WORST-CASE PERFORMANCE ANALYSIS OF SOME APPROXIMATION ALGORITHMS FOR MINIMIZING MAKESPAN AND FLOWTIME

PERUVEMBA SUNDARAM RAVI, LEVENT TUNÇEL, MICHAEL HUANG

Abstract. In 1976, Coffman and Sethi conjectured that a natural extension of LPT list scheduling to the bicriteria scheduling problem of minimizing makespan over flowtime optimal schedules, called the LD algorithm, has a simple worst-case performance bound: $\frac{5m-2}{4m-1}$, where m is the number of machines. We study the structure of potential minimal counterexamples to this conjecture, provide some new tools and techniques for the analysis of such algorithms, and prove that to verify the conjecture, it suffices to analyze the following case: for every $m \geq 4$, $n \in \{4m, 5m\}$, where n is the number of jobs.

Date: December 2013: revised September 6, 2015.

Key words and phrases. parallel identical machines, makespan, total completion time, approximation algorithms for scheduling.

Peruvemba Sundaram Ravi (**Corresponding Author**): Operations and Decision Sciences, School of Business and Economics, Wilfrid Laurier University, Waterloo, Ontario N2L 3C5, Canada (e-mail: pravi@wlu.ca, phone: (519)884-0710 ext.2674, fax: (519)884-0201)
Levent Tunçel: Department of Combinatorics and Optimization, Faculty of Mathematics, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada (e-mail: ltuncel@uwaterloo.ca)
Michael Huang: (e-mail:Michael-Huang@u.northwestern.edu).

1. Introduction

Various performance criteria may be used to schedule n independent jobs on m parallel identical machines. In applications where work-in-process inventory is very highly valued (resulting in potentially huge work-in-process inventory costs), or in applications where the total time spent in the system must be minimized, a fundamental objective function of choice would be the minimization of total flow time (or equivalently mean flow time). A schedule that minimizes mean flow time is termed a *flowtime-optimal schedule*. The mean flow time for a set of jobs on a set of parallel identical machines can be readily minimized by using the SPT (Shortest Processing Time) rule. The SPT rule generates a very large number (at least $(m!)^{\lfloor n/m \rfloor}$) of flowtime-optimal schedules. Another fundamental objective function of choice, as a secondary objective, would be the minimization of makespan. Minimizing the makespan has been shown to be an $\mathcal{NP}$-hard problem. Various versions of this problem have been studied by several researchers.

Graham (1966) examines the problem of makespan minimization for a set of jobs with a partial order (precedence constraints) on a set of parallel identical machines. Graham (1969) as well as Coffman and Sethi (1976b) develop bounds for solutions obtained by the application of the LPT (Longest Processing Time) rule to the makespan minimization problem with no precedence constraints. The bin-packing problem may be regarded as the dual problem to the makespan minimization problem. In the bin-packing problem, a set of items of—in general—unequal sizes $\ell(T_i)$ must be packed into a set of m bins, each of a given capacity C. The objective of the bin-packing problem is to minimize the number of bins m used for the packing. A very well-known heuristic for bin-packing problem is the FFD (First Fit Decreasing) algorithm (Johnson (1973) and Johnson et al. (1974)). Garey and Johnson (1981) and Coffman, Garey and Johnson (1983) have proposed various alternatives to the FFD algorithm for the bin-packing problem. Further, Coffman, Garey, and Johnson (1978) propose an algorithm for makespan minimization, the MultiFit algorithm, that is based on the FFD algorithm for the bin-packing problem, and obtain a bound for the performance of this algorithm. Recently, the exact performance ratio for the MultiFit algorithm was established as 24/19 by Hwang and Lim (2014). Moreover, de la Vega and Lueker (1981) propose a linear-time approximation scheme for the bin packing problem. Friesen (1984), Yue, Kellerer and Yu(1988), Yue (1990), and Cao (1995) also propose bounds for the performance of the MultiFit algorithm. Dosa (2000 and 2001) proposes generalized versions of the LPT and MultiFit methods. Chang and Hwang (1999) extend the MultiFit algorithm to a situation in which different processors have different starting times. Hochbaum and Shmoys (1987) propose a PTAS (Polynomial Time Approximation Scheme) for the makespan minimization problem for parallel identical machines. Ho and Wong (1995) propose an $O(2^n)$ algorithm to find the optimal solution for a two-machine version of this problem.

The problem of selecting the schedule with the smallest makespan among the class of all flowtime-optimal schedules is known to be $\mathcal{NP}$-hard (Bruno, Coffman and Sethi, 1974). We term this problem the *FM (Flowtime-Makespan) problem.* Coffman and Yannakakis (1984) study a more general version of the FM problem. They study the problem of permuting the elements within the columns of an m-by-n matrix so as to minimize its maximum row sum. Eck and Pinedo (1993) propose a new algorithm for the FM problem. Their algorithm, the LPT* algorithm, is a modified version of Graham's LPT algorithm. LPT* requires the construction of a new problem instance by replacing every processing time in a rank by the difference between it and the smallest processing time in that rank. For the two-machine case, the authors obtain a worst-case bound of 28/27 on the makespan ratio for the LPT* algorithm. Gupta and Ho (2001) build on the procedure developed by Ho and Wong (1995) for makespan minimization to develop three algorithms — an algorithm to find the optimal solution and two heuristic procedures — for the two-machine FM problem. Lin and Liao (2004) extend the procedures developed by Ho and Wong (1995) and by Gupta and Ho (2001) to construct a procedure to obtain an optimal solution to the FM problem in $O\left((m!)^{n/m}\right)$ time.

Conway, Maxwell and Miller (1967), in their seminal book, develop the notion of ranks for the FM problem. A schedule is flowtime-optimal if jobs are assigned in decreasing order of *ranks*, with the jobs in rank 1 being assigned last. If n is the number of jobs and m is the number of machines, we may assume that m divides n. (If it does not, we add $(\lceil n/m \rceil * m - n)$ dummy jobs with zero processing times.) If we assume that the jobs are numbered in nonincreasing order of processing times, with job 1 having the largest processing time, the set of jobs belonging to rank r are the following: $(r-1)m+1, (r-1)m+2, \cdots, (r-1)m+m$. Coffman and Sethi (1976a) propose two approximation algorithms for the FM problem. In the LI algorithm, ranks are assigned in decreasing order, starting with the rank containing the m jobs with the smallest processing times. In the LD algorithm, ranks are assigned in increasing order, starting with the rank containing the m jobs with the largest processing times. Jobs with the same rank are assigned largest-first onto distinct machines as they become available after executing the previous ranks. In the LD algorithm, the sequence thus obtained must be reversed and all jobs in the last rank must be set to the same starting time of zero to ensure that the schedule is flowtime-optimal. Coffman and Sethi show that the LI algorithm has a makespan ratio (ratio of the makespan to the optimal makespan) with a worst-case bound that is equal to $(5m-4)/(4m-3)$.

(1) Coffman and Sethi (1976a) conjecture that the LD algorithm has a makespan ratio with a worst-case bound equal to

$$\frac{5m-2}{4m-1}.$$

The next family of instances for the FM problem shows that the above conjectured ratio cannot be improved for any $m \geq 2$. For $m \geq 2$, let $n := 3m$ and consider

$$p_j := \begin{cases} 0 & \text{for } j \in \{1, 2, \ldots, m-1\} \\ m & \text{for } j = m \\ (j-1) & \text{for } j \in \{m+1, m+2, \ldots, 2m\} \\ (j-2) & \text{for } j \in \{2m+1, 2m+2, \ldots, 3m\}. \end{cases}$$

It is easy to verify that the ratio of the objective value of an LD schedule to the optimal objective value is $\frac{5m-2}{4m-1}$.

Note that the LD algorithm for the FM problem seems to be inspired by the widely-known LPT algorithm for the problem $Pm\ /\ /\ C_{\max}$. Further note that even though the data set for both scheduling problems $Pm\ /\ /\ C_{\max}$ and FM are the same, there are many fundamental distinctions:

1. For the above family of instances, the LPT and the LD schedules are the same. Consider the cases $m = 2$ and $m = 3$. As we prove in this paper, while LD schedules are the worst-possible for the FM problem (attaining the approximation ratio $\frac{5m-2}{4m-1}$), these are not the worst schedules for LPT for $Pm\ /\ /\ C_{\max}$. (As is well-known, the worst-case approximation ratio for LPT for the problem $Pm\ /\ /\ C_{\max}$ is $\frac{4}{3} - \frac{1}{3m}$). Therefore, even when the LD and LPT schedules are the same for a given instance and the LD schedule exhibits the worst possible performance for the FM problem, the same instance may not be a worst-case instance for LPT for the problem $Pm\ /\ /\ C_{\max}$.

2. Consider now a well-known instance for $m = 2$. Let the processing times be $3, 3, 2, 2, 2, 0$. This is the worst-case instance for LPT for the problem $Pm\ /\ /\ C_{\max}$. For this instance, the LPT and LD schedules are the same. While the LPT schedule has the worst possible approximation ratio of $\frac{7}{6}$ for the problem $Pm\ /\ /\ C_{\max}$, the LD schedule, and hence the LPT schedule, is optimal for the FM problem for this instance. Indeed, the worst-case instance for LD for the FM problem, in the two machine case, is given by a different instance, with the processing times $4, 3, 3, 2, 2, 2, 0$.

3. Finally, consider the instance with processing times $29, 5, 5, 4, 4, 3, 3, 2, 2, 1$ with $m = 2$. The LPT schedule leads to a makespan of 29 and is optimal for the problem $Pm\ /\ /\ C_{\max}$. However, this LPT schedule is not flowtime-optimal and hence is not feasible for the FM problem. The LD schedule (putting jobs $J_{10}, J_8, J_6, J_4, J_1$ on one machine and the jobs J_9, J_7, J_5, J_3, J_2 on the other machine) has a makespan of 39 and is optimal for the FM problem.

Our approach in obtaining a proof of Coffman and Sethi's conjectured bound is to identify properties of a hypothesized minimal counterexample to the conjecture. We utilize three techniques:

1. We show that there always exists a minimal counterexample with integer processing times (even if we allow irrational data).

2. Using integrality of the data and the other properties of the minimal counterexamples, we construct "smaller" problem instances by subtracting nonnegative integers from the number of machines, the number of jobs, or the integer-valued processing times. For each such problem instance, we can write a constraint that expresses the fact that the makespan ratio cannot exceed $\frac{5m-2}{4m-1}$. We then prove that minimal counterexamples must have a very small number of ranks.

3.The final stage of the proof is to verify that none of these small instances (with a small number of ranks) can be counterexamples. To establish this last bit, we introduce another approach which is to treat processing times of jobs as unknown variables and show how to set up the case analysis with a set of finitely many LP problems. By solving these finitely many LP problems, whose optimal objective function values can be certified (and easily verified) using primal and dual optimal solutions, we verify the conjecture for these small instances. Our approach is based on general techniques and may be applicable to other combinatorial optimization problems.

The rest of this paper is organized as follows. Section 2 contains a description of the LD algorithm. In Section 3, the notion of minimality that is used in this paper is defined. In Section 4, we discuss an LP-based approach that treats problem parameters as variables. We use this approach to prove that the Coffman-Sethi conjecture holds for problem instances with two machines and for problem instances with three machines and three ranks (this latter fact also becomes a corollary of a theorem in Section 5 and hence has two independent proofs). In Section 5, we derive several properties of a hypothesized minimal counterexample to the Coffman-Sethi conjecture. These include an upper bound on the number of ranks. Section 6 contains concluding remarks and outlines possible extensions and directions for future research.

2. LD Algorithm for Problem FM

Let p_j denote the processing time of job j, where the jobs are numbered in nonincreasing order of processing times. Thus, $p_j \geq p_{j+1}$. The set of jobs belonging to rank r are the following:

$$(r-1)m+1, (r-1)m+2, \cdots, (r-1)m+m.$$

A schedule in which all rank $(r+1)$ jobs are started before all rank r jobs (where $r \in \{1, 2, \cdots, (n/m)-1\}$) is said to satisfy the *rank restriction* or *rank constraint.* A schedule which satisfies the rank constraint and in which no idle time exists between successive jobs assigned to the same machine and all rank n/m jobs start at time 0, is termed a *flowtime-optimal schedule.* Let λ_r and μ_r denote the largest and smallest processing times respectively, in rank r. Note that

$$\lambda_1 \geq \mu_1 \geq \lambda_2 \geq \mu_2 \geq \ldots \geq \lambda_{k-1} \geq \mu_{k-1} \geq \lambda_k \geq \mu_k \geq 0, \tag{2}$$

where, $k := \lceil \frac{n}{m} \rceil$. The *profile* of a schedule after rank r is defined as the sorted set of completion times on m machines after rank r. If the jobs in r ranks (out of a total of k) have been assigned to machines, and if the jobs in the remaining ranks have not yet been assigned to machines, the profile after rank r is termed the *current profile.* We let $a(r) \in \mathbb{R}^m : a_1(r) \geq a_2(r) \geq \cdots \geq a_m(r)$ denote the current profile. Note that $a_i(\ell)$ is the ith largest completion time after rank ℓ.

The LD algorithm (Coffman and Sethi, 1976a) starts with rank 1, the rank containing the m jobs with the largest processing times, and works its way through ranks $1, 2, \ldots, k-1, k$. The sequence of jobs on each machine is then reversed to make it a flowtime-optimal schedule. The algorithm works as follows: Schedule the ranks in the following order: $1, 2, \ldots, k-1, k$. Let $a \in \mathbb{R}^m$: $a_1 \geq a_2 \geq \cdots \geq a_m$ denote the current profile. Schedule the jobs in the next rank so that the job with the largest processing time is matched with a_m, second largest with a_{m-1}, etc., and the smallest processing time is matched with a_1. After all the jobs are scheduled, reverse the schedule and left-justify it (i.e., start the first job on each machine at time zero).

3. A definition of minimality, and preliminaries

3.1. **A definition of minimality.** If an instance of the FM problem with m machines and k ranks has fewer than mk jobs, increasing the number of jobs to mk by including up to $m-1$ jobs with zero processing time does not result in any change in the total flowtime or the optimal makespan or the makespan of an LD schedule, though it may change the set of jobs allocated to each rank. From this observation, it follows that, for the purpose of proving/disproving conjectures regarding worst-case makespan ratios for algorithms for the problem FM, only problem instances with mk jobs need to be considered. Thus, we assume the property

$$n = mk. \tag{Property.1}$$

We define the ordered set of processing times P for a scheduling problem instance with m machines and k ranks to consist of elements equal to the processing times of these mk jobs arranged in nonincreasing order. We let $P(j)$ refer to the j_{th} element of P. Thus, $P(j) \geq P(j+1)$ for $j \in \{1, \ldots, mk-1\}$.

For the problem of characterizing minimal counterexamples to the Coffman-Sethi conjecture, minimality will be defined based on the following criteria: the number of machines, the number of ranks, and the set of processing times. A minimal counterexample is defined to be a counterexample with k ranks, m machines, and a set of processing times P_1 for which there does not exist another counterexample with one of the following (in a hierarchical order):
(i) number of ranks fewer than k
(ii) number of ranks equal to k and number of machines fewer than m
(iii) k ranks, m machines, and fewer jobs with nonzero processing times
(iv) k ranks, m machines, the same number of jobs with nonzero processsing times, and a worse approximation ratio
(v) [only for integer data] k ranks, m machines, the same number of jobs with nonzero processing times, the same approximation ratio, and a set of processing times P_2 satisfying
$\sum_{j=1}^{n} P_2(j) < \sum_{j=1}^{n} P_1(j)$.

In the latter parts of the paper, we will restrict our attention purely to integer data. There, we will use the minimality criterion (v). Note that due to integrality of the processing times, we will have a normalization of the data (the smallest nonzero processing time is at least 1).

We may assume that in a minimal counterexample,

$$\text{(Property.2)} \qquad \mu_r = \lambda_{r+1}, \forall r \in \{1, 2, \ldots, k-1\} \text{ and } \mu_k = 0.$$

If the above property fails, then either $\mu_k > 0$ (in this case, we subtract μ_k from the processing time of every job in rank k, the new instance is also a counterexample to the conjecture with the same number of machines, ranks and fewer jobs with nonzero processing times, a contradiction to the minimality of the original instance) or there exists $r \in \{1, 2, \ldots, k-1\}$ such that $\mu_r > \lambda_{r+1}$ (in this case, we subtract $(\mu_r - \lambda_{r+1})$ from the processing time of every job in rank r, the new instance is also a counterexample to the conjecture with the same number of machines, ranks, the number of jobs with nonzero processing times, and with a worse approximation ratio, a contradiction to the minimality of the original instance).

3.2. **Some preliminary results.** Next two results follow from classical work in the area.

Proposition 1. *The Coffman-Sethi conjecture holds for every m and n such that $n \leq 2m$.*

Proposition 1 can be proven using Graham's (1969) original analysis of LPT list scheduling. A *rectangular optimal schedule* is a feasible schedule in which the last job on every machine has the same completion time $\frac{\sum_{j=1}^{n} p_j}{m}$.

Lemma 1. *(Coffman and Sethi, 1976a) There always exists a problem instance with a worst-case t_{LD}/t^* ratio and with a rectangular optimal schedule.*

4. A Linear Programming based approach

We present an approach which may be utilized to prove or disprove any conjecture about a bound on the ratio of the solution generated by an approximation algorithm to the optimal solution for a maximization problem. In this approach, the parameters of the problem are treated as variables. The problem of maximizing the ratio is formulated as an LP problem. If bounds can be obtained on the size of the problem, only a finite number of LPs need to be examined. Indeed, the optimal objective value of each LP problem can be verified by checking the optimality conditions for the given pair of optimal solutions to the primal and the dual problem at hand (independent of the computations that led to these optimal solutions). In this section, this approach is used to show that the Coffman-Sethi conjecture is valid for the $m = 2$ and the $m = 3, k = 3$ cases, where m denotes the number of machines and k denotes the number of ranks. In subsequent sections, bounds are obtained on the number of ranks. In principle, this LP-based approach can be used to obtain a computer-based proof of the conjecture or a minimal counterexample for any finite number of machines.

Lemma 2. *If the Coffman-Sethi conjecture is false for $m = 2$ then there exists a minimal counterexample with two machines and three ranks.*

Proof. Suppose that the Coffman-Sethi conjecture is false. Take a minimal counterexample to the conjecture. Then, by (Property.2), in rank k, one of the two machines has a processing time of λ_k and the other machine has a processing time of 0. Clearly, the makespan is equal to the completion time after rank k on the machine with a processing time of λ_k. (If this is not the case, the last rank could be deleted to obtain a problem instance with the same or larger makespan ratio, contradicting the minimality of the original instance.) It follows that both completion times on the two machines after rank $(k-1)$ in the LD schedule are at least $(t_{LD} - \lambda_k)$. This implies $t^* \geq t_{LD} - \lambda_k + \lambda_k/2$. In a counterexample, $t_{LD}/t^* > 8/7$. Hence,

$$t^* \quad < \quad 7\lambda_k/2. \tag{3}$$

Clearly, $\mu_\ell \geq \lambda_k$ for $\ell \in \{1, 2, \ldots, k-1\}$. Thus, $t^* \geq k\lambda_k$ which together with (3) yields $k \leq 3$. Note that for $k = 1$ and $k = 2$, the LD schedule is optimal. Therefore, the claim follows. □

The following approach treats the data of problem FM (job processing times) as variables. For a given value of the optimal makespan, the problem of determining the values of the processing times that result in the LD makespan being maximized is set up as a set of LP problems. Each possible relationship between the processing times (subject to the rank restriction) results in a different LP. The solution to each LP is checked to determine if it violates the Coffman-Sethi conjecture. Note that a major advantage of LP formulations is that the optimal objective values can be verified independent of the original computations, by a much simpler computational step (to check feasibility of primal and dual solutions and then a comparison of their objective values).

Proposition 2. *The Coffman-Sethi conjecture holds for $m = 2$.*

Proof. By Lemma 2, we only need to consider the $k = 3$ case. For a contradiction, suppose the conjecture is false. Then, there exists a minimal counterexample to the conjecture. Moreover, using (Property.2), we may assume that a minimal counterexample has the processing times: $\lambda_1, \lambda_2, \lambda_2, \lambda_3, \lambda_3, 0$. Then, it suffices to consider only two LD schedules:

- LD schedule 1: Jobs with processing times $\lambda_1, \lambda_3, 0$ on machine 1, jobs with processing times $\lambda_2, \lambda_2, \lambda_3$ on machine 2.
- LD schedule 2: Jobs with processing times $\lambda_1, \lambda_3, \lambda_3$ on machine 1, jobs with processing times $\lambda_2, \lambda_2, 0$ on machine 2.

For a makespan ratio > 1, the second and third ranks must not be the same in the LD schedule and the optimal schedule. There is only one possible optimal schedule: Jobs with processing times $\lambda_1, \lambda_2, 0$ on machine 1, jobs with processing times $\lambda_2, \lambda_3, \lambda_3$ on machine 2. In each of the following cases, we set the optimal makespan equal to 1. The makespan ratio is then equal to the LD makespan. We seek to maximize the LD makespan in each case. There are four possible values for the LD makespan, resulting in the following four cases.

Case 1: $t_{LD} = \lambda_1 + \lambda_3$. This will be true only if $\lambda_1 \geq 2\lambda_2$. So, $\lambda_1 \geq 2\lambda_3$, and we deduce $t^* = \lambda_1 + \lambda_2$. This is clearly not possible.

Case 2: $t_{LD} = 2\lambda_2 + \lambda_3$. This will be true only if $\lambda_1 \leq 2\lambda_2$ and $\lambda_1 + \lambda_3 \geq 2\lambda_2$.

Case 2A: $\lambda_1 \leq 2\lambda_3$. So, $t^* = \lambda_2 + 2\lambda_3$. Consider the LP problem

$$\max\left\{2\lambda_2 + \lambda_3 : \; \lambda_1 \leq 2\lambda_3, \; \lambda_2 + 2\lambda_3 = 1, \; 2\lambda_2 \leq \lambda_1 + \lambda_3\right\}.$$

This LP can be simplified as follows:

$$\max\left\{2\lambda_2 + \lambda_3 : \; 2\lambda_2 \leq 3\lambda_3, \; \lambda_2 + 2\lambda_3 = 1\right\}.$$

An optimal solution is $\lambda_2 = 3/7, \lambda_3 = 2/7$, with objective function value $8/7$ and a dual optimal solution $[3/7, 8/7]$.

Case 2B: $\lambda_1 \geq 2\lambda_3$. Thus, $t^* = \lambda_1 + \lambda_2$. Consider the LP problem

$$\max\left\{2\lambda_2 + \lambda_3 : \; \lambda_1 + \lambda_2 = 1, \; \lambda_1 \geq 2\lambda_3, \; \lambda_1 \leq 2\lambda_2, \; 2\lambda_2 \leq \lambda_1 + \lambda_3\right\}.$$

This LP can be simplified as follows:

$$\max\left\{2\lambda_2 + \lambda_3 : \; \lambda_2 \geq 1/3, \; \lambda_2 + 2\lambda_3 \leq 1, \; 3\lambda_2 \leq 1 + \lambda_3\right\}.$$

An optimal solution is $\lambda_2 = 3/7, \lambda_3 = 2/7$, with the objective function value $8/7$ and a dual optimal solution $[5/7, 3/7]$.

Case 3: $t_{LD} = \lambda_1 + 2\lambda_3$. This will be true only if $\lambda_1 + \lambda_3 \leq 2\lambda_2$ and $\lambda_1 + 2\lambda_3 \geq 2\lambda_2$.

Case 3A: $\lambda_1 \leq 2\lambda_3$. In this case, we have $t^* = \lambda_2 + 2\lambda_3$. Consider the LP problem:

$$\max\left\{\lambda_1 + 2\lambda_3 : \; \lambda_1 + \lambda_3 \leq 2\lambda_2, \; \lambda_1 + 2\lambda_3 \geq 2\lambda_2, \; \lambda_2 + 2\lambda_3 = 1, \; \lambda_1 \leq 2\lambda_3\right\}.$$

The constraints of the above LP problem can be replaced by the following equivalent set of constraints:

$$\lambda_1 + 5\lambda_3 \leq 2, \; \lambda_1 + 6\lambda_3 \geq 2, \; \lambda_1 \leq 2\lambda_3.$$

An optimal solution is $\lambda_1 = 4/7, \lambda_2 = 3/7, \lambda_3 = 2/7$, with objective function value $8/7$ and a dual optimal solution $[4/7, 3/7]$.

Case 3B: $\lambda_1 \geq 2\lambda_3$. Thus, $t^* = \lambda_1 + \lambda_2$. Consider the LP problem

$$\max\left\{\lambda_1 + 2\lambda_3 : \; \lambda_1 + \lambda_2 = 1, \; \lambda_1 \geq 2\lambda_3, \; \lambda_1 + \lambda_3 \leq 2\lambda_2, \; \lambda_1 + 2\lambda_3 \geq 2\lambda_2\right\}.$$

This LP problem can be simplified as follows:

$$\max\left\{\lambda_1 + 2\lambda_3 :\ \lambda_1 \geq 2\lambda_3,\ 3\lambda_1 + \lambda_3 \leq 2,\ 3\lambda_1 + 2\lambda_3 \geq 2\right\}.$$

An optimal solution is $\lambda_1 = 4/7, \lambda_2 = 3/7, \lambda_3 = 2/7$, with the objective function value $8/7$ and a dual optimal solution $[5/7, 4/7]$.

Case 4: $t_{LD} = 2\lambda_2$. This will be true only if $2\lambda_2 \geq \lambda_1 + 2\lambda_3$.

Case 4A: $\lambda_1 \leq 2\lambda_3$. Hence, $t^* = \lambda_2 + 2\lambda_3$. Consider the LP problem

$$\max\left\{2\lambda_2 :\ \lambda_1 \leq 2\lambda_3,\ \lambda_2 + 2\lambda_3 = 1,\ 2\lambda_2 \geq \lambda_1 + 2\lambda_3\right\}.$$

The constraints of the above LP problem can be replaced by the following equivalent set of constraints:

$$\lambda_1 + \lambda_2 \leq 1,\ -\lambda_1 + 3\lambda_2 \geq 1.$$

An optimal solution is $\lambda_1 = 1/2, \lambda_2 = 1/2, \lambda_3 = 1/4$, with the objective function value 1 and a dual optimal solution $[1/2, -1/2]$.

Case 4B: $\lambda_1 \geq 2\lambda_3$. Then, $t^* = \lambda_1 + \lambda_2$. Consider the LP problem

$$\max\left\{2\lambda_2 :\ \lambda_1 + \lambda_2 = 1,\ \lambda_1 \geq 2\lambda_3,\ \lambda_1 + 2\lambda_3 \leq 2\lambda_2\right\}.$$

This LP problem can be simplified as follows:

$$\max\left\{2\lambda_2 :\ \lambda_2 \geq \lambda_3,\ 2\lambda_3 + \lambda_2 \leq 1,\ -2\lambda_3 + 3\lambda_2 \geq 1\right\}.$$

An optimal solution is $\lambda_1 = 1/2, \lambda_2 = 1/2, \lambda_3 = 1/4$, with the objective function value 1 and a dual optimal solution $[1/2, -1/2]$.

We conclude that, in all cases, the ratio of the makespan of every LD schedule is at most $8/7$ times the optimum makespan. This is a contradiction to the existence of a counterexample. Thus, the Coffman-Sethi conjecture holds for $m = 2$, and $k = 3$. Therefore, by Lemma 2, the Coffman-Sethi conjecture holds for $m = 2$. □

Using the same technique as in the proof of Proposition 2 we can prove the conjecture for $m = 3$ and $k = 3$. A proof of the next proposition is provided in the appendix.

Proposition 3. *The Coffman-Sethi conjecture holds for the $m = 3$, $k = 3$ case.*

5. Properties of minimal counterexamples and their analysis

5.1. **Properties of minimal counterexamples.** Most scheduling problems are considered in the context of the Turing machine model of computation and, as a result, the data are assumed to be drawn from the rationals. In our problem FM, this would mean that $p_j \in \mathbb{Q}, \forall j \in \{1, 2, \ldots, n\}$. Even if irrational data is permitted, a counterexample to a conjectured makespan ratio with irrational processing times could exist only if there existed a counterexample with rational processing times. For the details and a proof of the following result, see Ravi (2010).

Proposition 4. *For the FM problem and the LD algorithm, the following must hold: If there exists a counterexample E to a conjectured t_{LD}/t^* ratio, then there exists a counterexample EI with integer processing times.*

Moreover, if the conjecture is false, there must exist a counterexample with integer processing times and a *rectangular optimal schedule.* Therefore, we define two types of problem instances, counterexamples and minimal counterexamples to the Coffman-Sethi conjecture:

- A problem instance of *Type I* is one with integer processing times.
- A problem instance of *Type IR* is one with integer processing times and a rectangular optimal schedule.

A counterexample of a particular type (I or IR) is a problem instance of that type that violates the Coffman-Sethi conjecture. A *minimal counterexample* of a particular type is a counterexample of that type for which there does not exist a smaller counterexample (based on the notion of minimality defined in Section 3.1) of the same type. In the remaining part of this paper, we will assume that all problem instances and hypothesized counterexamples are of Type I.

The following lemma examines the effect of an increase in processing times in a given rank subject to the rank constraint. An increase in processing times that is subject to the rank constraint is an increase that is not large enough to result in a job being reassigned to another rank. Note that, for the first rank, no increase in processing times can result in a job being reassigned to another rank.

Lemma 3. *An increase in one or more processing times of jobs in rank r for $r \in \{1, 2, \dots, k-1\}$ with no change in the remaining processing times, and subject to the rank constraint, does not result in a reduction in any element of the profile $b(\ell)$ of an LD schedule after rank $\ell \in \{r, r+1, \dots, k\}$.*

Proof. We proceed by induction on ℓ. Let us assume that the lemma holds for ℓ' ranks, where $\ell' \in \{r, r+1, \dots, s\}$. Let $\tau_{i,h}$ refer to the i^{th} largest processing time in rank h. The induction hypothesis states that an increase in processing times in rank r does not cause a reduction in $b_{m-i+1}(\ell')$ for $i \in \{1, 2, \dots, m\}$. Note that the increase in processing times in rank r leaves $\tau_{i,\ell'+1}$ unchanged for $i \in \{1, 2, \dots, m\}$. The profile $b(\ell'+1)$ after rank $\ell'+1$ consists of the following m elements:

$$\left[b_{m-i+1}(\ell') + \tau_{i,\ell'+1}\right] \text{ for } i \in \{1, 2, \dots, m\}.$$

It follows that none of the elements of the profile $b(\ell'+1)$ gets reduced as a result of the increase in processing times in rank r. Thus, the theorem holds for rank $\ell'+1$, for $\ell' \in \{r, r+1, \dots, s\}$. The base case follows from the fact that the result holds trivially for $\ell' = r$. □

For problem instances of Type I, an optimal schedule that is not rectangular can be made rectangular by increasing the lengths of all jobs in the first rank that are performed on machines that have a completion time after the last rank that is strictly less than the makespan. For example, consider a set of 9 jobs assigned to 3 machines. The processing times are as follows: $1, 4, 5, 5, 6, 7, 7, 8, 9$. An optimal schedule for this instance of the FM problem is given in Figure 1. The completion times on the three machines are 17, 17, and 18 respectively. Adding 1 unit to the processing times of the jobs in the first rank of machines 1 and 2 would convert this schedule to a rectangular schedule with a completion time of 18 on every machine (see Figure 2).

Lemma 4. *There always exists a problem instance of Type I with a worst-case t_{LD}/t^* ratio and with a rectangular optimal schedule.*

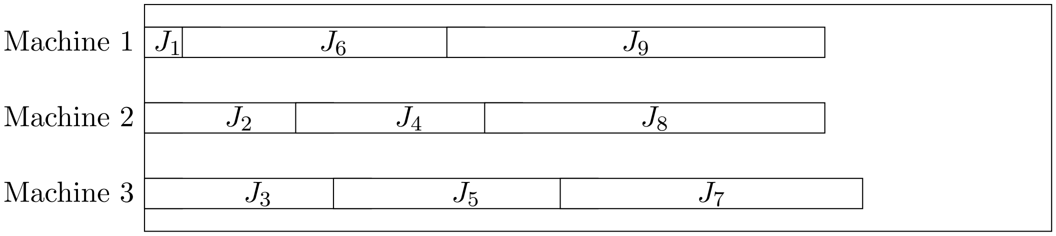

FIGURE 1. An optimal schedule

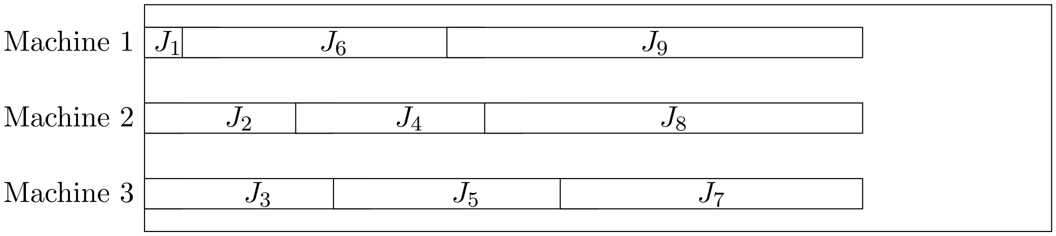

FIGURE 2. Optimal schedule made rectangular, with the same makespan

Note that Coffman and Sethi (1976a) state this result for problem instances with processing times that are not required to be integers. Also note that, if the Coffman-Sethi conjecture is false, a problem instance with a worst-case t_{LD}/t^* ratio would be a counterexample to the conjecture. This leads to the following corollary.

Corollary 1. *If the Coffman-Sethi conjecture is false, then there exists a minimal counterexample to the conjecture of Type IR.*

Proof. Follows from Proposition 4 and the proof of Lemma 4. □

5.2. **Results based on Type I and Type IR.**

Theorem 1. *If the Coffman-Sethi conjecture is false, then every minimal counterexample to the conjecture of Type IR or I satisfies*

$$\frac{t_{LD}}{t^*} < \frac{k}{k-1}.$$

Proof. Proof for minimal counterexamples of Type IR:
Suppose that the Coffman-Sethi conjecture is false. Among all counterexamples of Type IR, consider a minimal counterexample P1. Now, we apply the following two-step process:
Step 1: Reduce each nonzero processing time by 1 to construct a new problem instance P2. Clearly, the assignment of jobs in each rank to machines can be kept unchanged for the new LD schedule. The schedule that was originally an optimal rectangular schedule for P1 will not be rectangular, but will be optimal for P2 after the reduction in processing times. In the modified schedule, there are only $(k-1)$ jobs assigned to the one or more machines with a zero processing time job in the last rank. This implies that the new optimal makespan is equal to $[t^* - (k-1)]$.
Step 2: For every job in rank 1 of the optimal schedule for P2 that is processed on a machine with a completion time after rank k that is less than the makespan, increase the processing time

so that the completion time after rank k becomes equal to the makespan. This produces a problem instance P2R of Type IR. P2R will have an optimal makespan that is equal to the optimal makespan of P2. Utilizing Lemma 3, it follows that the LD makespan of P2R cannot be less than the LD makespan of P2.
The optimal makespan of P2R is equal to $[t^* - (k-1)]$. For the LD schedule for P2R, there are two possibilities: (i) t_{LD} gets reduced to a value that is greater than or equal to $[t_{LD} - (k-1)]$. This results in a new problem instance of Type IR, with a makespan ratio that is at least $[t_{LD} - (k-1)] \,/\, [t^* - (k-1)]$. This new ratio is larger than t_{LD}/t^*, thus contradicting the assumption that the original problem instance was a minimal counterexample of Type IR. (ii) t_{LD} gets reduced to $(t_{LD} - k)$. This results in a new problem instance with a makespan ratio at least $[t_{LD} - k] \,/\, [t^* - (k-1)]$. The original problem instance P1 was a minimal counterexample of Type IR. This implies that $(t_{LD} - k)/(t^* - (k-1))$ is less than t_{LD}/t^*. Therefore, $t_{LD}/t^* < k/(k-1)$.
Proof for minimal counterexamples of Type I:
Consider a minimal counterexample P1′ of Type I. Now reduce each nonzero processing time by 1 to construct a new problem instance P2′. The assignment of jobs in each rank to machines can be kept unchanged for the new LD schedule and the new optimal schedule. The new optimal makespan is less than or equal to $t^* - (k-1)$. For the new LD schedule, there are two possibilities:
(i) t_{LD} gets reduced to $[t_{LD} - (k-1)]$. This results in a new problem instance with a makespan ratio that is at least $[t_{LD} - (k-1)] \,/\, [t^* - (k-1)]$. This new ratio is larger than t_{LD}/t^*, thus contradicting the assumption that the original problem instance was a minimal counterexample of Type I.
(ii) t_{LD} gets reduced to $t_{LD} - k$. This results in a new problem instance with a makespan ratio that is at least $[t_{LD} - k] \,/\, [t^* - (k-1)]$. The original problem instance P1′ was a minimal counterexample of Type I. This implies that $(t_{LD} - k)/\,[t^* - (k-1)]$ is less than t_{LD}/t^*. Therefore, $t_{LD}/t^* < (k/(k-1))$. This completes the proof for minimal counterexamples of Type I. □

The next result follows directly from the above theorem and from Propositions 1, 2 and 3.

Corollary 2. *The Coffman-Sethi conjecture is true, iff it is true for $k \in \{3, 4, 5, 6\}$. In particular,*

- *for $m \geq 4$, settling the cases $k \in \{3, 4, 5\}$ suffices;*
- *for $m = 3$, settling the cases $k \in \{4, 5, 6\}$ suffices.*

We define a minimal counterexample of Type IR1 as follows. If the Coffman-Sethi conjecture is false, a minimal counterexample of Type IR1 is a minimal counterexample of Type IR that has an LD schedule with the following property: Every machine with a completion time after rank k equal to the makespan has a job with processing time equal to λ_k in rank k, where k denotes the number of ranks.

Lemma 5. *If the Coffman-Sethi conjecture is false, then there exists a minimal counterexample to the conjecture of Type IR1, and every minimal counterexample of Type IR is a minimal counterexample of Type IR1.*

Proof. Suppose the Coffman-Sethi conjecture is false. Then, by Corollary 1, a counterexample of Type IR exists. Suppose, for a contradiction, that there exists a minimal counterexample P1 of Type IR that is not Type IR1. Now, apply the following two-step process.

Step 1: Construct a new problem instance P2 as follows. Subtract 1 time unit from the processing time of every job in rank $(k-1)$. Also subtract 1 time unit from the processing time of every job in rank k that has a processing time of λ_k. Note that, for a minimal counterexample, all processing times in rank k are not equal, therefore there exist processing times in rank k that are less than λ_k. Leave these processing times unchanged. Note that the assignment of jobs in each rank to machines in an LD schedule remains unchanged. Thus, P2 has an LD schedule with objective value $(t_{LD}-1)$, where t_{LD} is the objective value of the LD schedule for P1. Let t^* denote the optimal makespan of problem instance P1. Problem instance P1 had an optimal rectangular schedule. Reducing the processing time of every job in rank $(k-1)$ by 1 leaves the rectangular property unchanged and results in a reduction of 1 in the optimal makespan. A further reduction of 1 in one or more, but not all, jobs in rank k results in no further reduction in the optimal makespan. Thus, problem P2 has an optimal makespan that is equal to (t^*-1).
Step 2: For every job in rank 1 of the optimal schedule for P2 that is processed on a machine with a completion time after rank k that is less than the makespan, increase the processing time so that the completion time after rank k becomes equal to the makespan.

This produces a problem instance P2R of Type IR. By Lemma 3, the LD makespan of P2R cannot be less than the LD makespan of P2. So, it is at least $(t_{LD}-1)$. By construction, the optimal makespan for P2R is equal to the optimal makespan of P2, (t^*-1). Thus, P2R gives a strictly worse approximation ratio than P1, a contradiction to the minimality of P1. Therefore, P1 (as well as any other minimal counterexample of Type IR) is a minimal counterexample of Type IR1. □

5.3. **Properties of Type I1 and Type I2 counterexamples.**

We define a problem instance, a counterexample and a minimal counterexample of *Type I1* as follows. A problem instance of *Type I1* is a problem instance of Type I that has an LD schedule with the following properties:

(i) It has only one machine i' with a completion time after rank k equal to the makespan.
(ii) Machine i' has a processing time equal to λ_{k-1} in rank $(k-1)$ and λ_k in rank k.

We summarize the relationships among various types of counterexamples in Figure 3 (the arrows in the figure indicate for instance that Type I1 is a further refinement of Type I) and Table 1.

Lemma 6. *If the Coffman-Sethi conjecture is false, then there exists a minimal counterexample to the conjecture of Type I1.*

Proof. Suppose the Coffman-Sethi conjecture is false. Then, by Lemma 5, there exists a minimal counterexample of Type IR1. Call this instance P1. Now, construct a new problem instance P2 as follows. Subtract 1 time unit from the processing time of every job in rank $(k-2)$. Also subtract 1 time unit from the processing time of every job in rank $(k-1)$ that has a processing time of λ_{k-1}. Note that, for a minimal counterexample, all processing times in rank $(k-1)$ are not equal, therefore there exist processing times in rank $(k-1)$ that are less than λ_{k-1}. Leave these processing times unchanged. Note that the assignment of jobs in each rank to machines in the LD schedule remains unchanged, except possibly in rank k. Problem instance P1 had an optimal rectangular schedule. Reducing the processing time of every job in rank $(k-2)$ by 1 leaves the rectangular property unchanged and results in a reduction of 1 in the optimal makespan. A further reduction of 1 in one or more, but not all, jobs in rank $(k-1)$ results in no further reduction

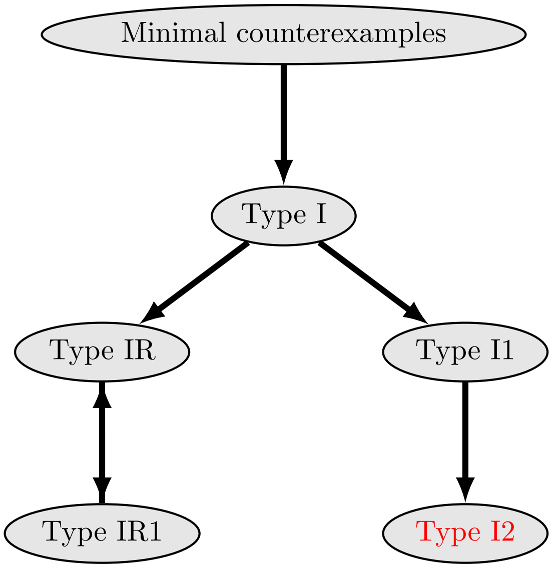

FIGURE 3. Refinement relationships among the minimal counterexamples in terms of required properties (note that by Lemma 5, the refinement relation between IR and IR1 is symmetric)

in the optimal makespan. Thus, problem P2 has an optimal makespan equal to $(t^* - 1)$. The new LD objective value is either $(t_{LD} - 1)$ (if there is at least one machine whose completion time originally equaled t_{LD} and after the modification of the processing times, now equals $(t_{LD} - 1)$ since in rank $(k - 1)$ it had a job with processing time less than λ_{k-1}) or $(t_{LD} - 2)$ (if every machine, with completion time equal to t_{LD} for P1, had a job in rank $(k - 1)$ with processing time λ_{k-1}). We will show below that the former case cannot happen.

Now, construct a problem instance P2R of Type IR by adding 1 unit to the processing time of every job in rank 1 that is performed on a machine with completion time after rank k in the optimal schedule that is less than the makespan. The optimal makespan of P2R is equal to the optimal makespan of P2, $(t^* - 1)$. Since P1 is a minimal counterexample of Type IR, using Lemma 3 and the above argument, we deduce that the makespan of the LD schedule for problem instance P2R is $(t_{LD} - 2)$. Thus, in the LD schedule for P1, every machine with completion time equal to t_{LD} had a job in rank $(k - 1)$ with processing time λ_{k-1}. By Lemma 5, every machine with completion time equal to t_{LD} has jobs with processing times λ_{k-1} and λ_k in ranks $(k - 1)$ and k respectively. Pick one of these machines, call it i'.

Construct a new counterexample P3 as follows. In the LD schedule for P1, for every machine $i \neq i'$ with a completion time after rank k equal to the makespan, delete the job in rank k. The makespan and the set of jobs assigned to i' in the LD schedule for P3 are the same as those in the LD schedule for P1. The makespan of the optimal schedule for P3 is less than or equal to the makespan of the optimal schedule for P1. However, the optimal schedule for P3 may not be rectangular. P3 is clearly a counterexample to the conjecture of Type I1. It follows that there exists a minimal counterexample to the conjecture of Type I1. □

Type I: Integer data	
IR: I + rectangular optimal schedule	I1: I + LD schedule with unique critical m/c i' having λ_{k-1}, λ_k in the last two ranks
IR1: IR + LD schedule with all critical m/c's having λ_k in the last rank	I2: I1 + m/c i' having $\lambda_2, \lambda_3, \ldots, \lambda_{k-1}, \lambda_k$ in the last $(k-1)$ ranks

Table 1. Brief descriptions of various types of counterexamples

We define a problem instance, a counterexample and a minimal counterexample of *Type I2* as follows. A problem instance of *Type I2* is a problem instance of Type I1 with the following property in an LD schedule: The machine i' with a completion time equal to the makespan has a processing time equal to λ_r in rank r for every $r \in \{2, 3, \ldots, k\}$.

Lemma 7. *If the Coffman-Sethi conjecture is false, then there exists a minimal counterexample to the conjecture of Type I2.*

Proof. Suppose that the Coffman-Sethi conjecture is false. We proceed by induction on the number of ranks for which the claim holds. The base case is given by Lemma 6. The induction hypothesis is that there exists a minimal counterexample P1 of Type I1 that has an LD schedule with a machine i' which has a processing time equal to λ_r in rank r for every $r \in \{h, h+1, \ldots, k\}$, where $h \geq 3$. If machine i' has a job in rank $(h-1)$ with processing time λ_{h-1} then we are done; otherwise, construct P2 from P1 by subtracting 1 time unit from the processing time of every job in rank $(h-2)$, and subtracting 1 time unit from the processing time of every job in rank $(h-1)$ that has a processing time of λ_{h-1}. Leave the remaining processing times unchanged. Note that either a new LD schedule assigns the same jobs to machine i', or another machine i'' which had the same completion time as machine i' after rank $(h-1)$ and a job with processing time λ_{h-1} now has a completion time one less and has the jobs originally scheduled on machine i' for ranks $r \in \{h, h+1, \ldots, k\}$. In the former case, P2 has a strictly worse approximation ratio than P1, a contradiction. Therefore, we must be in the latter case. In the latter case, go back to instance P1. After rank $(h-1)$, both machines i' and i'' have the same completion time. For the ranks $h, h+1, \ldots, k$, swap all jobs on the machines i' and i''. We still have an LD schedule for P1 with makespan t_{LD}; moreover, machine i'' is the unique machine with completion time equal to the makespan. Finally, the processing times on machine i'' are $\lambda_{h-1}, \lambda_h, \lambda_{h+1}, \ldots, \lambda_k$ as desired. It follows that there exists a minimal counterexample to the conjecture of Type I2. □

Lemma 8. *If the Coffman-Sethi conjecture is false, in a minimal counterexample of Type I2, the sole machine i' with a completion time after rank k equal to the makespan in the LD schedule has a processing time equal to μ_1 in rank* 1*.*

Proof. Suppose the Coffman-Sethi conjecture is false. Then, by the previous lemma, there exists a minimal counterexample to the conjecture of Type I2, call it P1. If, in the LD schedule for P1, there exists only one machine with a processing time equal to λ_r in rank r for $r \in \{2, 3, \ldots, k\}$, the lemma clearly holds. Assume that there exists a set of two or more machines with a processing time equal to λ_r in rank $r \in \{2, 3, \ldots, k\}$. We may assume that machine i' has a job with processing time strictly greater than μ_1 in rank 1. Then, there exists machine i'' which has processing time μ_1 in rank one, then by the definition of the LD algorithm, machine i'' must have the processing times: $\mu_1, \lambda_2, \lambda_3, \ldots, \lambda_k$ respectively. We delete all the jobs on machine i'' and delete the machine i'' to generate a new instance P2.

Note that the LD schedule for P2 is unchanged on machine i' and all the other machines except i''. Therefore, t_{LD} is unchanged. Next, we prove that the optimal makespan for P2 is no larger than that for P1. Consider an optimal schedule for P1. A machine i_1 has the jobs with processing times $\mu_1, q_2, q_3, \ldots, q_k$ respectively. Clearly, $q_r \leq \lambda_r$, for every $r \in \{2, 3, \ldots, k\}$. If the equality holds throughout, then we are done. Otherwise, for each r that the inequality is strict, we find the job with processing time λ_r in rank r and swap it with q_r. These operations may increase the completion time on machine i_1; however, they will not increase it on any other machine. At the end, we delete machine i_1 (now, with the processing times $\mu_1, \lambda_2, \lambda_3, \ldots, \lambda_k$). What remains is a feasible schedule for instance P2 whose makespan is at most the optimal makespan for P1.

We repeat the above procedure, until there exists only one machine in the LD schedule with a processing time equal to λ_r in rank $r \in \{2, 3, \ldots, k\}$. From the mechanics of the LD algorithm, it follows that no other machine has a smaller processing time in rank 1. This completes the proof. □

Lemma 9. *If the Coffman-Sethi conjecture is false, then in the LD schedule for a minimal counterexample of Type I2, the smallest completion time after rank k on any machine is at least*

$$t_{LD} - \max_{r \in \{2,3,\ldots,k\}} \{\lambda_r - \mu_r\}.$$

Proof. Suppose the Coffman-Sethi conjecture is false. Then, by Lemma 7, there exists a minimal counterexample of Type I2. By Lemma 8, in a minimal counterexample of Type I2, the sole machine i' with a completion time after rank k equal to the makespan has a processing time equal to μ_1 in rank 1. For $i \in \{1, 2, \ldots, m\}$, $i \neq i'$, let r_i denote the smallest value of $r \in \{1, 2, \ldots, k\}$, for which the completion time after rank r_i on machine i is less than the completion time after rank r_i on machine i'. By Lemma 7, it follows that there exists a value $r_i \leq k$ for all $i \neq i'$. Lemma 8 implies that $r_i \geq 2$. There are two possible cases:

(a) $r_i = k$. In this case, the completion time after rank k on machine i is greater than or equal to $t_{LD} - \lambda_k$.

(b) $r_i < k$. In this case, from the mechanics of the LD algorithm, it is evident that the processing time of the job on machine i in rank $(r_i + 1)$ and all of the following ranks must be equal to λ_r. Therefore, the completion time after rank k on machine i is greater than or equal to $t_{LD} - (\lambda_{r_i} - \mu_{r_i})$.

This completes the proof of the lemma. □

Lemma 10. *If the Coffman-Sethi conjecture is false, then in the LD schedule for any minimal counterexample of Type I2, there exists at least one value of i'' that satisfies $i'' \neq i'$, for which the completion time after rank $(k-1)$ on machine i'' is greater than or equal to the completion time after rank $(k-1)$ on machine i', where i' denotes the sole machine with a completion time after rank k equal to the makespan.*

Proof. Assume that the claim of the lemma does not hold (we are seeking a contradiction). Therefore, there exists a minimal counterexample of Type I2 for which every machine $i \neq i'$ has a completion time after rank $(k-1)$ that is less than the completion time after rank $(k-1)$ on machine i'. From the mechanics of the LD algorithm, it is evident that every machine $i \neq i'$ has a processing time in rank k that is greater than or equal to the processing time of the job assigned to machine i' in rank k. Hence every machine has a job with processing time λ_k assigned to it in rank k. Therefore, all jobs in rank k can be removed from the counterexample to obtain a smaller counterexample of Type I2 with a larger t_{LD}/t^* ratio. This contradicts the assumption that the original counterexample was a minimal counterexample of Type I2. □

Lemma 11. *If the Coffman-Sethi conjecture is false, then in the LD schedule for any minimal counterexample of Type I2, there exists at least one value of i'' that satisfies $i'' \neq i'$, for which the completion time after rank k on machine i'' is less than the completion time after rank $(k-1)$ on machine i', where i' denotes the sole machine with a completion time after rank k equal to the makespan.*

Proof. Assume that the claim of the lemma does not hold (we are seeking a contradiction) and there exists a minimal counterexample of Type I2 for which every machine $i \neq i'$ has a completion time after rank k that is greater than or equal to the completion time after rank $(k-1)$ on machine i'. A minimal counterexample must have at least 3 ranks. We first prove the lemma for the $k \geq 4$ case and then prove the lemma for the $k = 3$ case. We have $t^* \geq t_{LD} - \lambda_k + \frac{\lambda_k}{m}$. Supposing that we have a counterexample, we deduce

$$\begin{aligned} t^* &> \left(\frac{5m-2}{4m-1}\right) t^* - \left(\frac{m-1}{m}\right) \lambda_k \\ t^* &< \left(4 - \frac{1}{m}\right) \lambda_k. \qquad (4) \end{aligned}$$

For $k \geq 4$, $t^* > 4\lambda_k$, we reached a contradiction. This completes the proof of the lemma for $k \geq 4$.

For $k = 3$, there exists at least one machine in the optimal schedule with a job with processing time of λ_3 in the third rank. Therefore,

$$t^* \geq \mu_1 + \mu_2 + \lambda_3.$$

From the preceding lemmas, it follows that $t_{LD} = 2\lambda_2 + \lambda_3$. For a counterexample, we must have

$$\frac{t_{LD}}{t^*} > \frac{5m-2}{4m-1}.$$

Therefore, using (Property.2), we have

$$\frac{2\lambda_2 + \lambda_3}{\lambda_2 + 2\lambda_3} > \frac{5m-2}{4m-1}.$$

It follows that

$$\lambda_2 > \left(2 - \frac{1}{m}\right) \lambda_3. \qquad (5)$$

Utilizing the inequality $t^* \geq t_{LD} - \lambda_k + \frac{\lambda_k}{m}$, and the fact $t_{LD} = 2\lambda_2 + \lambda_3$, we deduce

$$t^* \geq (2\lambda_2 + \lambda_3) - \lambda_3 + \frac{\lambda_3}{m}. \qquad (6)$$

From inequalities (5) and (6), it follows that

$$t^* \quad > \quad \left(4 - \frac{1}{m}\right)\lambda_3. \tag{7}$$

From inequalities (4) and (7), we have a contradiction for $k = 3$. This completes the proof of the lemma. □

5.4. **Proofs for three ranks and three machines.**

Theorem 2. *The Coffman-Sethi conjecture holds for k equal to* 3.

Proof. Suppose the statement of the theorem is false. Consider a minimal counterexample of Type I2 with $k = 3$. Consider the LD schedule for this minimal counterexample. Let i' denote the sole machine with a completion time after rank k equal to the makespan. In the LD schedule, let M_1 denote the set of m_1 machines with a completion time after rank $(k-1)$ that is greater than or equal to the completion time after rank $(k-1)$ on machine i'. Note that set M_1 includes machine i' and that machine i' is the only machine in M_1 with a processing time of λ_3 in the third rank. The remaining $(m_1 - 1)$ machines have no job assigned to them in the third rank.

From Lemma 10, it is evident that $m_1 \geq 2$. Let M_2 denote the set of $m_2 := m - m_1$ machines with a completion time after rank $(k-1)$ that is less than the completion time after rank $(k-1)$ on machine i'. Every machine in M_2 has a processing time of λ_3 in the third rank. The total number of machines with λ_3 in the third rank is $(m - m_1 + 1)$.

From Lemma 11, it is evident that $m_2 \geq 1$. Select a machine i_2 in the set M_2 with a completion time after rank k that is less than the completion time after rank $(k-1)$ on machine i'. Let α_1 denote the processing time in rank 1 of machine i_2 and let α_2 denote the processing time in rank 2 of machine i_2.

For any machine i_1 in the set M_1: Let β_1 denote the processing time in rank 1 of machine i_1 and let β_2 denote the processing time in rank 2 of machine i_1. Note that $\beta_1 + \beta_2 \geq 2\lambda_2 > \alpha_1 + \alpha_2 + \lambda_3$.

Two possible cases need to be considered.

Case 1: $\beta_1 \leq \alpha_1$. In this case, $\beta_2 \geq 2\lambda_2 - \beta_1 > \alpha_1 + \alpha_2 + \lambda_3 - \beta_1$. Hence,
$\beta_2 > \alpha_2 + \lambda_3 \geq \mu_2 + \lambda_3$. Therefore, $\beta_2 > 2\lambda_3$.

Case 2: $\beta_1 > \alpha_1$. From the dynamics of the LD algorithm, $\beta_2 \leq \alpha_2$. In this case,
$\beta_1 \geq 2\lambda_2 - \beta_2 > \alpha_1 + \alpha_2 + \lambda_3 - \beta_2$, whence $\beta_1 > \alpha_1 + \lambda_3 \geq \mu_1 + \lambda_3$. Therefore, $\beta_1 > \lambda_2 + \lambda_3$.

Now, it is evident that, in any schedule (including the optimal schedule), one of the following must be true (recall that the total number of jobs with processing time λ_3 in rank 3 is $(m-m_1+1)$):
Case A: There exists a machine with a processing time of λ_3 in rank 3 and a processing time greater than $2\lambda_3$ in rank 2.
Case B: There exists a machine with a processing time of λ_3 in rank 3 and a processing time greater than $(\lambda_2 + \lambda_3)$ in rank 1.
Case C: There exists a machine with a processing time greater than $2\lambda_3$ in rank 2 and a processing time greater than $(\lambda_2 + \lambda_3)$ in rank 1.
In Case A, the makespan must be greater than $\lambda_3 + 2\lambda_3 + \mu_1$.
In Case B, the makespan must be greater than $[\lambda_3 + \mu_2 + (\lambda_2 + \lambda_3)]$.
In Case C, the makespan must be greater than $[2\lambda_3 + (\lambda_2 + \lambda_3)]$.

Thus, in all three cases, the makespan must be greater than $(\lambda_2+3\lambda_3)$. Therefore, $t^* > \lambda_2+3\lambda_3$. By Lemma 8, $t_{LD} = 2\lambda_2 + \lambda_3$. Thus, we have

$$\frac{5m-2}{4m-1} < \frac{t_{LD}}{t^*} < \frac{2\lambda_2+\lambda_3}{\lambda_2+3\lambda_3}.$$

In a minimal counterexample of Type I2, $m \geq 3$. We have

$$\frac{13}{11} < \frac{2\lambda_2+\lambda_3}{\lambda_2+3\lambda_3}.$$

Therefore,

$$\frac{\lambda_2}{\lambda_3} \quad > \quad \frac{28}{9}. \tag{8}$$

Also, note that, since $t^* > \mu_1 + \lambda_2 + \mu_3$, we have

$$\frac{13}{11} < \frac{t_{LD}}{t^*} < \frac{2\lambda_2+\lambda_3}{\mu_1+\lambda_2+\mu_3}.$$

Therefore,

$$\frac{\lambda_2}{\lambda_3} \quad < \quad \frac{11}{4}. \tag{9}$$

From inequalities (8) and (9), we obtain a contradiction. This completes the proof of the theorem. □

Theorem 3. *The Coffman-Sethi conjecture holds for $m = 3$.*

Proof. Consider a minimal counterexample of Type I2 with m equal to 3. From Lemmas 10 and 11, it follows that, for this counterexample, the following statements hold for an LD schedule. (i) There is one machine i' with a completion time after rank k that is equal the makespan. (ii)There is one machine i'' with a completion time after rank k that is less than the makespan and a completion time after rank $(k-1)$ that is greater than or equal to the completion time after rank $(k-1)$ on machine i'. (iii) There is one machine i''' with a completion time after rank k that is less than the completion time after rank $(k-1)$ on machine i'.

From the above given statements and from Lemma 9, it follows that

$$3t^* \geq t_{LD} + (t_{LD} - \lambda_k) + \left(t_{LD} - \max_{r\in\{2,3,\ldots,k\}} \{\lambda_r - \mu_r\}\right). \tag{10}$$

Considering machine i''' , and Lemma 9, it is clear that

$$\max_{r\in\{2,3,\ldots,k\}} \{\lambda_r - \mu_r\} > \lambda_k.$$

Therefore,

$$\max_{r\in\{2,3,\ldots,k\}} \{\lambda_r - \mu_r\} = \max_{r\in\{2,3,\ldots,k-1\}} \{\lambda_r - \mu_r\}.$$

It follows that

$$\max_{r\in\{2,3,\ldots,k-1\}} \{\lambda_r - \mu_r\} \leq \lambda_2 - \mu_{k-1} = \lambda_2 - \lambda_k. \tag{11}$$

From (10) and (11), it follows that

$$3t^* \geq t_{LD} + (t_{LD} - \lambda_k) + [t_{LD} - (\lambda_2 - \lambda_k)]. \tag{12}$$

Therefore,

$$t^* \geq t_{LD} - \frac{\lambda_2}{3}. \tag{13}$$

For a counterexample to the Coffman-Sethi conjecture with m equal to 3, we must have

$$\frac{t_{LD}}{t^*} > \frac{13}{11}. \tag{14}$$

From inequalities (13) and (14), it follows that

$$t^* < \frac{11}{6}\lambda_2. \tag{15}$$

For two or more ranks, t^* must be at least equal to $\mu_1 + \lambda_2$ or $2\lambda_2$. Therefore, inequality (15) cannot hold and we have a contradiction. This completes the proof. □

Theorems 2 and 3 allow us to strengthen the previous corollary.

Corollary 3. *The Coffman-Sethi conjecture is true, iff it is true for $m \geq 4$ and for $k \in \{4, 5\}$.*

6. Conclusion and Future Research

We studied the structure of potential minimal counterexamples to the Coffman-Sethi (1976) conjecture and proved that to establish the correctness of the conjecture, it suffices to prove the conjecture for $m \geq 4$ and $k \in \{4, 5\}$. Moreover, for each $m \geq 4$, these remaining cases $k \in \{4, 5\}$ can be verified by employing our approach from Section 4 and the Appendix, via solving a finite number of LP problems.

As a by-product of our approach, we introduced various techniques to analyze worst-case performance ratios. These techniques may be useful in analyzing the performance of approximation algorithms for other scheduling problems, or in general, other combinatorial optimization problems of a similar nature.

Acknowledgment: The work was supported in part by Discovery Grants from NSERC (Natural Sciences and Engineering Research Council of Canada) as well as an Undergraduate Research Assistantship from NSERC.

References

[1] Bruno, J., E. G. Coffman and R. Sethi, Algorithms for minimizing mean flow time, *INFORMATION PROCESSING '74: Processings of the IFIPS Conference*, 504–510, North-Holland Publishing Company, 1974.

[2] Cao, F., Determining the performance ratio of algorithm MULTIFIT for scheduling, in *Nonconvex Optimization Applications, Volume 4: Minimax and applications*, Kluwer Academic Publishers, Dordrecht, 79–96, 1995.

[3] Chang, S. Y. and H. C. Hwang, The worst-case analysis of the MULTIFIT algorithm for scheduling nonsimultaneous parallel machines, *Discrete Applied Mathematics*, 92(2–3), 135–147, 1999.

[4] Coffman, E. G., M. R. Garey and D. S. Johnson, An application of bin-packing to multiprocessor scheduling. *SIAM Journal of Computing* 1, 1–17, 1978.

[5] Coffman, E. G., M. R. Garey and D. S. Johnson, Dynamic bin packing, *SIAM Journal of Computing*, 12, 227–258, 1983.

[6] Coffman, E. G. and R. Sethi, Algorithms minimizing mean flowtime: schedule-length properties, *Acta Informatica* 6, 1–14, 1976a.

[7] Coffman, E. G. and R. Sethi, A generalized bound on LPT sequencing, *Proceedings of the 1976 ACM SIGMETRICS Conference on Computer Performance Modeling Measurement and Evaluation*, 306–310, 1976b.

[8] Coffman, E. G. and M. Yannakakis, Permuting elements within columns of a matrix in order to minimize maximum row sum, *Mathematics of Operations Research*, 9(3), 384–390, 1984.

[9] Conway, R. W., W. L. Maxwell and L. W. Miller, *Theory of Scheduling*, Addison Wesley, MA, USA, 1967.

[10] Dosa, G. Generalized multifit-type methods II, *Alkalmaz. Mat. Lapok*, 20(1) (2000), 91–111, 2000.

[11] Dosa, G. Generalized multifit-type methods for scheduling parallel identical machines, *Pure Mathematics and Applications*, 12(3), 287–296, 2001.

[12] Eck, B. T. and M. Pinedo, On the minimization of the makespan subject to flowtime optimality, *Operations Research*, 41(4), 1993.

[13] Friesen, D. K., Tighter bounds for the MULTIFIT processor scheduling algorithm, *SIAM Journal of Computing* 13, 179–181, 1984.

[14] Garey, M. R. and D. S. Johnson, Approximation algorithms for bin packing problems-a survey, in *Analysis and Design of Algorithms in Combinatorial Optimization*, G. Ausiello and M. Lucertini, (eds.), Springer-Verlag, New York, 147–172, 1981.

[15] Graham, R. L., Bounds for certain multiprocessing anomalies, *Bell Systems Technical Journal*, 45(9), 1563–1581, 1966.

[16] Graham, R. L., Bounds on multiprocessing timing anomalies, *SIAM Journal of Applied Mathematics*, 17(2), 416–429, 1969.

[17] Gupta, J. N. D. and J. C. Ho, Minimizing makespan subject to minimum flowtime on two identical parallel machines, *Computers and Operations Research*, 28, 705–717, 2001.

[18] Ho, J. C. and J. S. Wong, Makespan minimization for m parallel identical processors, *Naval Research Logistics*, 42, 935–948, 1995.

[19] Hochbaum, D. S. and D. B. Shmoys, Using Dual Approximation Algorithms for Scheduling Problems: Theoretical and Practical Results, *Journal of the ACM*, 34(1), 144–162, 1987.

[20] Hwang, H.-C. and K. Lim, Exact performance of MULTIFIT for nonsimultaneous machines, *Discrete Appl. Math.*, 167, 172–187, 2014.

[21] Johnson, D. S., Near optimal bin packing algorithms, *Ph.D. Thesis*, Mathematics Dept., MIT, Cambridge, MA, 1973.

[22] Johnson, D. S., A. Demers, J. D. Ullman, M. R. Garey and R. L. Graham, Worst-case performance bounds for simple one-dimensional packing algorithms, *SIAM Journal of Computing*, 3, 299–325, 1974.

[23] Lin, C. H. and C. J. Liao, Makespan minimization subject to flowtime optimality on identical parallel machines, *Computers and Operations Research*, 31, 1655–1666, 2004.

[24] Ravi, P. S., *Techniques for Proving Approximation Ratios in Scheduling*, M.Math. Thesis, Dept. of Combinatorics and Optimization, Faculty of Mathematics, University of Waterloo, Canada, 2010.

[25] de la Vega, W. F. and G. S. Lueker, Bin packing can be solved within $1 + \epsilon$ in linear time, *Combinatorica*, 1(4),349–355, 1981.

[26] Yue, M. Y., On the exact upper bound for the multifit processor scheduling algorithm. *Annals of Operations Research*, 24, 1–4, 233–259, 1990.

[27] Yue, M., H. Kellerer and Z. Yu, A simple proof of the inequality $R_M(MF(k)) \leq 1.2 + 1/2^k$ in multiprocessor scheduling, *Report No. 124, Institut fur Mathematik, Technische Universitat Graz*, 1–10, 1988.

7. Appendix

7.1. **Three-Machine Case.** For $r \in \{1, 2, 3\}$, let λ_r, α_r and μ_r denote the processing times in rank r, where $\lambda_r \geq \alpha_r \geq \mu_r$. Clearly, $\mu_1 = \lambda_2$ and $\mu_2 = \lambda_3$ and $\mu_3 = 0$ and either $\alpha_3 = \lambda_3$ or $\alpha_3 = \mu_3 = 0$.

The first two ranks of the LD schedule will look like

$$\begin{pmatrix} \lambda_1 & \lambda_3 \\ \alpha_1 & \alpha_2 \\ \lambda_2 & \lambda_2 \end{pmatrix}.$$

The third rank will fit depending on the length of processing times on the first two ranks. We will use case analysis to cover all possible LD schedules.

For all cases we will have $\lambda_1 \geq \alpha_1 \geq \lambda_2 \geq \alpha_2 \geq \lambda_3 > 0$. This results in the following 5 constraints.

$$\begin{array}{cccccccl}
-\lambda_1 & & & +\alpha_1 & & \leq & 0 & (1)\\
& \lambda_2 & & -\alpha_1 & & \leq & 0 & (2)\\
& -\lambda_2 & & & +\alpha_2 & \leq & 0 & (3)\\
& & \lambda_3 & & -\alpha_2 & \leq & 0 & (4)\\
& & -\lambda_3 & & & \leq & 0 & (5)
\end{array}$$

We begin by looking at the cases when $\alpha_3 = \lambda_3$.

Consider the case $\lambda_1 + \lambda_3 \geq \alpha_1 + \alpha_2 \geq 2\lambda_2$, then we have the constraints

$$\begin{array}{cccccccl}
-\lambda_1 & & -\lambda_3 & +\alpha_1 & +\alpha_2 & \leq & 0 & (6)\\
& 2\lambda_2 & & -\alpha_1 & -\alpha_2 & \leq & 0 & (7)
\end{array}$$

Further, the LD schedule will look like

$$\begin{pmatrix} \lambda_1 & \lambda_3 & 0 \\ \alpha_1 & \alpha_2 & \lambda_3 \\ \lambda_2 & \lambda_2 & \lambda_3 \end{pmatrix}$$

$t_{LD} = \max(\lambda_1 + \lambda_3, \alpha_1 + \alpha_2 + \lambda_3)$. If $t^* = \lambda_1 + \lambda_3$, then t^* is equal to a lower bound and $\frac{t_{LD}}{t^*} = 1$. So, we may assume $t^* = \alpha_1 + \alpha_2 + \lambda_3$. Consider an optimal configuration for this problem. For the third job, the machine with λ_1 in the first rank must have a job in the third rank with processing time 0, otherwise that machine will have processing time at least $\lambda_1 + 2\lambda_3 \geq \alpha_1 + \alpha_2 + \lambda_3 = t_{LD}$. Since the machine with λ_1 in the frst rank has a job with processing time 0 in the last rank, the machine with α_1 in the first rank has a job with processing time λ_3 in the third rank. In order for t^* to be less than t_{LD}, the machine with α_1 in the first rank must have a job with processing time λ_3 in the second rank as well. Thus, the optimal configuration must be either

$$\begin{pmatrix} \lambda_1 & \alpha_2 & 0 \\ \alpha_1 & \lambda_3 & \lambda_3 \\ \lambda_2 & \lambda_2 & \lambda_3 \end{pmatrix} \text{ or } \begin{pmatrix} \lambda_1 & \lambda_2 & 0 \\ \alpha_1 & \lambda_3 & \lambda_3 \\ \lambda_2 & \alpha_2 & \lambda_3 \end{pmatrix}.$$

We consider the case where the former is the optimal configuration. Suppose $t^* = \lambda_1 + \alpha_2$. Then, we have $\lambda_1 + \alpha_2 \geq \alpha_1 + 2\lambda_3$ and $\lambda_1 + \alpha_2 \geq 2\lambda_2 + \lambda_3$. This leads to two more constraints

$$\begin{array}{cccccccl}
-\lambda_1 & & +2\lambda_3 & +\alpha_1 & -\alpha_2 & \leq & 0, & (8)\\
-\lambda_1 & +2\lambda_2 & +\lambda_3 & & -\alpha_2 & \leq & 0. & (9)
\end{array}$$

Since the processing times of the jobs can be scaled by any positive number (here, we are no longer requiring integer or even rational processing times in the input data), we let $t^* = 1$. We add the constraint

$$\lambda_1 \quad + \quad \alpha_2 \quad \leq \quad 1. \quad (10)$$

We now consider the linear program

$$\begin{array}{ll}
\max & t_{LD} = \alpha_1 + \alpha_2 + \lambda_3 \\
\text{subject to:} & (1),\ (2),\ \cdots,(10).
\end{array}$$

The optimal objective value will give the worst-case ratio of $\frac{t_{LD}}{t^*}$ for this particular case. Solving the LP yields the optimal objective value $\frac{9}{8}$ when $[\lambda_1, \lambda_2, \lambda_3, \alpha_1, \alpha_2] = \frac{1}{8}$ [5, 3, 2, 4, 3], with dual optimal solution $\frac{1}{8}$[0, 0, 2, 0, 0, 3, 0, 5, 1, 9].

We now suppose $t^* = \alpha_1 + 2\lambda_3$. Constraints (8)–(10) are now replaced with

$$\begin{array}{cccccccc}
\lambda_1 & & -2\lambda_3 & -\alpha_1 & +\alpha_2 & \leq & 0 & (8') \\
& 2\lambda_2 & -\lambda_3 & -\alpha_1 & & \leq & 0 & (9') \\
& & 2\lambda_3 & +\alpha_1 & & \leq & 1. & (10')
\end{array}$$

This new LP also yields the optimal objective value $\frac{9}{8}$ when $[\lambda_1, \lambda_2, \lambda_3, \alpha_1, \alpha_2] = \frac{1}{8}$ [5, 3, 2, 4, 3], with dual optimal solution $\frac{1}{8}$[0, 0, 2, 0, 0, 3, 0, 3, 1, 9].

We now suppose $t^* = 2\lambda_2 + \lambda_3$. Constraints (8)–(10) are replaced with

$$\begin{array}{cccccccc}
\lambda_1 & -2\lambda_2 & -\lambda_3 & & +\alpha_2 & \leq & 0 & (8'') \\
& -2\lambda_2 & +\lambda_3 & +\alpha_1 & & \leq & 0 & (9'') \\
& 2\lambda_2 & +\lambda_3 & & & \leq & 1. & (10'')
\end{array}$$

Again the new LP yields the optimal objective value $\frac{9}{8}$ when $[\lambda_1, \lambda_2, \lambda_3, \alpha_1, \alpha_2] = \frac{1}{8}$ [5, 3, 2, 4, 3], with dual optimal solution $\frac{1}{8}$[0, 0, 2, 0, 0, 3, 0, 3, 5, 9].

We now consider the other possible configuration as the optimal configuration. We consider each possibility for t^* by modifying the constraints (8)–(10) to accommodate each t^*. For example, when $t^* = \lambda_1 + \lambda_2$, constraints (8) - (10) are replaced with

$$\begin{array}{cccccccc}
-\lambda_1 & -\lambda_2 & +2\lambda_3 & +\alpha_1 & & \leq & 0 & (8''') \\
-\lambda_1 & & +\lambda_3 & & +\alpha_2 & \leq & 0 & (9''') \\
\lambda_1 & +\lambda_2 & & & & \leq & 1. & (10''')
\end{array}$$

For all three LPs for this configuration, the optimal objective value is $\frac{9}{8}$ which is attained when $[\lambda_1, \lambda_2, \lambda_3, \alpha_1, \alpha_2] = \frac{1}{8}$ [5, 3, 2, 4, 3], with dual optimal solution $\frac{1}{8}$[0, 0, 4, 0, 0, 3, 0, 5, 1, 9].

We have exhausted this case and now move to the case where $\lambda_1 + \lambda_3 \geq 2\lambda_2 \geq \alpha_1 + \alpha_2$. We modify the constraints (6) and (7) to reflect the inequalities related to this case. We continue in this manner and set up and solve LP problems for every possible case for the 3-machine, 3-rank problem. In every case, the makespan ratio is less than or equal to 13/11. This shows that the Coffman-Sethi conjecture holds for the 3-machine, 3-rank problem. Further, an LD makespan of 13/11 is actually attained in some cases, thus showing, one more time, that the Coffman-Sethi conjecture provides a tight bound for the 3-machine, 3-rank problem. If the size of a hypothesised minimal counterexample to the Coffman-Sethi conjecture can be shown to satisfy $m \leq 3$ and $k \leq 3$, then the above analysis would imply that the Coffman-Sethi conjecture holds for all problem instances.