On the Asymptotic Optimality of Work-Conserving Disciplines in Completion Time Minimization

Wenxin Li  
Department of ECE  
The Ohio State University  
wenxinlw.1@gmail.com  
li.7328@osu.edu

Ness Shroff  
Department of ECE and CSE  
The Ohio State University  
shroff.11@osu.edu

Abstract—In this paper, we prove that under mild stochastic assumptions, work-conserving disciplines are asymptotic optimal for minimizing total completion time. As a byproduct of our analysis, we obtain tight upper bound on the competitive ratios of work-conserving disciplines on minimizing the metric of flow time.

Index Terms—completion time, flow time, competitive ratio, asymptotic optimal

I. INTRODUCTION

Minimizing the (weighted) total completion time, one of the most basic performance metric in scheduling theory, has been extensively studied since the 1990s [23], and the earliest study can be traced back to 1950s [26]. Formally, we are given a set of $n$ jobs $N = \{1, 2, \ldots, n\}$, each job has a workload of $p_j$. Let $C_j$ denote the completion time of job $j$, the goal is to find a schedule that minimizes the total (weighted) completion time $\sum_{j \in [n]} C_j$.

The most basic problem in this context is the single machine model with batch arrivals, (i.e., $1||\sum_j C_j$ in the standard 3-field notation\footnote{In the $\alpha||\beta\gamma$-notation, $\alpha$ denotes the machine environment, $\beta$ describes the processing constraints, $\gamma$ denotes the objective to be minimized. Here $\alpha = 1$ represents the single machine environment and $\sum_j C_j$ represents the total completion time objective.} introduced by Graham et al. [13]), which can be solved exactly by the Shortest Processing Time (SPT). There are numerous generalizations of this classic formulation, including the setting with multiple machines, precedence constraints and release dates [4]. Almost all but a few relatively simple variants under consideration are NP-hard, for which various efficient offline approximation algorithms are available [4], [5]. Recently there has been a line of work on improving the approximation guarantee for total weighted completion time objective [3], [17], [25], [20]. The corresponding online setting is also an active area of research, in which jobs arrive online and each job becomes known to the algorithm only after its arrival. For instance, Anderson and Potts [1] considered the problem of minimizing the weighted completion time in the non-preemptive single machine model, and proved that a simple modification of the shortest weighted processing time rule achieves the optimal competitive ratio of two. Shmoys et al. [24] showed how to obtain a $2\rho$-competitive online non-clairvoyant algorithm from an offline $\rho$-approximation algorithm. In a similar flavor, Hall et al. [14] presented a technique for converting a $\rho$-approximation algorithm of the maximum scheduled weighted problem to a $4\rho$-competitive algorithm for completion minimization.

To compare the performance of different disciplines, deterministic models always consider the worst possible input, which does not correspond to any inherent properties of the input. In addition to the aforementioned results in the deterministic setting, there has also been a considerable amount of work on stochastic models. Specifically, there is a line of work that utilizes asymptotic analysis to evaluate system performance in a large scale, often with certain stochastic assumptions on the input data. Chou et al. [7] studied the weighted completion time minimization problem with release dates in single machine model, and proved that the expected weighted completion time under the non-preemptive weighted shortest expected processing time among available jobs (WSEPTA) algorithm is asymptotically optimal when the number of jobs increases to infinity, if job workload and weights are bounded and the job workload are mutually independent random variables. Kaminsky and Simchi-Levi [18] proved the asymptotic optimality of SPT for the total completion time objective in flow shop model, where each job must be sequentially processed on the machines and every job has the same routing.

It is observed that the metric of completion time is less dependent on specific scheduling polices, compared to the flow time (i.e., the time spent by a job starting from its arrival until its completion) objective. In addition, from all the mentioned results above, we can see that although the suggested approaches for completion time optimization are ad-hoc, various different scheduling disciplines all admit desirable performance guarantee. For example, the seminal list-scheduling algorithm [12] achieves a constant gap of two to the optimal, even in the worst case scenario, which is usually overly pessimistic, while WSEPTA and SPT are asymptotically optimal in the stochastic model. Collectively, these observations lead to the question of whether there is a unifying characterization or explanation on the excellent performance of a certain class of scheduling disciplines in different settings.

Our first main result answers this question in affirmative, which is formally stated in Theorem 1, and can be summarized...
in words as,

As long as machines are kept busy whenever possible, the total job completion time are optimum when the number of jobs is sufficiently large.

To show this result, we establish tight competitive ratio bounds for work-conserving disciplines on flow time are established, which is summarized in Table I and Section III.

Related Work. For minimizing the total flow time, Zheng et al. [27], [28] considered the famous Map Reduce setting and proved that with probability one, any work-conserving scheduling algorithms have bounded gap with respect to the optimal algorithm, which holds under bounded job size assumption and stochastic assumptions on the input data.

The most relevant to our work is [6], which provides the optimality condition when the job workloads are upper bounded by a constant, together with the additional assumption that job workload and interarrival time are i.i.d distributed. However, it is natural to allow the maximum job length to be unbounded when the number of jobs increases to infinity. Moreover, the input data cannot assumed to be identical distributed in every situation. Our objective is to provide a deeper understanding of the completion time metric, which is potentially useful to identify disciplines that are both effective (in minimizing completion time) and easy to implement.

A. Main contributions

Asymptotic optimality in minimizing the completion time.

The appeal of our main result is that the assumptions are fundamentally natural and general. We do not require identical distribution assumption or assume any specific distributions on the input.

Theorem 1. Under assumption 4, together with assumption 5 or 6, any work conserving algorithm $\pi$ is almost surely asymptotically optimal for online completion time minimization problem

\[
\lim_{n \to \infty} \frac{\sum_{i \in [n]} C_i^\pi}{\sum_{i \in [n]} C_i} = 1 \quad (\forall \pi \in \Pi_W)
\]

holds almost surely for any input instance, where $\Pi_W$ denotes the class of work-conserving disciplines.

In addition, our result can be further applied in the following settings.

- **Finite number of interjob precedence phase.** Interjob precedence constraint $i \rightarrow j$ implies that job $i$ must be finished before we start to process job $j$.
- **Multitask job with arbitrary intertask precedence constraint.** Each job consists of multiple tasks, the job is considered to be completed only when all its tasks are finished. The precedence constraints between tasks within the same job can be arbitrary.

Tight competitive ratio bound for flow time minimization.

To the best of our knowledge, this is the first tight characterization on the worst case performance of work-conserving scheduling algorithms. In Theorem 7 we show that the total flow time under any work-conserving algorithm is always no more than $2B$ times that under the optimal algorithm. Here parameter $B = \frac{p_{\text{max}}}{p_{\text{min}}}$ represents the ratio of the maximum to the minimum job workload. In addition, this competitive ratio upper bound is shown to be tight up to a constant. On the negative side, no non-preemptive scheduling algorithm can achieve a competitive ratio better than $B^{1/2}$ for any given constant $\varepsilon > 0$. Together with the well-known competitive ratio lower bound for preemptive scenario, our result is summarized in Table I.

The rest of this paper is organized as follows. Section II describes model and definitions. We prove the competitive ratio bound and optimality condition in Section III and Section IV respectively. Possible generalizations and numerical results are given in Section V and VI. The paper in concluded in Section VII.

II. Model and Definitions

In this paper, we consider minimizing the completion time in the multiple-machine environment. There are $n$ jobs and a set of $m$ identical machines in the system. Each job $i$ is assigned a processing time $p_i$ and arrival time $r_i$, and we use $\Delta r_i = r_i - r_{i-1}$ to denote the interarrival time between job $i - 1$ and $i$. We focus on work-conserving disciplines, which is formally defined as following.

Definition 2 (Work-conserving scheduling discipline [15]). A scheduling discipline $\pi$ is called work-conserving if it never idles machines when there exists at least one feasible job or task awaiting the execution in the system. Here a job or task is called feasible, if it satisfies all the given constraints of the system (e.g. precedence constraints, preemptive and non-preemptive constraints, etc).

Our result in this paper holds for both preemptive and non-preemptive models. In the preemptive model, the job that is running can be interrupted and later continued on any machine, while the system must follow the “run to completion” rule in the non-preemptive setting. For any scheduling discipline $\pi$, we compare it with an oblivious adversary, i.e., the optimal offline algorithm, for which there are no restrictions, it can have full knowledge of the input sequence in advance, together with the choices of $\pi$.

Definition 3 (Competitive Ratio). Let $CR_\pi$ denote the competitive ratio of discipline $\pi$. It is defined as

\[
CR_\pi = \max_I \frac{G_\pi(I)}{G_{\pi^*}(I)},
\]

where we use $G_\pi(I)$ to denote the objective value under discipline $\pi$ and instance $I$. 
TABLE I: Summarization of worst case performance of work-conserving algorithms

| Scenario        | Competitive ratio | Supremum | Infimum |
|-----------------|-------------------|----------|---------|
| Preemptive      | Θ(B)              | (Theorem 9) | 1       |
| Single machine  |                   |          |         |
| Multiple machine|                   |          |         |
| Non-preemptive  | Θ(B)              | (Theorem 9) | Ω(B^(k+1)) |

A. Assumptions

In this paper we use \(\mu_p^{(k)}, \mu_i^{(k)}\) to denote the expected value of the \(k\)-th job workload and interarrival time, respectively. The assumptions utilized in this paper are stated as following.

**Assumption 4.** Job workload \(\{p_i\}_{i \in [n]}\) and interarrival time \(\{\Delta t_i\}_{i \in [n]}\) are independently distributed, and the \((2 + \epsilon)\)-th moment of job workload and the second moment of interarrival time is finite.

**Assumption 5.** \(\rho^{(n)} \leq 1 + \alpha(n^{-1/2})\), where \(\rho^{(n)}\) is defined as

\[\rho^{(n)} = \sup_{t \in [n]} \mu_p^{(l)} \cdot \mu_i^{(k)}\]

When the job workload and interarrival time are i.i.d distributed, \(\rho = \frac{\mu_p}{\mu_i}\) represents the traffic intensity.

**Assumption 6.** The stochastic system driven by the arrival and service processes is stable and one of the following conditions holds,

- The mean values of interarrival time are almost identical, i.e., \(|\mu_i^{(k)} - \mu_i^{(j)}| = \alpha(n^{-1/2}) (\forall i, j \in [n])\)
- \(\{\mu_i^{(k)}\}\) is non-decreasing, i.e., \(\mu_i^{(k)} \leq \mu_i^{(k+1)}\).

A special case is when the mean values of interarrival time are identical, i.e., there exists \(\mu_r\) such that \(\mu_i^{(k)} = \mu_r\) for \(\forall k \in [n]\).

III. TIGHT COMPETITIVE RATIO BOUND ON FLOW TIME

We establish a tight characterization on the performance of work-conserving disciplines on flow time in this section.

A. Upper bound

**Theorem 7.** The competitive ratio of any work-conserving scheduling discipline is no more than \(2B\).

**Proof.** In the following proof, we use \(W_\pi(t)\) to represent the remaining workload under discipline \(\pi\) at time \(t\), and let \(\pi^*\) denote the optimal scheduling discipline. The main idea of our proof is to relate \(n_\pi(t)\), the number of jobs alive under \(\pi\) to that under the optimal discipline \(\pi^*\), which is achieved by comparing the amount of unfinished workload under these two disciplines. The bounded job size ratio parameter allows us to convert the relation between remaining workload to that between the number of unfinished jobs.

To start with, observe that

\[n_\pi(t) \leq \frac{W_\pi(t)}{p_{\text{min}}}(2)\]

\[= B \cdot \frac{W_{\pi^*}(t) + (W_\pi(t) - W_{\pi^*}(t))}{p_{\text{max}}}\]

\[\leq B \left( n_{\pi^*}(t) + \frac{W_\pi(t) - W_{\pi^*}(t)}{p_{\text{max}}} \right), (\forall t \geq 0) (3)\]

where the equality follows from the definition of \(B\) and \(p_{\text{min}}\);

The last inequality is due to the fact that \(n_{\pi^*}(t) \geq \frac{W_{\pi^*}(t)}{p_{\text{max}}}\). In the following we let

\[\mathcal{I} = \left\{ t \mid n_{\pi^*}(t) + n_\pi(t) > 0 \right\}\]

be the set of non-trivial time slots, i.e., in which unfinished jobs exist either under \(\pi\) or \(\pi^*\), and let \(\mathcal{I}_\pi^*\) denote the collection of time slots in which idle machines exist under discipline \(\pi\). We argue that \(W_\pi(t)\) is no more than \((m - 1) \cdot p_{\text{max}}\) for \(\forall t \in \mathcal{I}_\pi^*\). To show this fact, note that \(\pi\) is a work-conserving algorithm, which implies that there are less than \(m\) unfinished jobs at time \(t\) due to the existence of idle machines. As a consequence, we have

\[\frac{W_\pi(t) - W_{\pi^*}(t)}{p_{\text{max}}} \leq m - 1, (\forall t \in \mathcal{I}_\pi^*) (4)\]

which follows from the non-negativity of \(W_{\pi^*}(t)\). On the other hand, we claim that bound (4) still holds for \(t \in \mathcal{I} \setminus \mathcal{I}_\pi^*\), i.e., when all the machines are busy under \(\pi\). To see this fact, for each \(t \in \mathcal{I} \setminus \mathcal{I}_\pi^*\), we define its related time slot in \(\mathcal{I}_\pi^*\) as,

\[t^2 = \max \left\{ \bar{t} \mid \bar{t} \in \mathcal{I}_\pi^* \cap [0, t] \right\}\]

we next claim that for \(\forall t \geq 0\),

\[W_\pi(t) - W_{\pi^*}(t) \leq W_\pi(t^2) - W_{\pi^*}(t^2) \leq (m - 1) \cdot p_{\text{max}}. \]

(5)

This is because the remaining workload under \(\pi\) deceases at the maximum speed of \(m\) during time interval \((t^2, t]\), hence the difference of the remaining workload between \(\pi\) and \(\pi^*\) must be non-increasing in \((t^2, t]\). Combining (3) and (5), we have

\[n_\pi(t) \leq B \cdot \left( n_{\pi^*}(t) + m - 1 \right), (\forall t \geq 0). \]

(6)

Now we are ready to bound the total flow time of \(\pi\) as following,

\[F_\pi = \int_{t \in \mathcal{I}} n_{\pi}(t)dt \]

(7)

\[= \int_{t \in \mathcal{I}_\pi^*} n_{\pi}(t)dt + \int_{t \in \mathcal{I} \setminus \mathcal{I}_\pi^*} n_{\pi}(t)dt\]

(a)

\[\leq \int_{t \in \mathcal{I}_\pi^*} n_{\pi}(t)dt + \int_{t \in \mathcal{I} \setminus \mathcal{I}_\pi^*} B \cdot \left( n_{\pi^*}(t) + m - 1 \right)dt\]

(b)

\[\leq B \cdot \left[ \left( \int_{t \in \mathcal{I}_\pi^*} n_{\pi}(t)dt + \int_{t \in \mathcal{I} \setminus \mathcal{I}_\pi^*} (m - 1)dt \right) + F_{\pi^*} \right] \]

(c)

\[\leq 2B \cdot F_{\pi^*}, \]

(8)
time (SRPT) discipline, which always serves the job with shortest remaining processing time, minimizes the number of jobs in the system in single machine model. Note that for multiple machines, it is impossible to achieve a competitive ratio of $o(\log B)$ [19], for either work-conserving or non-work-conserving disciplines. For non-preemptive case, similar to the proof in [19], a lower bound of $B^{1-\varepsilon}$ on the competitive ratios of work-conserving disciplines can be shown, via reduction to the numerical three-dimensional matching (N3DM) problem.

To summarize, we have the following lemma.

Lemma 10. Let $W_p$ and $W_p^*$ be the collection of preemptive and non-preemptive work-conserving disciplines. The infimum of preemptive work-conserving disciplines satisfies

$$\inf_{\pi \in W_p} CR_{\pi} = \begin{cases} 1 & m = 1 \\ \log B & m \geq 2 \end{cases}$$

The competitive ratio infimum of work-conserving scheduling algorithms satisfies that $\inf_{\pi \in W_p^*} CR_{\pi} \geq B^{1-\varepsilon}$ for arbitrary positive number $\varepsilon > 0$, unless $P \in \text{NP}$. Consequently we have $CR_{\pi} \in [B^{1-\varepsilon}, 2B]$ for any $\pi \in W_p^*$.

Discussions. For the more restricted class of non-size based scheduling algorithms (i.e., algorithms that do not make use of job sizes), the following competitive ratio lower bound has been established in [22]. Combined with Theorem 7, we can further obtain Theorem 12.

Fact 11 ([22]). No (deterministic) non-size based scheduling algorithm can achieve a competitive ratio that is less then $B$.

Theorem 12. All non-size based and work-conserving scheduling algorithms achieve the same competitive ratio (up to a constant of two). More specifically, $CR_{\pi} \in [B, 2B]$ holds for all $\pi \in \Pi^*$. Our conclusion can be regarded as the worst case counterpart of the well-known result for $M/G/1$ queue, which is summarized in the following Fact 13.

Fact 13. For the case of an $M/G/1$ queue, all non-preemptive and work-conserving service orders that do not make use of job sizes have the same distribution of the number of jobs in the system [8], [15].

Compared with Fact 13, it is also important to point out that Theorem 12 holds in the more general setting of multiple servers and arbitrary arrival distribution. In addition, our result indeed indicates that in the worst case, both preemptive and non-preemptive non-size-based scheduling algorithms achieve almost identical competitive ratio, while Fact 13 only applies for non-preemptive algorithms.

IV. OPTIMALITY IN STOCHASTIC ONLINE COMPLETION TIME MINIMIZATION

In this section, we show the asymptotic optimality condition, utilizing our tight characterization on the worst case performance of work conserving algorithms with respect to
the metric of flow time. We first state the following fact that will be useful for establishing Theorem 1.

**Lemma 14.** For a sequence of random variable sequence \( \{X_i\}_{i \in [n]} \), the equation
\[
\lim_{n \to \infty} \frac{\max_{i \in [n]} X_i}{n^{1/r}} = 0
\]
holds almost surely, under one of the following conditions:
- \( \{X_i\}_{i \in [n]} \) are i.i.d and \( \mathbb{E}[X_1^r] < \infty \), i.e., the \( r \)-th moment of \( X_i \) is finite.
- \( \mathbb{E}[X_1^r] < \infty \) holds for some \( \epsilon > 0 \), i.e., the \( (r + \epsilon) \)-th moment of \( X_i \) is finite.

In addition, \( \mathbb{P}(\max_{i \in [n]} X_i \leq n^{1/r}) = 1 \) when \( \{X_i\}_{i \in [n]} \) are i.i.d and \( \mathbb{E}[X_1] = \infty \).

**Proof.** The proof of the i.i.d distributed case mainly relies on the Borel-Cantelli Lemma, which is a simple consequence of [10], here we provide a direct and simpler proof. Proof of the general case simply utilizes the Markov inequality. The details are deferred to [21]. \( \square \)

We next make the following observation.

**Observation 15.** The asymptotic optimal condition holds for \( \forall \pi \in \Pi_{1/2} \), if the total flow time under the optimal scheduling algorithm satisfies that \( \sum_{i \in [n]} f_i^* = o(n^2/B(n)) \), where \( B(n) = \max_{i \in [n]} p_i \) represents the job size ratio.

**Proof.** See [21]. \( \square \)

**Remark.** We derive an \( \Omega(n^2) \) lower bound on optimal total completion time in the proof of Observation 15, which still holds without any assumptions on the arrival process, if there is a lower bound \( \Delta = \Theta(1) \) on job workload. To see this, we re-index the jobs by their completion time order as \( C_{\sigma_k} \leq C_{\sigma_{k+1}} \) \( \forall k \in [n-1] \), then we have \( C_{\sigma_k} \geq \frac{\sum_{i \in [n]} p_{\sigma_i}}{m} \) and
\[
\sum_{k \in [n]} C_k \geq \frac{1}{m} \sum_{k \in [n]} \sum_{i \in [k]} p_{\sigma_i} \geq \frac{(n+1)n}{2m} \cdot \Delta = \Omega(n^2).
\]

**A. Lower bound on the optimal flow time**

**Lemma 16.** For a single server system with interarrival time \( \{\Delta r_k\}_{k \in [n]} \) and job workload \( \{p_k\}_{k \in [n]} \),
\[
\mathbb{P}\left( \lim_{n \to \infty} \frac{\sum_{k \in [n]} f_i^{*}}{n^{1/2+\epsilon}} = 0 \right) = 1,
\]
for any \( \epsilon > 0 \), if for \( \mu_{v}^{(k)} = \mu_{r}^{(k)} = \mu_{p}^{(k)} \),
\[
\frac{\sum_{k \in [n]} \mu_{v}^{(k)} \cdot I_{\mu_{v}^{(k)} > 0}}{n} = o(n^{-1/2+\epsilon}).
\]

**Proof.** In this proof, we let \( W_k \) denote the waiting time of the \( k \)-th arriving job under first come first serve (FCFS) discipline. For general input job workload and arrival time distributions, we have the following recursion according to Lindley equation [2].
\[
W_{k+1} = (W_k + p_k - \Delta r_k)^+ \equiv (W_k + v_k)^+,
\]
where we let \( v_k = p_k - \Delta r_k, \mu_{v}^{(k)} = \mathbb{E}[v_k] = \mu_{p}^{(k)} - \mu_{r}^{(k)} \) for \( \forall k \in [n] \) and \( x^+ = \max\{x, 0\} \) for \( \forall x \in \mathbb{R} \). Solving the recursive equation, it can be shown that [2],
\[
W_n = \max\left\{ T_n, T_n - T_1, T_n - T_2, \ldots, T_n - T_{n-1}, 0 \right\}
= T_n - \min_{k \in [n]} T_k,
\]
where \( T_k = \sum_{i \in [k]} v_i \) \( \forall k \in [n] \). Hence the total waiting time under FCFS is,
\[
\sum_{k \in [n]} W_k = \sum_{k \in [n]} \left[ T_k - \min_{i \in [k]} T_i \right],
= \sum_{k \in [n]} \left[ T_k - \sum_{i \in [k]} \mu_{v}^{(i)} \right]
+ \sum_{k \in [n]} \left[ \max_{i \in [k]} \{ -T_i \} - \sum_{i \in [k]} (\mu_{v}^{(i)}) \right],
\]
Note that
\[
\frac{\Sigma_1}{n^{3/2+\epsilon}} = \frac{\sum_{k \in [n]} \left\{ T_k - \sum_{i \in [k]} \mu_{v}^{(i)} \right\}}{n^{3/2+\epsilon}}
= \sum_{k \in [n]} \left( \frac{1}{n^{1/2+\epsilon}} - \frac{k-1}{n^{3/2+\epsilon}} \right)(v_k - \mu_{v}^{(k)}).
\]
Applying Chebyshev inequality, we know that for any \( \epsilon > 0 \),
\[
\lim_{n \to \infty} \mathbb{P}\left( \frac{\Sigma_1}{n^{3/2+\epsilon}} \geq \epsilon \right)
= \lim_{n \to \infty} \frac{\text{var}(\sum_{k \in [n]} \left( \frac{1}{n^{1/2+\epsilon}} - \frac{k-1}{n^{3/2+\epsilon}} \right)(v_k - \mu_{v}^{(k)}))}{\epsilon^2}
\leq \lim_{n \to \infty} \sum_{k \in [n]} \text{var}(v_k - \mu_{v}^{(k)})
= \lim_{n \to \infty} \sum_{k \in [n]} \left( \sigma_r^{(k)} + \sigma_p^{(k)} \right)
= \frac{\sum_{k \in [n]} \sigma_r^{(k)} + \sigma_p^{(k)}}{n^{1+2\epsilon}} = 0,
\]
where the last inequality holds as \( \sup_{n} \sigma_r^{(n)} \sigma_p^{(n)} \) \( \infty \).
Hence \( \lim_{n \to \infty} \frac{\Sigma_1}{n^{3/2+\epsilon}} = 0 \) almost surely.

On the other hand, we have the following inequality for \( \lambda = \Theta(n^{1/2+\epsilon}) \),
\[
\mathbb{P}\left( \Sigma_2 \leq \lambda n \right)
= \mathbb{P}\left( \sum_{k \in [n]} \left\{ \max_{i \in [k]} \{ -T_i \} - \sum_{i \in [k]} (\mu_{v}^{(i)}) \right\} \leq \lambda n \right)
\geq \mathbb{P}\left( \max_{i \in [k]} \{ -T_i \} - \sum_{i \in [k]} (\mu_{v}^{(i)}) \leq \lambda, \forall k \in [n] \right)
= \mathbb{P}\left( - T_i \leq \lambda \sum_{j \in [k]} (\mu_{v}^{(j)}) \right) = \mathbb{P}\left( - T_i \leq \lambda \sum_{j \in [k]} (\mu_{v}^{(j)}) \cdot I_{\mu_{v}^{(j)} < 0}, \forall i \in [k], k \in [n] \right),
\]

where \( \lambda_k = \lambda - \sum_{j \in [k]} \mu_v^{(j)} \cdot \mathbb{1}_{\mu_v^{(j)} > 0} \) (\( \forall k \in [n] \)). Combining with the facts that \( \lambda_k \leq \lambda' = \lambda - \sum_{j \in [n]} \mu_v^{(j)} \cdot \mathbb{1}_{\mu_v^{(j)} > 0} \) for \( \forall k \in [n], \) and \( \sum_{i \in [k]} (-\mu_v^{(i)} \cdot \mathbb{1}_{\mu_v^{(i)} < 0}) \) is non-decreasing with respect to the index \( k, \) we further have
\[
\begin{align*}
\mathbb{P}(\Sigma_2 \leq \lambda n) & \geq \mathbb{P} \left( -T_k \leq \lambda' + \sum_{i \in [k]} (-\mu_v^{(i)} \cdot \mathbb{1}_{\mu_v^{(i)} < 0}), \forall k \in [n] \right) \\
& = \mathbb{P} \left( \max_{k \in [n]} \left\{ (-T_k) - \sum_{i \in [k]} (-\mu_v^{(i)} \cdot \mathbb{1}_{\mu_v^{(i)} < 0}) \right\} \leq \lambda' \right) \\
& \geq \mathbb{P} \left( \max_{k \in [n]} \left\{ (-T_k) - \sum_{i \in [k]} (-\mu_v^{(i)}) \right\} \leq \lambda' \right) \\
& \geq \mathbb{P} \left( \max_{k \in [n]} \left\{ T_k - \sum_{i \in [k]} \mu_v^{(i)} \right\} \leq \lambda' \right) \\
& \geq 1 - \frac{\sum_{k \in [n]} (\sigma^2_k + \sigma^2) v}{\lambda^2} \to 0,
\end{align*}
\]
where the last inequality follows from Kolmogorov’s inequality [9]. Hence \( \sum_{n=1}^{\infty} \) also converges to 0 almost surely. Combined with (12), we have
\[
\mathbb{P}(\lim_{n \to \infty} \sum_{k \in [n]} W_k / n^{3/2+\epsilon} = 0) = 1.
\]

The minimum total flow time is no more than that incurred by FCFS, i.e., \( F^{*} \leq F_{FCFS} = \sum_{k \in [n]} (W_k + p_k) = o(n^{3/2+\epsilon}) \) holds with whole probability. The proof is complete.

\[\square\]

**Proposition 17.** If the single server system is stable, i.e., \( \mathbb{P}(\lim_{n \to \infty} W_n < \infty) = 1, \) then
\[
\sum_{k \in [n]} \mu_v^{(k)} \cdot \mathbb{1}_{\mu_v^{(k)} > 0} = o(n^{-1/2}).
\]

**Proof.** See [21].

\[\square\]

**Proposition 18.** The stability of a multiple server system with job workload \( \{p_k\}_{k \in [n]} \) and interarrival time \( \{\Delta r_k\}_{k \in [n]} \) implies that
\[
\sum_{k \in [n]} (\mu_v^{(k)} - m\mu_v^{(k)} + m\mu_v^{(k)} + \sum_{i \in [k]} \mu_v^{(i)}) = o(n^{-1/2+\epsilon}).
\]

**Proof.** We consider a single server system \( \Sigma^* \) with the same input distributions of interarrival and service time, while the server is \( m \) times as fast as that in the multiple server system \( \Sigma^m. \) We remark that the remaining workload in this single server system is always no more than that in \( \Sigma^m, \) since the server in \( \Sigma^* \) always reduces the workload at the same rate as the case when all the \( m \) servers in \( \Sigma^m \) are busy, while the newly arriving jobs in these two systems are identical. Hence if the remaining workload in \( \Sigma^* \) goes to infinity, then the remaining workload in \( \Sigma^m \) is also unbounded, thus condition (16) follows from Proposition 17.

\[\square\]
- \(|\mu_r(i) - \mu_r(j)| = o(n^{1/2})\), since \(\sum_{k \in [n]} (m\mu_r - \sum_{k=i}^{k+m-1} \mu_i) + (m-1)n \cdot \sup_{i,j} |\mu_r(i) - \mu_r(j)| = o(n^{1/2})\).

- \(\{\mu_r(k)\}_{k \in [n]}\) is non-decreasing, which implies that \(\sum_{k=i}^{k+m-1} \mu_r(k) = 0\).

By (18) and Lemma 16, we have \(\mathbb{P}(F_{\pi^*} = o(n^{3/2})) = 1\) for all \(i \in [n]\), consequently we know that \(F_{\pi^*} = \int_{A_i} F_{\pi^*} = \sum_{i \in [m]} o(n_i^2) = o(n^2)\) holds almost surely. The proof is complete. \(\square\)

B. Putting things together

We first derive the asymptotic upper bound on the maximum job size \(B^{(n)}\). For any fixed \(\Delta\), without loss of generality we can assume that \(p_{min} \geq \Delta\). Otherwise, consider another benchmark system \(\Sigma'\), in which the sizes of jobs with workload below the threshold \(\Delta\) are increased to \(\Delta\). Then the total flow time under any algorithm \(\pi\) in system \(\Sigma\) is no more than that in \(\Sigma'\), i.e., \(F_{\pi} \leq F_{\pi'}(\forall r)\), and \(F_{\pi^*} \leq F_{\pi^*'} \leq F_{\pi^*}\).

Combined with Theorem 7, we have
\[
F_{\pi} \leq F'_{\pi} \leq B^{(n)} \cdot F'_{\pi^*},
\]
where \(B^{(n)} = p_{max}/\Delta\) satisfies
\[
\lim_{n \to \infty} B^{(n)} = \lim_{n \to \infty} \frac{\max_{k \in [n]} p_k}{\Delta - n^{1/2}} = 0, \text{ w.p.1},
\]
according to Lemma 14. In addition, the \(\alpha\)-th moment of the job size in system \(\Sigma'\) is also finite,
\[
\mathbb{E}[p'^\alpha] = \int_{[0,\Delta]} p'^\alpha f'(p')dp' + \int_{[\Delta,\infty]} p'^\alpha f'(p')dp' \leq \Delta^\alpha + \mathbb{E}[p'^\alpha] < \infty, \forall \alpha > 0,
\]
and the job workload distributions are independent, which implies that \(\mathbb{P}(F'^{\pi'}_{\pi^*} = o(n^{3/2+\epsilon})) = 1\). Combining with Observation 15, the proof for identical machine setting is complete.

Machines with different speeds. Now we prove the optimality condition for machines in parallel with different speeds. We would like to point out that our result still holds when there are \(m\) machines in parallel with different speeds and the mean values of job interarrival time are identical.

In the following we use \(s_i\) to denote the speed of machine \(i\). Firstly, to bound \(F_{\pi^*}\), we reduce the problem to a single server system by assigning each arriving job to machine \(i\) with probability \(p_i = \frac{p_i}{\sum_{j=1}^{m} s_j}\), which is similar to [6]. It can be seen that the job interarrival time at machine \(i\) can be expressed as the summation of \(n_i\) random variables with mean value of \(\mu_r\), where \(n_i\) follows geometric distribution with success probability \(p_i\). Then the mean value of interarrival time at machine \(i\) is equal to \(\Delta_r(i) = \mathbb{E}[\Delta_r(i)] = \mu_r/p_i = \mu_r \cdot \sum_{j \in [m]} s_j\). Similar to the proof of Lemma 18, the stability of the multi-server system implies that
\[
\sum_{i \in [n]} \left(\frac{\mu_r(i)}{\sum_{j \in [m]} s_j} - \mu_r\right) = o(n^{1/2}),
\]
from which we can obtain the \(o(n^{3/2})\) bound on the optimal flow time at each machine, based on Lemma 16 and the following fact,
\[
\sum_{j \in A_i} \left(\frac{\mu_r(j)}{s_j} - \mu_r\right) = \sum_{j \in A_i} \left(\frac{\mu_r(j)}{s_j} - \mu_r \cdot \sum_{j \in [m]} s_j\right) = \frac{m-1}{s_i} \cdot \sum_{j \in [m]} s_j \cdot \left(\frac{\mu_r(j)}{s_j} - \mu_r\right) = o(n^{1/2}).
\]

The remaining proof is similar as the identical machine setting. For example, similar as Theorem 7, we can show that
\[
F_{\pi} \leq B \cdot \left[\left(\int_{t \in [T_i]} n_r(t)dt + \int_{t \in [T_i]} (m-1)dt\right) + F_{\pi^*}\right] \leq \left(2 + \sum_{j \in [m]} s_j\right) B \cdot F_{\pi^*} = O(B) \cdot F_{\pi^*}.
\]
The proof is complete.

V. FURTHER GENERALIZATIONS

Minimizing the weighted completion time. Indeed for the more general problem of minimizing weighted total completion time, the asymptotic optimality of work-conserving algorithms still hold, which can be proved via the same arguments as the unit weight case. The proof is deferred to [21].

Proposition 20. Any work conserving algorithm \(\pi\) is almost surely asymptotically optimal for minimizing the weighted total completion time under Assumption 4 or 5, and the interarrival time, job workload and weight \(\{\omega_k\}_{k \in [n]}\) defined on \([1, +\infty)\) are independent with
- finite second, \(\alpha\)-th and \(\beta\)-th moments respectively, where \(1/\alpha + 1/\beta = 1 - \epsilon\).
- finite second, \(2 + \epsilon\)-th moments and finite generating function (i.e., \(\mathbb{E}[e^{\omega_k}] < \infty \forall k \in [n]\)) respectively.

Relaxing the independence assumption. It is clear that our analysis indeed carries over beyond the independence assumptions on job workload and arrival process. The asymptotic optimality condition requires nothing more than the convergence results in inequalities (12) and (14). We remark that Theorem 1 can be indeed generalized to the setting when Assumption 4 is replaced by the following condition. The proof is deferred to [21].

Assumption 21. There exists \(\{u_r(k)\}_{k \in [n]}\) and \(\{u_r(k)\}_{k \in [n]}\) such that for \(\forall 1 \leq i \leq j \leq n\),
\[
\mathbb{E}\left[\sum_{k=i}^{j} (p_k - \mu_r(k))\right] < \sum_{i \leq \ell \leq j} u_r(\ell),
\]
\[
\mathbb{E}\left[\sum_{k=i}^{j} (\Delta_r(k) - \mu_r(k))\right] < \sum_{i \leq \ell \leq j} u_r(\ell).
\]
VI. Numerical Results

In this section we conduct simulations to validate the convergence of competitive ratios of various work-conserving disciplines. We consider a computing system with $m = 20$ machines. Job processing times are i.i.d distributed and follow from exponential distribution with mean $\mu$. Jobs arrive according to a Poisson process with rate $\lambda = m \rho \mu$, where $\rho$ represents the traffic intensity. As the system will be less congested when the traffic intensity $\rho$ is small, intuitively the resulting total completion time should be close to the minimum total completion time. Therefore we focus on scenarios when $\rho$ is close to 1. More specifically, we let $\mu = 1/40$, $\lambda = 0.45$ and $0.49$, where $\rho = 0.9$ and $\rho = 0.98$ respectively.

![Estimated Cumulative Distribution Function](image1)

Fig. 1: Empirical Distribution Function under FCFS

As shown in Figure 1–3, for each discipline, we plot the empirical distribution function, i.e., the estimated cumulative distribution function (CDF), of the ratio between total completion time incurred and that incurred under the optimal discipline OPT. It is worth mentioning that finding the exact total completion under OPT is computationally expensive, due to the NP-hardness and exponential search space. We use the total arrival time, an explicit lower bound of total completion time, to calculate the ratios. Hence, the ratios in the figures are indeed pessimistic estimations of the real value and for every fixed sample path, the ratios under different disciplines are amplified by the same factor. The list of (work-conserving) scheduling disciplines considered and corresponding results are summarized as following.

- **First Come First Serve (FCFS)**. FCFS is the simplest form of scheduling algorithm, which always processes the jobs by the order of their arrival. FCFS is the default scheduler in Hadoop. Results are shown in Figure 1.
- **Shortest Remaining Processing Time (SRPT)**. As it is well-known, jobs with lower remaining processing time have a higher priority under SRPT. SRPT is efficient in optimizing the metric of mean response time and has been applied in several real life applications, including web servers [16]. Results are shown in Figure 2.

![Estimated Cumulative Distribution Function](image2)

Fig. 2: Empirical Distribution Function under SRPT

- **Longest Remaining Processing Time (LRPT)**. As opposed to SRPT, LRPT always processes the job with longest remaining processing time. Since large jobs are handled slowly, we can get a sense of how poor that the performances of work-conserving algorithms can be by considering LRPT. Results are shown in Figure 3.

**Discussion on the numerical results.** We can see that for all the disciplines and sample paths in the experiments, the total completion time has a small constant gap (no more than 7) to the optimum. This result coincides with the intuition that completion time is a relatively robust evaluation metric. On the other hand, almost for each fixed discipline, the ratio between the total completion time and the optimum total completion time incurred by a large number of jobs, is stochastically smaller than (first-order stochastically dominated) that incurred by a smaller number of jobs. In addition, the empirical CDF converges to the unit step function at ratio of 1, which verifies our asymptotic optimality conclusion. Note that in the

3Random variable $A$ has a first order dominance over random variable $B$ if $P(A \geq x) \geq P(B \geq x)$ for all $x$ and for some $x$, $P(A \geq x) > P(B \geq x)$. 

![Estimated Cumulative Distribution Function](image3)
results above, the gap between work-conserving disciplines and OPT is close to 1 when the number of jobs is in the order of $10^3$, which is common in large scale applications.

\[\text{Estimated Cumulative Distribution Function}\]

[Fig. 3: Empirical Distribution Function under LRPT]

VII. CONCLUSION

In this paper, we proved that in parallel machine environment, all work-conserving disciplines are asymptotic optimal in minimizing total completion time, as long as the interarrival time and job workload are independently distributed and have finite second and $(2 + \varepsilon)$-th moment respectively, while the mean interarrival time are almost identical or non-decreasing. We further discussed simple generalization to weighted completion time minimization and showed possible relaxations on the independence assumption. To establish the result, we also obtained a complete characterization in competitive ratio bounds for work-conserving disciplines and the objective of flow time.

ACKNOWLEDGMENT

This work has been supported in part by NSF grants CNS-1901057 and CNS-1717060, ONR grant N00014-17-1-2417, and by the IITP grant (MSIT), (2017-0-00692, Transport-aware Streaming Technique Enabling Ultra Low-Latency AR/VR Services).

REFERENCES

[1] Edward J. Anderson and Chris N. Potts. On-line scheduling of a single machine to minimize total weighted completion time. In SODA, pages 548–557, 2002.

[2] Søren Asmussen. Applied probability and queues. Springer-Verlag, 2008.

[3] Nikhil Bansal, Aravind Srinivasan, and Ola Svensson. Lift-and-round to improve weighted completion time on unrelated machines. SIAM Journal on Computing, (0):138–159, 2019.

[4] Chandra Chekuri and Sanjeev Khanna. Approximation algorithms for minimizing average weighted completion time. Handbook of Scheduling: Algorithms, Models, and Performance Analysis, 2004.

[5] Chandra Chekuri, Rajeev Motwani, Balas Natarajan, and Clifford Stein. Approximation techniques for average completion time scheduling. SIAM Journal on Computing, 31(1):146–166, 2001.

[6] Gang Chen and Zuo-Jun Max Shen. Probabilistic asymptotic analysis of stochastic online scheduling problems. IIE Transactions, 39(5):525–538, 2007.

[7] Mabel C. Chou, Hui Liu, Maurice Queyranne, and David Simchi-Levi. On the asymptotic optimality of a simple on-line algorithm for the stochastic single-machine weighted completion time problem and its extensions. Operations Research, 54(3):464–474, 2006.

[8] Richard Walter Conway, William L. Maxwell, and Louis W. Miller. Theory of scheduling. Courier Corporation, 2003.

[9] Rick Durrett. Probability: Theory and Examples. Cambridge University Press, New York, USA, 4th edition, 2010.

[10] Janos Galambos. The asymptotic theory of extreme order statistics. Robert E. Krieger Publishing Company, 1978.

[11] John Gittins, Kevin Glazebrook, and Richard Weber. Multi-armed bandit allocation indices. John Wiley & Sons, 2011.

[12] Ronald L. Graham, Bounds on multiprocessing timing anomalies. SIAM Journal on Applied Mathematics, 17(2):416–429, 1969.

[13] Ronald L. Graham, Eugene L. Lawler, Jan Karel Lenstra, and Ahg Rinnoy Kan. Optimization and approximation in deterministic sequencing and scheduling: a survey. In Annals of Discrete Mathematics, volume 5, pages 287–326, 1979.

[14] Leslie A. Hall, Andreas S. Schulz, David B. Shmoys, and Joel Wein. Scheduling to minimize average completion time via random offsets from non-uniform distributions. In FOCS, pages 138–147, 2016.

[15] Philip Kaminsky and David Simchi-Levi. The asymptotic optimality of the spt rule for the flow shop mean completion time problem. Operations Research, 49(2):293–304, 2001.

[16] Stefano Leonardi and Danny Raz. Approximating total flow time on parallel machines. In STOC, pages 110–119, 1997.

[17] Shi Li. Scheduling to minimize total weighted completion time via time-indexed linear programming relaxations. In FOCS, pages 283–294, 2017.

[18] Wenxin Li and Ness Shroff. On the asymptotic optimality of work-conserving disciplines in completion time minimization. arXiv:1912.12535, 2019.

[19] Rajeev Motwani, Steven J. Phillips, and Eric Torg. Non-clairvoyant scheduling. In SODA, pages 422–431, 1993.

[20] Cynthia Phillips, Clifford Stein, and Joel Wein. Minimizing average completion time in the presence of release dates. Mathematical Programming, 82(1-2):199–223, 1998.

[21] David B. Shmoys, Joel Wein, and David P. Williamson. Scheduling parallel machines on-line. SIAM Journal on Computing, 24(6):1331–1333, 1995.

[22] Martin Skutella. A 2.542-approximation for precedence constrained single machine scheduling with release dates and total weighted completion time objective. Operations Research Letters, 44(5):676–679, 2016.

[23] Wayne E. Smith. Various optimizers for single-stage production. Naval Research Logistics Quarterly, 3(1-2):59–66, 1956.

[24] Yousi Zheng, Ness B. Shroff, and Prasun Sinha. A new analytical technique for designing provably efficient mapreduce schedulers. In INFOCOM, pages 1600–1608, 2013.

[25] Yousi Zheng, Prasun Sinha, and Ness B. Shroff. Performance analysis of work-conserving schedulers for minimizing total flow-time with phase precedence. In Allerton, pages 1721–1728, 2012.