*-WSEPT algorithm is a $(\delta + 2)$-approximation for the* Sos *problem $1|r_j|\mathbb{E}\left[\sum w_j C_j\right]$, for δ-*NBUE *processing times. The best choice of the parameter α is $\alpha = 1$.*

For NBUE distributed processing times this result matches the best known LP based performance bound derived by Möhring et. al [1] for stochastic scheduling. In the online setting, the best possible algorithm is exactly 2-competitive which is shown in [3], [4].

2.2 Stochastic online scheduling on parallel machines

In the parallel machine environment we apply the following simple, combinatorial scheduling policy that we call MinIncrease. We schedule jobs that have been assigned to the same machine in the α-Shift-WSEPT order. The online decisions on job-to-machine assignments are made as follows: as soon as a job is presented, we assign it to that machine where it causes the minimal increase in total expected objective value, given the jobs on each machine would be scheduled in WSEPT order (non-increasing ratios of weight over expected processing times). Note, that in order to assign jobs to machines, we completely ignore release dates and information about real processing times which could be observed from the schedule.

Theorem 2. *Consider the stochastic online scheduling problem* $\mathrm{P}|r_j|\mathbb{E}\left[\sum w_j\, C_j\right]$. *Given that all processing times are δ-NBUE, the* MinIncrease *policy is a ρ–approximation, where*

$$\rho = 1 + \max\{1 + \frac{\delta}{\alpha}\ ,\ \alpha + \delta + \frac{(m-1)(\Delta+1)}{(2m)}\,\}\,.$$

Here, Δ is such that $\mathrm{Var}[P_j]/\mathbb{E}\left[P_j\right]^2 \leq \Delta$ for all jobs j. In particular, since all processing times are δ-NBUE, we know that $\Delta \leq 2\delta - 1$ in the above performance bound.

For NBUE processing times, where we can choose $\Delta = \delta = 1$, the approximation ratio is minimal for $\alpha = (\sqrt{5m^2 - 2m + 1} - m + 1)/(2m)$, obtaining a ratio of less than $(5 + \sqrt{5})/2 - 1/(2m) \approx 3.62 - 1/(2m)$, improving upon the previously best known approximation ratio of $4 - 1/m$ from [1] for the stochastic problem. Moreover, for deterministic instances this performance matches the currently best known competitive ratio of 3.28 from [2] for deterministic online scheduling.

In the special setting where all release dates are equal, our MinIncrease policy indeed chooses for each job j the machine where it causes the least increase in the expected value, given the previously assigned jobs. We prove a performance ratio of

$$1 + \frac{(\Delta + 1)(m - 1)}{2m}$$

which matches exactly the currently best known performance guarantee for the classical stochastic setting, which was derived for the performance of the WSEPT rule in [1]. The WSEPT rule, however, requires the knowledge of all jobs with their weights w_j and expected processing times $\mathbb{E}[P_j]$ at the outset.

Finally, we remark that the MinIncrease policy can be seen as derandomized version of a policy that assign jobs uniformly at random to one of the machines.

References

1. Möhring, R.H., Schulz, A.S., Uetz, M.: Approximation in stochastic scheduling: the power of LP-based priority policies. Journal of the ACM **46** (1999) 924–942
2. Megow, N., Schulz, A.S.: On-line scheduling to minimize average completion time revisited. Operations Research Letters **32(5)** (2004) 485–490
3. Anderson, E.J., Potts, C.N.: On-line scheduling of a single machine to minimize total weighted completion time. Mathematics of Operations Research **29** (2004) 686–697
4. Hoogeveen, H., Vestjens, A.P.A.: Optimal on-line algorithms for single-machine scheduling. In Cunningham, W.H., McCormick, S.T., Queyranne, M., eds.: Proceedings of the Fifth Conference on Integer Programming and Combinatorial Optimization IPCO. Volume 1084 of Lecture Notes in Computer Science., Berlin, Springer (1996) 404–414