Mathematics of Operations Research

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Corrigendum: Greed Works—Online Algorithms for Unrelated Machine Stochastic Scheduling

Varun Gupta , Benjamin Moseley , Marc Uetz , Qiaomin Xie

MATHEMATICS OF OPERATIONS RESEARCH
Vol. 46, No. 3, August 2021, pp. 1230–1234
ISSN 0364-765X (print), ISSN 1526-5471 (online)
http://pubsonline.informs.org/journal/moor

Corrigendum: Greed Works—Online Algorithms for Unrelated Machine Stochastic Scheduling

Varun Gupta,[a] Benjamin Moseley,[b] Marc Uetz,[c] Qiaomin Xie[d]

[a] University of Chicago, Chicago, Illinois 60637; [b] Carnegie Mellon University, Pittsburgh, Pennsylvania 15213; [c] University of Twente, Enschede 7500AE, Netherlands; [d] Cornell University, Ithaca, New York 14850

Contact: varun.gupta@chicagobooth.edu, https://orcid.org/0000-0001-7373-1734 (VG); moseleyb@andrew.cmu.edu, https://orcid.org/0000-0001-8162-017X (BM); m.uetz@utwente.nl, https://orcid.org/0000-0003-4223-2435 (MU); qiaomin.xie@cornell.edu, https://orcid.org/0000-0003-2834-6866 (QX)

Received: July 10, 2020
Accepted: February 11, 2021
Published Online in Articles in Advance: July 16, 2021

MSC2000 Subject Classifications: Primary: 90B36, 68M20; secondary: 90C27, 90C59
OR/MS Subject Classification: Primary: analysis of algorithms: suboptimal algorithms; production/scheduling: stochastic, multiple machine, sequencing: stochastic, approximations/heuristic

https://doi.org/10.1287/moor.2021.1149

Abstract. This corrigendum fixes an incorrect claim in the paper Gupta et al. [Gupta V, Moseley B, Uetz M, Xie Q (2020) Greed works—online algorithms for unrelated machine stochastic scheduling. *Math. Oper. Res.* 45(2):497–516.], which led us to claim a performance guarantee of 6 for a greedy algorithm for deterministic online scheduling with release times on unrelated machines. The result is based on an upper bound on the increase of the objective function value when adding an additional job j to a machine i (Gupta et al., lemma 6). It was pointed out by Sven Jäger from Technische Universität Berlin that this upper bound may fail to hold. We here present a modified greedy algorithm and analysis, which leads to a performance guarantee of 7.216 instead. Correspondingly, also the claimed performance guarantee of $(6 + 3\Delta)h(\Delta)$ in theorem 4 of Gupta et al. for the stochastic online problem has to be corrected. We obtain a performance bound $(7.216 + 3.608\Delta)h(\Delta)$.

In Gupta et al. [1], we claim several results for stochastic online scheduling on unrelated machines to minimize the total weighted completion time $\sum_j w_j C_j$. For the version of the problem where jobs are released over time and become known at individual release times r_j, the analysis of the greedy algorithm as presented in Gupta et al. [1], section 6.1.1, is incorrect, as the upper bound given in lemma 6 in Gupta et al. [1], section 6.1.2, may fail to hold. To understand what the issue is, recall that the greedy algorithm works by modifying release times r_j to r_{ij} when job j is assigned to a machine i, where $r_{ij} := \max\{r_j, c \cdot p_{ij}\}$, and $c \geq 0$ is a parameter that is optimized later in the analysis. The reason why lemma 6 in Gupta et al. [1] may fail to hold is potential low-priority jobs with respect to a given job j—that is, jobs h with $w_h/p_{ih} < w_j/p_{ij}$—that could be released in the open time interval (r_j, r_{ij}). Such jobs could delay the start of job j beyond the upper bound claimed in Gupta et al. [1], lemma 6.

This problem can be fixed by simply accounting for the potential additional delay that such jobs could impose on job j. This can be done because we know *at most one* low-priority job could be in process upon time r_{ij} and impose such additional delay. We therefore suggest a small modification of the greedy algorithm, namely, in the way jobs are assigned to machines in step 3 of the algorithm: Instead of letting this assignment depend on the *actual* increase in the objective function value, we let it depend on an *upper bound* on the increase in the objective function value. This upper bound also includes the potential delay that could be caused by one low-priority job. Technically speaking, in contrast to Gupta et al. [1], we now *define* $\text{cost}(j \to i)$ as an upper bound on the increase of the objective function value when assigning job j to machine i and assign a job to any of the machines minimizing this quantity. This leads to several necessary changes in the subsequent analysis, including a different choice for several of the parameters used in the analysis (e.g., parameter c is chosen to be 2/3 instead of 1). The rest of this corrigendum gives all necessary changes for the analysis.

Greedy Algorithm (Online Time Model for Deterministic Processing Times) Consider any fixed job j that is released at time $t = r_j$ with processing times p_{ij} on machines $i = 1, \ldots, m$. Then, we proceed as follows.

1. Modified release times: On machine i, the release time of job j is modified to $r_{ij} := \max\{r_j, c \cdot p_{ij}\}$; we will optimize parameter $c \geq 0$ later.

2. Let $U_i(t)$ denote the set of jobs that have been assigned to machine i at time t and that have not been started yet (excluding the fixed job j).

3. To decide on the assignment of job j to a machine, we define cost($j \to i$) as an upper bound on the additional cost of job j, when included into a hypothetical greedy weighted shortest processing time (WSPT) schedule of jobs $U_i(r_j)$ on machine i:

$$\mathrm{cost}(j \to i) := w_j \left(\left(1 + \frac{1}{c}\right) r_{ij} + p_{ij} + \sum_{k \in U_i(r_j), \frac{w_k}{p_{ik}} \geq \frac{w_j}{p_{ij}}} p_{ik} \right) + \sum_{k \in U_i(r_j), \frac{w_k}{p_{ik}} < \frac{w_j}{p_{ij}}} w_k p_{ij}. \tag{1}$$

The reason to work with this upper bound is potential jobs that could be released in the interval (r_j, r_{ij}). One of these could delay the earliest possible start time of job j beyond r_{ij}. In defining cost($j \to i$), we account for the maximum additional delay that such jobs could impose on j.

4. Among all machines $i \in \{1, \ldots, m\}$, assign job j to a machine $m(j)$ that minimizes cost($j \to i$), ties broken arbitrarily.

5. On each machine i, schedule jobs following the greedy weighted shortest processing time rule with modified release times r_{ij}. That is, as soon as a machine falls idle at time t, we schedule among all unscheduled jobs k assigned to machine i with $r_{ik} \leq t$, any job j with maximal ratio w_k / p_{ik}.

The following lemma replaces lemma 6 of Gupta et al. [1]:

Lemma 6. *If* ALG *denotes the objective function value of the above greedy algorithm, and if m(j) is the machine to which job j got assigned, then*

$$\mathsf{ALG} \leq \sum_{j \in J} \mathrm{cost}(j \to \mathrm{m}(j)).$$

Proof. Denote by $X_i(t)$ the remaining processing time of a job that is in process on machine i at time t, with $X_i(t) = 0$ if no such job exists. Consider a fixed job j's contribution to the objective $\sum_j w_j C_j$. When job j is released at time r_j, it is assigned to a machine that minimizes cost($j \to i$).

We bound the contribution of this job j to $\sum_j w_j C_j$ as follows: First, we estimate the contribution of job j itself: j can be started no earlier than time r_{ij}, and at time r_{ij}, the machine might be blocked for another $X_i(r_{ij})$ time units by some job h. Note that such job h could even get released later than r_j, namely, in time interval (r_j, r_{ij}). Independent of this, job j's start can be further delayed by "high-priority" jobs k from $U_i(r_j)$, where high priority means that $w_k/p_{ik} \geq w_j/p_{ij}$. Second, the fact that job j got assigned to machine i could delay the "low-priority" jobs from $U_i(r_j)$, where low priority means that $w_k/p_{ik} < w_j/p_{ij}$.

Altogether, the increase of $\sum_j w_j C_j$, caused by job j being assigned to machine i, is at most

$$w_j \left(r_{ij} + X_i(r_{ij}) + \sum_{k \in U_i(r_j), \frac{w_k}{p_{ik}} \geq \frac{w_j}{p_{ij}}} p_{ik} + p_{ij} \right) + \sum_{k \in U_i(r_j), \frac{w_k}{p_{ik}} < \frac{w_j}{p_{ij}}} w_k p_{ij} \leq \mathrm{cost}(j \to i).$$

To see why the last inequality is true, let h be the potential job in process at time r_{ij}, then

$$X_i(r_{ij}) \leq p_{ih} \leq \frac{r_{ih}}{c} \leq \frac{r_{ij}}{c}.$$

The claim of the lemma now follows by summing over all jobs $j \in J$, and because of the following observation: In time interval (r_j, r_{ij}), an even higher priority job k could get released, and such a job k can cause j's start to be delayed even further. But the delay that these jobs will impose on j will be accounted for in the term cost($k \to i$). The set of all low-priority jobs that could get released in interval (r_j, r_{ij}) can cause j's start to be delayed by at most $X_i(r_{ij})$.

Dual Lower Bound

The analysis in Gupta et al. [1] proceeds by defining a linear programming (LP) relaxation for the deterministic version with processing times $\mathbb{E}[P_{ij}]$, which is termed (P$_r$), and finding a lower bound on the optimal solution value z^{P_r} of this LP relaxation by any feasible solution to its dual linear program, which is:

$$\begin{aligned} \max \quad & z^{D_r} = \sum_{j \in J} \alpha_j - \sum_{i \in M} \sum_{s \in \mathbb{Z}_{\geq 0}} \beta_{i,s} \\ \text{s.t.} \quad & \frac{\alpha_j}{\mathbb{E}[P_{ij}]} \leq \beta_{i,s} + w_j \left(\frac{s + \frac{1}{2}}{\mathbb{E}[P_{ij}]} + \frac{1}{2} \right) && \text{for all } i \in M, j \in J, s \in \mathbb{Z}_{\geq r_j}, \\ & \beta_{i,s} \geq 0 && \text{for all } i \in M, s \in \mathbb{Z}_{\geq 0}. \end{aligned} \tag{D$_r$}$$

Presaging the speed-scaling analysis, define

$$\begin{aligned} \alpha_j &:= \mathrm{cost}(j \to m(j)) \\ \beta_{i,s} &:= \sum_{k:m(k)=i;\, r_k \le s; C_k \ge s} w_k, \end{aligned} \tag{2}$$

and their analogues in the modified instance where both the release times and processing times are scaled by a factor $f \ge 1$,

$$\begin{aligned} \alpha_j^f &= \frac{\alpha_j}{f}, \\ \beta_{i,s}^f &= \beta_{i,fs}. \end{aligned} \tag{3}$$

In words, α_j is the previously defined upper bound on the contribution of job j to the objective, and $\beta_{i,s}$ is the total weight of jobs assigned to machine i that were released by time s, but are yet unfinished. The values α_j^f and $\beta_{i,s}^f$ are the corresponding values for an identical instance where time is scaled by factor f. The following lemma replaces lemma 7 of Gupta et al. [1]:

Lemma 7. *With $\boldsymbol{\alpha}^f$ and $\boldsymbol{\beta}^f$ as defined in (3), the values $(\boldsymbol{\alpha}^f/a, \boldsymbol{\beta}^f/b)$ are a feasible solution for the dual (D_r), given that $a \ge 1$, $f \ge 1$, $af \ge 2(2+c)$, $1/c \le f(a-1)$, and $af \ge b$. Specifically, one feasible solution is obtained when $c = 2/3$, $a = 32/23$, $b = 16/3$, and speed $f = 23/6$.*

Proof. Denoting $\mathbb{E}[P_{ij}]$ by p_{ij}, the dual constraints require that, for all jobs j and machines i, and for all times $s \ge r_j$,

$$\frac{\alpha_j}{p_{ij}} \le \beta_{i,s} + w_j \frac{s + \frac{1}{2}}{p_{ij}} + w_j \cdot \frac{1}{2}. \tag{4}$$

Fixing job j and machine i, and plugging in the values α_j^f/a and $\beta_{i,s}^f/b$, we need to show

$$\frac{\alpha_j^f}{a \cdot p_{ij}} \le \frac{\beta_{i,s}^f}{b} + w_j \frac{s + \frac{1}{2}}{p_{ij}} + w_j \cdot \frac{1}{2}, \tag{5}$$

for all $s \ge r_j$. Equivalently, noting that $\alpha^f = \alpha/f$, we have to show that

$$\frac{\alpha_j}{p_{ij}} \le af \cdot \frac{\beta_{i,s}^f}{b} + w_j \frac{s + \frac{1}{2}}{p_{ij}} \cdot af + w_j \cdot \frac{af}{2}. \tag{6}$$

Because $\beta_{i,s}^f = \beta_{i,fs}$, and replacing $s + 1/2$ by s, it therefore suffices to show

$$\frac{\alpha_j}{p_{ij}} \le af \cdot \frac{\beta_{i,fs}}{b} + w_j \frac{s}{p_{ij}} \cdot af + w_j \cdot \frac{af}{2} \tag{7}$$

for all $s \ge r_j$. Because of our choice of α_j as minimizer of $\mathrm{cost}(j \to i)$, we have for all machines i

$$\frac{\alpha_j}{p_{ij}} \le \frac{w_j}{p_{ij}} \cdot \left(\left(1 + \frac{1}{c}\right) r_{ij} + p_{ij} + \sum_{k \in U_i(r_j), \frac{w_k}{p_{ik}} \ge \frac{w_j}{p_{ij}}} p_{ik} \right) + \sum_{k \in U_i(r_j), \frac{w_k}{p_{ik}} < \frac{w_j}{p_{ij}}} w_k. \tag{8}$$

Hence, it suffices to show that the right-hand side in (8) is upper-bounded by the right-hand side in (7). To that end, we even show that a slightly stronger inequality is true: Note that $\beta_{i,fs}$ is the total weight of jobs k assigned to machine i and unfinished at time fs, but with $r_k \le fs$. As long as $f \ge 1$, and because $r_j \le s$, we have $r_j \le fs$. Hence, $\beta_{i,fs} \ge \sum_{k:m(k)=i, r_k \le r_j, C_k \ge fs} w_k \ge \sum_{k \in U_i(r_j), C_k \ge fs} w_k$. Therefore, it suffices to show that the right-hand side of (8) is bounded from above by

$$\begin{aligned} &\frac{af}{b} \cdot \sum_{k \in U_i(r_j), C_k \ge fs} w_k + w_j \frac{s}{p_{ij}} \cdot af + w_j \cdot \frac{af}{2} \\ &= \left(\frac{af}{b} \cdot \sum_{k \in U_i(r_j), C_k \ge fs} w_k + \frac{w_j}{p_{ij}} \cdot (fs - r_j) \right) + \frac{w_j}{p_{ij}} \cdot \left(fs(a-1) + r_j \right) + w_j \cdot \frac{af}{2}. \end{aligned}$$

Multiplying everything with p_{ij}, we therefore need to argue that the following inequality is true:

$$w_j\left(\left(1+\frac{1}{c}\right)r_{ij}+p_{ij}+\sum_{k\in U_i(r_j),\frac{w_k}{p_{ik}}\geq\frac{w_j}{p_{ij}}}p_{ik}\right)+\sum_{k\in U_i(r_j),\frac{w_k}{p_{ik}}<\frac{w_j}{p_{ij}}}w_kp_{ij}$$
$$\leq\left(\frac{af}{b}\cdot\sum_{k\in U_i(r_j):C_k\geq fs}w_kp_{ij}+w_j\cdot\left(fs-r_j\right)\right)+w_j\cdot\left(fs(a-1)+r_j\right)+w_jp_{ij}\cdot\frac{af}{2}.$$

Let us rewrite this more conveniently as

$$\underbrace{w_j\cdot\left(\left(1+\frac{1}{c}\right)r_{ij}+p_{ij}\right)}_{I}+\underbrace{\sum_{k\in U_i(r_j),\frac{w_k}{p_{ik}}\geq\frac{w_j}{p_{ij}}}w_jp_{ik}+\sum_{k\in U_i(r_j),\frac{w_k}{p_{ik}}<\frac{w_j}{p_{ij}}}w_kp_{ij}}_{II}$$
$$\leq\underbrace{\frac{af}{b}\cdot\sum_{k\in U_i(r_j):C_k\geq fs}w_kp_{ij}+w_j\cdot(fs-r_j)}_{II^*}+\underbrace{w_j\cdot\left(\left(fs(a-1)+r_j\right)+p_{ij}\cdot\frac{af}{2}\right)}_{I^*}. \tag{9}$$

The following observations and conditions are sufficient for the above inequality to be true:

1. $I\leq I^*$: Distinguish two cases. When $r_{ij}=r_j$, we have $I=(1+1/c)r_{ij}+p_{ij}=r_j+r_j/c+p_{ij}$. Moreover, since $s\geq r_j$, $I^*=(fs(a-1)+r_j)+p_{ij}\cdot af/2\geq r_j+f(a-1)r_j+p_{ij}\cdot af/2$. Therefore, we get that $I\leq I^*$ under the conditions that $1/c\leq f(a-1)$, and $af\geq 2$. On the other hand, when $r_{ij}=cp_{ij}$, we get $I=(2+c)p_{ij}$, and we get that $I\leq I^*$ under the condition that $2(2+c)\leq af$, whenever $a\geq 1$. Summarizing, we get that $I\leq I^*$ for both cases, conditioned on $1/c\leq f(a-1)$ and $2(2+c)\leq af$.
2. $II\leq II^*$: We have by definition of $U_i(r_j)$ that

$$w_j(fs-r_j)\geq w_j\sum_{k\in U_i(r_j),C_k<fs}p_{ik}.$$

Therefore, under the condition that $af/b\geq 1$, we get that $II\leq II^*$, because then

$$II^*-II\geq\sum_{k\in U_i(r_j),\frac{w_k}{p_{ik}}\geq\frac{w_j}{p_{ij}},C_k\geq fs}(w_kp_{ij}-w_jp_{ik})+\sum_{k\in U_i(r_j),\frac{w_k}{p_{ik}}<\frac{w_j}{p_{ij}},C_k<fs}(w_jp_{ik}-w_kp_{ik})\geq 0.$$

One can now check that parameter values $c=2/3$, $a=32/23$, $b=16/3$, and $f=23/6$ fulfill all necessary requirements.

Main Results

Finally, the following theorem replaces theorem 3 of Gupta et al. [1]. Note that the competitive ratio 7.216 is still an improvement over the best prior algorithm that was known to be 8-competitive (Hall et al. [2]).

Theorem 3. *The greedy algorithm for the deterministic online scheduling problem with release times has competitive ratio 7.216 for minimizing the total weighted completion times* $\sum_j w_jC_j$ *on unrelated machines. That is,* $\mathsf{ALG}\leq 7.216\,\mathsf{OPT}$.

Proof. By definition of $\boldsymbol{\alpha}$ in (2) and by Lemma 6, we have

$$\mathsf{ALG}\leq\sum_j\alpha_j.$$

Moreover, because $\sum_i\beta_{i,s}$ is the total weight of the jobs that have been released by time s, but are yet unfinished, $\sum_{i,s}\beta_{i,s}=\sum_j w_j\,(C_j-r_j)\leq\sum_j w_jC_j$. Therefore, we have

$$\mathsf{ALG}\geq\sum_{i,s}\beta_{i,s}.$$

Consider now a modified problem instance where both the release times and the processing times are scaled by a factor f as follows:

$$r_j^f:=\frac{r_j}{f}\text{ and }p_{ij}^f:=\frac{p_{ij}}{f},$$

so that we also have

$$r_{ij}^f = \frac{r_{ij}}{f}.$$

Then, α_j^f as defined in (3) is the analogous upper bound on the increase in total weighted completion time due to the presence of job j in the modified instance, and $\beta_{i,s}^f$ as defined in (3) is the weight of the released, but yet unfinished, jobs on machine i at time s in the modified instance. Here, we assume without loss of generality that all job sizes and release times are integer multiples of f, which can be achieved by scaling. Also, let us denote by ALG^f the value achieved by the greedy algorithm for the modified instance, and note that $\mathsf{ALG}/f = \mathsf{ALG}^f \le \sum_j \alpha_j^f$, and $\mathsf{ALG}/f = \mathsf{ALG}^f \ge \sum_{i,s} \beta_{i,s}^f$.

By Lemma 7, which gives a lower bound on the optimal solution value OPT via a feasible solution for (D_r) of the form $(\boldsymbol{\alpha}^f/a, \boldsymbol{\beta}^f/b)$ for constants (a, b), we get

$$\mathsf{OPT} \ge \sum_j \frac{\alpha_j^f}{a} - \sum_{i,s} \frac{\beta_{i,s}^f}{b} \ge \frac{\mathsf{ALG}}{f}\left(\frac{1}{a} - \frac{1}{b}\right),$$

or

$$\mathsf{ALG} \le \frac{f \cdot \mathsf{OPT}}{1/a - 1/b}.$$

Now, because parameters $c = 2/3$, $a = 32/23$, $b = 16/3$, and speed $f = 23/6$ are feasible choices for Lemma 7, we get that $\mathsf{ALG} \le (7 + 11/51) \cdot \mathsf{OPT} < 7.216 \cdot \mathsf{OPT}$.

The above rational parameters have been obtained by using a basic nonlinear solver, combined with binary search on one of the parameters. As a matter of fact, optimizing over the four parameters a, b, f, c using SCIP as a solver for nonlinear constrained optimization yields as optimal solution $f/(1/a - 1/b) \approx 7.2151018 < 7 + 11/51$, however, with nonrational solution values. For the sake of simplicity and for feasibility of scaling arguments, we decided to go with the rational solution as presented above.

The proof of theorem 4 of Gupta et al. [1] is unchanged; the new competitive ratio of Theorem 3 implies the following modification to the main result of Gupta et al. [1]:

Theorem 4. *The greedy algorithm has a performance guarantee of* $(7.216 + 3.608\Delta)h(\Delta)$ *for online scheduling of stochastic jobs with release times on unrelated machines to minimize the expectation of the total weighted completion times* $\mathbb{E}\left[\sum_j w_j C_j\right]$. *That is,* $\mathsf{ALG} \le (7.216 + 3.608\Delta)\mathrm{h}(\Delta)\mathsf{OPT}$.

Acknowledgments

The authors are grateful to Sven Jäger from TU Berlin for pointing out the flaw in lemma 6 in the original version of the paper (Gupta et al. [1]). The authors also thank the reviewers of this corrigendum for their careful reading and remarks and for suggesting another bug fix in the proof of Theorem 3.

References

[1] Gupta V, Moseley B, Uetz M, Xie Q (2020) Greed works—online algorithms for unrelated machine stochastic scheduling. *Math. Oper. Res.* 45(2):497–516.

[2] Hall LA, Schulz AS, Shmoys DB, Wein J (1997) Scheduling to minimize average completion time: Off-line and on-line approximation algorithms. *Math. Oper. Res.* 22(3):513–544.