TOTAL TARDINESS LOAD BALANCING IN PARALLEL MACHINES

Timur Keskintürk,
Department of Quantitative Methods, School of Business, Istanbul University,
Avcilar Campus, 34320 Istanbul, Turkey tkturk@istanbul.edu.tr

Mehmet B. Yildirim
Department of Industrial and Manufacturing Engineering
Wichita State University, Wichita, KS 67260, USA
bayram.yildirim@wichita.edu

Mehmet Barut
Department of Finance, Real Estate, and Decision Sciences
Barton School of Business, Wichita State University
Wichita, KS 67260, USA

ABSTRACT

In this paper, we introduce a scheduling problem which minimizes the average relative percentage of tardiness imbalance (ARPTI) in a parallel-machine environment. We propose a mixed integer programming formulation for ARPTI. Since this problem needs full enumeration to determine the optimal solution, we propose a metaheuristic approach and also several simple heuristics to obtain good solutions in reasonable amount of time. We present numerical results, and insights using computational experimentation.

Keyword: Optimization, Parallel Machines, Meta-heuristic, Ant Colony Optimization, Scheduling, Tardiness

INTRODUCTION

In this paper we present a mathematical model for a parallel machine problem where the goal is to minimize the average relative percentage of tardiness imbalance (ARPTI). Rajakumar et al. (2004, 2006, and 2007) reported that the minimization of total completion time imbalance may reduce idle time and work in process, maximize throughput, minimize the finished goods inventory, and lower operating expenses. The impact of the machine which has the highest workload, represents a bottleneck that precludes getting high system throughput.

The motivation of this paper is as follows: Jobs with due dates needs to be assigned to workers. At the end of the day, the performance measure is tardiness related, i.e., how many of jobs have been finished on time and total lateness. So it is important to assign jons to all workers in such a way that the performance of all workers are similar in tardiness and at an acceptable level. The goal of workload balancing is to distribute jobs/tasks to resources in such a way that the relative imbalance in total tardiness is minimized.

Workload balancing on parallel machines has been studied by various authors especially with total completion time objective. Rajakumar et al. (2006) studied load balancing on identical parallel machines with deterministic processing times. Aubry et al. (2008) considered the problem on parallel multi-purpose machines with dependent setup times. Yildirim et al. (2007) studied workload balancing on unrelated parallel machines in the presence of sequence-dependent setups. The researchers of this paper studied the workload-balancing problem using sequence-dependent setups, with the objective of minimizing total relative imbalance (Keskintürk et al., 2012).

The organization of the paper is as follows: In the next section, we present the notation utilized in this paper and propose a mixed-integer mathematical model that minimizes ARPTI. Sections 3 presents an ant colony optimization algorithm. The experimental setup is followed by computational experimentation.

MATHEMATICAL MODEL

. In parallel-machine scheduling problem with the minimum average relative percentage of total tardiness balancing, the following notation is utilized: K is the set of parallel machines, and $|K|$ denotes the number of machines, i.e., the cardinality of set K. Similarly, J is the set of jobs that needs to be processed on these machines. p_{ik} is the processing time of job i on machine k and d_i is the duedate of job i. All parameters are deterministic

Let T_k be the total tardiness of all jobs assigned to machine k. The imbalance is defined as $\mathrm{T}_{\max} - \mathrm{T}_k$, where the minimization objective will force $\mathrm{T}_{\max}$ to have the value of maximum tardiness of jobs on all parallel machines, i.e.,

$$\mathrm{T}_{\max} = \max_{k \in K} \mathrm{T}_k .$$

The relative imbalance on machine k is the ratio of imbalance and the maximum completion time on all machines, i.e.,

$$\text{relative imbalance}_k = \frac{\mathrm{T}_{\max} - \mathrm{T}_k}{\mathrm{T}_{\max}} .$$

Below is a mathematical program to minimize average relative percentage imbalance. The goal of the mathematical model for Parallel-Machine Workload Balancing with Tardiness Objective (PMWBT) is to schedule jobs on parallel machines to minimize the average relative percentage of imbalance (ARPI):

$$\min \left(\frac{1}{|K|} \sum_{k \in K} \frac{\mathrm{T}_{\max} - \mathrm{T}_k}{\mathrm{T}_{\max}} \right) * 100 \qquad (1)$$

Mathematically, the total tardiness on machine k, T_k is defined as

$$\mathrm{T}_k = \sum_{i \in J} \mathrm{T}_{ik}, \qquad k \in K$$

where $\mathrm{T}_{ik} = \max(0, d_i y_{ik} - c_{ik}), \qquad \forall k \in K, \forall i \in J$

In addition, the total completion time for a job can be calculated as

$$c_{jk} \geq c_{ik} + p_{ik} x_{ijk}, \qquad \forall k \in K, \forall i \in J$$

where

$$y_{ik} = \begin{cases} 1 & \text{if job } i \text{ is assigned to machine } k \\ 0 & \text{otherwise} \end{cases}$$

and

$$x_{ijk} = \begin{cases} 1 & \text{if job } i \text{ is the immediate predecessor of job } j \text{ on machine } k \\ 0 & \text{otherwise} \end{cases}$$

Constraint (3) ensures that the maximum workload/tardiness is greater than or equivalent to individual workloads.

$$T_{\max} \geq T_k \qquad k \in K \tag{3}$$

Constraint (4) ensures that each job is assigned to only one machine.

$$\sum_{k \in K_i} y_{ik} = 1 \qquad i \in J \tag{4}$$

Constraint (5) guarantees that a job cannot precede another job on machine *k* unless it has been assigned to machine *k*.

$$x_{ijk} \leq y_{ik} \qquad i \in J, k \in K_i, j \in J_k \tag{5}$$

Constraint (6)/constraint (7) ensures that a job must be before/after another job on a production line.

$$\sum_{i \in J_k} x_{ijk} \leq y_{ik} \qquad k \in K, j \in J_k \tag{6}$$

$$\sum_{j \in J_k} x_{ijk} \leq y_{ik} \qquad k \in K, i \in J_k \tag{7}$$

Constraint (8) represents sub-tour elimination constraints, which ensure that a job cannot be the immediate predecessor or successor of two or more different jobs at the same time.

$$\sum_{i \in J_k^{'}} \sum_{j \in J_k^{'}} x_{ijk} \leq | J_k^{'} | - 1 \qquad J_k^{'} \subseteq J_k \tag{8}$$

Note that any solution with a zero ARPI value is optimal. The PMWBT problem is closely related to the set partitioning problem: PMWBSDS partition jobs into subsets (i.e., assign jobs to machines) and then sequence the jobs in such a way that all machines have the same total tardiness value. However, one must determine the "correct completion time" while having a non-delay schedule while achieving a minimum relative imbalance. PMWBT has an exponential number of possible solutions. This motivated us to develop a metaheuristic solution approach, an ant colony optimization algorithm to determine good solutions in a reasonable amount of time. In the next section, we provide a summary of the proposed ant colony metaheuristic.

ANT COLONY OPTIMIZATION FOR TOTAL TARDINESS LOAD BALANCING IN PARALLEL MACHINES

Ant colony optimization (ACO) is a metaheuristic that solves complex optimization problems (Dorigo et al., 1996). ACO algorithm is inspired by ants' ability to find the shortest route between locations. At the beginning ants choose paths with equal chance and after a certain period of time, they converge to the optimal route. Since there is more pheromone on shorter routes compared to longer routes as a result of a higher number of trips occurring on shorter routes in unit time.

This paper uses a modified ant colony optimization by Keskinturk et al. (2012). The goal is to assign and sequence |*J*| jobs over |*K*| machines to minimize the total imbalance of tardiness on all machines. In our problem, using a metaheuristic, two decisions should be made: first, the assignment of jobs to machines should be determined, and then the order of jobs in order to obtain a good relative tardiness imbalance should be found.

The graph that is used in ACO is generated as follows: jobs are represented as supernodes (which can also be defined as node clusters). Each supernode has $|K|$ nodes, which represent the machines on which each job can be processed if visited. The nodes on each supernode are not connected. However, every node in a supernode is connected to all other nodes in other supernodes. An illustration for graph construction is given in Figure 1 (Keskinturk et al., 2012).

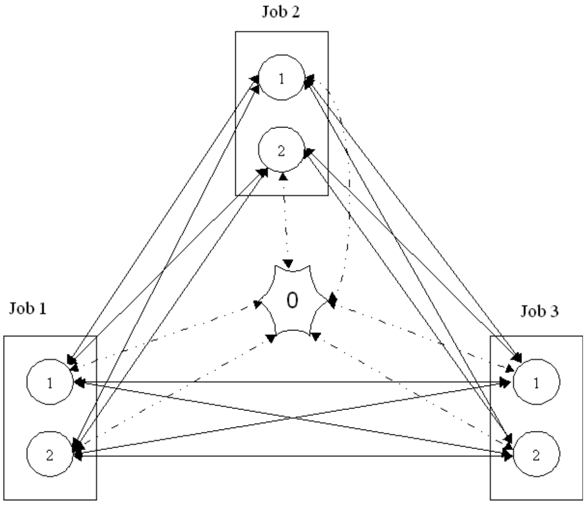

Figure 1. Network representation of load-balancing problem on ACO.

COMPUTATIONAL EXPERIMENTATION

The computational experimentation is performed on problems with two to six machines and 20 jobs. The processing time for each job has been generated from a uniform distribution, U[0, 20]+5. The duedates for each jobs are generated from a uniform distribution as follows:

$$d_i \in U\left[\frac{P\left[1-TF-\frac{RDD}{2}\right]}{|K|},\frac{P\left[1-TF+\frac{RDD}{2}\right]}{|K|}\right]$$

Where *P* is the total processing time, *TF* is the tardiness factor and *RDD* is the relative due date parameter.

We performed parameter optimization for Ant Colony Optimization metaheuristic, which resulted in a population size of 20, an initial pheromone level is 10^{-8}, and running the algorithm for 1000 iterations. Below, we present a summary of our findings running the algorithm with the proposed experimental design.

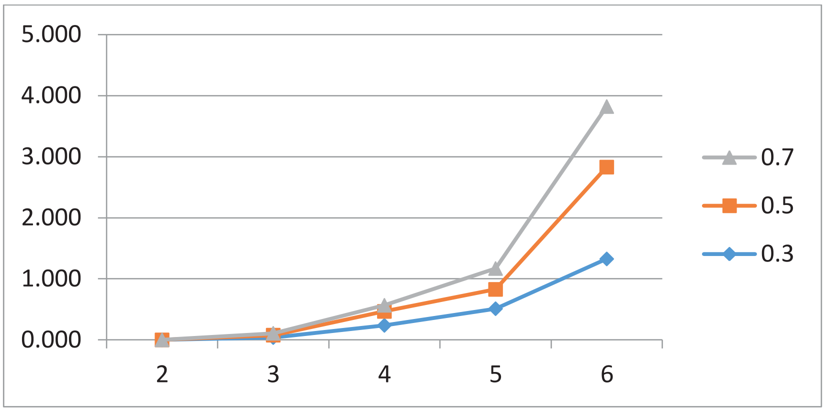

Figure 2. Tardiness Imbalance: machine number versus relative due date

Figure 2 shows the impact of relative due date (RDD) parameter on the tardiness imbalance. It is clear that when the RDD value increases from 0.3 to 0.7 the tardiness imbalance gets worse. This observation is also true for the number of parallel machines in the system. When the number of machines, the tardiness balance increase exponentially.

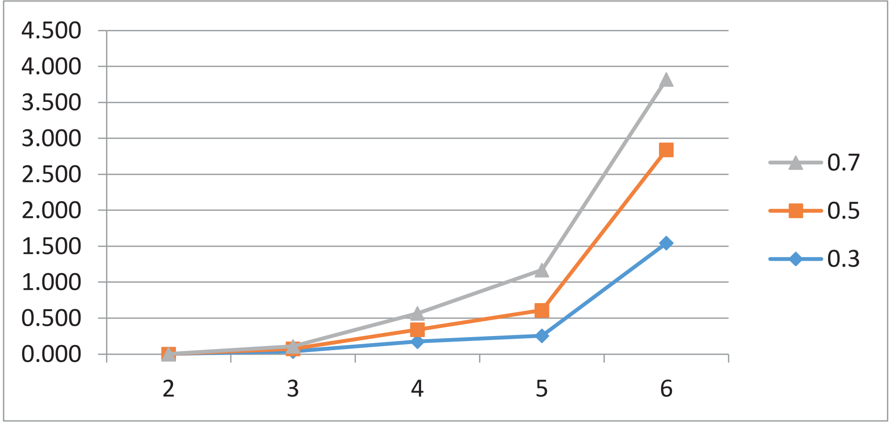

Figure 3. Tardiness Imbalance: machine number versus tardiness factor

Figure 3 summarizes the impact of tardiness factor (TF) on the imbalance. In this experimentation, the TF values are varied from 0.3 to 0.7. We have observed that when the number of machines or TF increase, the tardiness imbalance increase exponentially. This observation is similar to RDD observation. However, note that the magnitude of the imbalance increases more in varying TF values than the RDD values. This may be due to relative increase in variability of duedates, because TF value contributes twice compared to RDD in varying duedates,

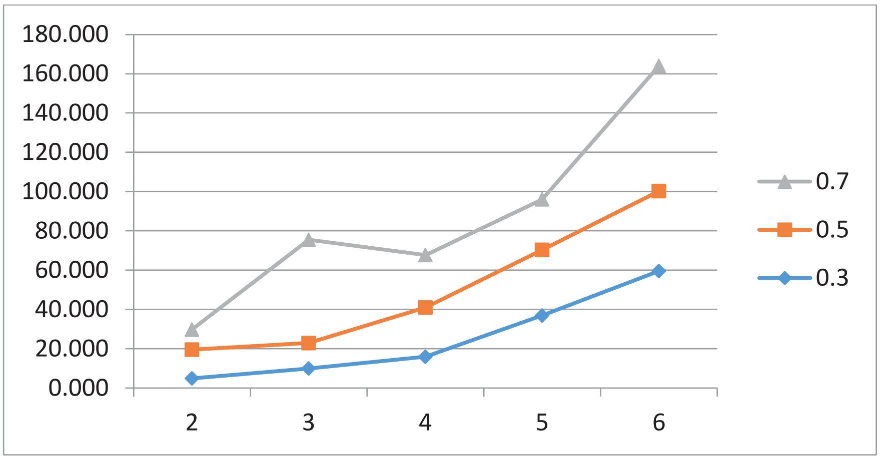

Figure 3. Completion time imbalance: machine number versus relative due date

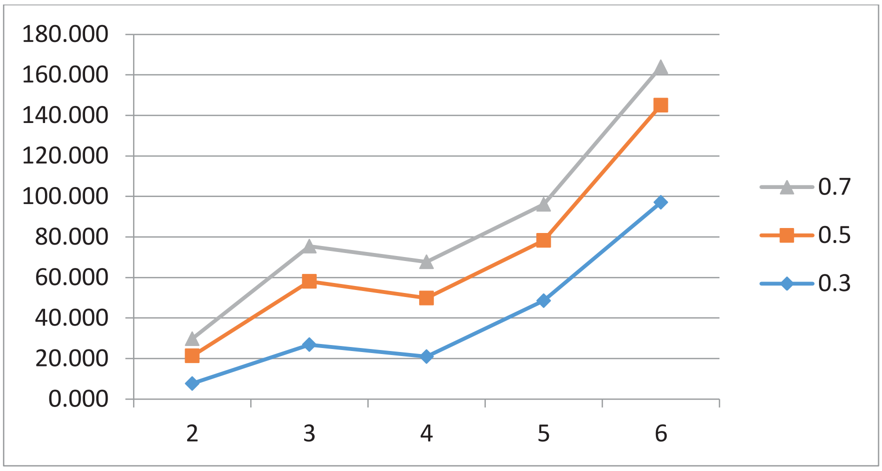

Figure 4. Completion time imbalance: machine number versus tardiness factor

In Fugure 3 and 4, we illustrate how total completion time imbalance changes for the solutions obtained by the tardiness imbalance objective. We observe that increasing the number of machines or the values for TF or RDD results in increasing completion time imbalance. Howver, this imbalance is more linear when compared to tardiness imbalance.

CONCLUSION

In this paper, we have proposed a new mathematical model for the tardiness imbalancing problem and developed an ant colony optimization algorithm to solve this problem in reasonable amount time. With computational experimentation, we illustrated the impact of different parameters on tardiness imbalance such as the number of machines, relative duedates and tardiness factor. Furthermore, we also showed how the imbalancing problem with total completion time objective for the solutions obtained with the imbalancing problem with total tardiness objective, and observed that good tardiness imbalance does not imply good completion time imbalance.

This study can be expanded in several ways: First of all, new dispatching rules and other metaheuristics can be proposed to solve this problem. In addition, the proposed metaheuristics can be improved by incorporating methods such as local search.

We also plan to incorporate the following constraint in our mathematical model to ensure that the total tardiness in any machine is within α *100% of the average tardiness, i.e.,

$$T_k \leq \frac{1}{|K|} T_{total}(1+\alpha) \text{ and } T_k \geq \frac{1}{|K|} T_{total}(1-\alpha) \text{ for } k \in K.$$

REFERENCES

Rajakumar, S. Arunachalam, V.P. & Selladurai V. 2004. Workflow balancing strategies in parallel machine scheduling. *International Journal of Advanced Manufacturing Technology*, 23, 366–374.

Rajakumar, S. Arunachalam, V.P. & Selladurai V. 2006. Workflow balancing in parallel machine scheduling with precedence constraints using genetic algorithm. *Journal of Manufacturing Technology Management*, 17 (2), 239–254.

Aubry, A. Rossi, A. Espinouse, M.L. & Jacomino, M. 2008. Minimizing setup costs for parallel multi-purpose machines under load-balancing constraint. *European Journal of Operations Research*, 187, 1115–1125

Yildirim MB, Duman E, Krishna K, Senniappan K. Parallel machine scheduling with load balancing and sequence dependent setups. International Journal of Operations Research 2007;4(1):1–8.

Keskinturk, T. Yildirim, M.B. & Barut, M. 2012. An ant colony optimization algorithm for load balancing in parallel machines with sequence-dependent setup times. *Computers & Operations Research*, 39 (6), 1225-1235.

Dorigo, M., Maniezzo, V. & Colorni, A. 1996. Ant System: Optimization by a colony of cooperating agents. IEEE Transactions on Systems, Man, and Cybernetics – Part B, 26 (1), 29–41.