Abstract number:025-0010

Abstract Tile: New Dispatching Rules to Minimize the Total Weighted Tardiness on the Single Machine

Author Information:

Yue Xi
Industrial Manufacturing Engineering Department
University of Wisconsin Milwaukee
2200 E Kenwood Blvd
Milwaukee, WI, U.S.A53202
Yuexi@uwm.edu
Tel: 414-426-8959

Jaejin Jang
Industrial Manufacturing Engineering Department
University of Wisconsin Milwaukee
2200 E Kenwood Blvd
Milwaukee, WI, U.S.A. 53202
Jang@uwm.edu
Tel: 414-229-2978

POMS 23rd Annual Conference
Chicago, Illinois, U.S.A
April 20 to April 23, 2012

New Dispatching Rules to Minimize the Total Weighted Tardiness on the Single Machine

Abstract

This paper classifies the sequence dependent setup into two categories: *the continuous sequence dependent setup* and *the separable sequence dependent setup*. The former is the conventional type considered by most research, while the latter allows the machine to be idle between setup and processing of the part. Two dispatching rules, Modified Apparent Tardiness Cost with Setup and Ready time (MATCSR) and Apparent Tardiness Cost with Separable Setup and Ready time (ATCSSR), are proposed for $1|r_j,s_{ij},con|\sum w_jT_j$ and $1|r_j,s_{ij},sep|\sum w_jT_j$ respectively. These two rules are compared with a few existing ATC based rules under two performance criteria by *the best of the best test* and *the territory test*.

Keyword: single machine, sequence dependent setup, total weighted tardiness

1. Introduction

The single machine scheduling problem is important and received much attention in the past. Even in a complex production system, production is often limited by a single bottleneck machine. Tardiness related criteria, such as tardiness, weighted tardiness, are relevant to penalties for not meeting predefined due dates. Based on Wisner and Siferd (1995), only 58% of industrial planners manage to meet delivery due dates.

Sequence dependent setup is often important in production. Examples can be found in petroleum production, printing, car spraying facilities, metallurgical and textile dying industries

(Luo *et al.* 2006, Arroyo *et al.* 2009).

This paper classifies the sequence dependent setup into *the separable sequence dependent setup* (*SSDS*) and *the continuous sequence dependent setup* (*CSDS*). If setup of a job can start before the job is ready for processing, it is separable sequence dependent setup; otherwise it is continuous sequence dependent setup. Most studies on the sequence dependent setup focus only on the continuous type; however, separable sequence dependent setup is also common. An example is product coloring, where the color change from a darker color to a lighter color takes longer time than in the opposite case. The setup for a new color does not require parts on a machine. Press die change and machine tooling change are examples at the machine level, and assembly line setup change is an example at the production line level. These setups do not need parts to be ready and are sequence dependent.

The Modified Apparent Tardiness Cost with Setup and Ready time (MATCSR) rule is proposed to minimize the total weighted tardiness of single machine scheduling with future ready time and continuous sequence dependent setup. This problem is donated as $1|r_j,s_{ij},con|\sum w_jT_j$, where *con* means the setup is the continuous type. For the same problem but with separable setup, the Apparent Tardiness Cost with Separable Setup and Ready time (ATCSSR) rule is proposed. This problem is denoted as $1|r_j,s_{ij},sep|\sum w_jT_j$, where *sep* means the setup is the separable type. When a job is completed, both rules select a waiting job to process next. MATCSR allows a machine to be idle only before the setup, while ATCSSR allows a machine to remain idle after the setup (*e.g.*, the setup of the selected job starts at the decision time and can be finished before this selected job becomes ready).

This paper is organized as follows: in section 2, we describe the problems. The Apparent

Tardiness Cost based (ATC-based) dispatching rules are reviewed in section 3. Section 4 introduces the MATCSR rule. Section 5 evaluates the performance of MATCSR. In section 6, we introduce the ATCSSR rule. Section 7 evaluates the performance of ATCSSR. Conclusions and future research are discussed in section 8.

2. Problem description

This paper considers two problems: $1|r_j,s_{ij},con|\sum w_jT_j$ and $1|r_j,s_{ij},sep|\sum w_jT_j$. The first problem is stated as: there are n jobs arriving to a continuously available single machine at different times. Each job j has its ready time (r_j), processing time (p_j), due date (d_j), and job weight (w_j). The setup (s_{ij}) of each pair of jobs i and j is continuous sequence dependent. The objective is minimizing the total weighted tardiness of jobs, $\sum_{j=1}^{n} w_jT_j$, T_j is the tardiness of job j, max$\{0,C_j\text{-}d_j\}$, where C_j is the competition time of job j. The second problem is the same as the first problem except that the setup is the separable type. Both problems assume the following:

- The job attributes (p_j, d_j, w_j, r_j, s_{ij}) are known in advance.
- The machine can process at most one job at a time.
- Job preemption is not allowed.
- Interruption such as machine breakdown and order cancellation does not happen.

3. Literature review

Minimizing the total weighted tardiness on the single machine is NP hard. Compared to implicit enumeration algorithms (*e.g.*, dynamic programming and branch and bound method) and interchange methods (*e.g.*, tabu search, genetic algorithm and simulated annealing), the dispatching procedure is a method that creates a schedule by scheduling a job in a position at each step (Wodecki 2008). At each decision time, the dispatching procedure selects a job from the waiting jobs based on a certain priority index.

Composite dispatching rules, such as ATC-based rules, combine several factors or rules to make schedules. To minimize the total weighted tardiness on a single machine, Rachamadugu and Morton (1982) first propose the Apparent Tardiness Cost (ATC) rule whose index combines the weighted shortest processing time rule and the least slack rule. The ATC index of an unprocessed job *j* is:

$$I_{ATC}(t,j) = \frac{w_j}{p_j}\exp(-\frac{\max(d_j - p_j - t,0)}{k\overline{p}}) \tag{1}$$

where t and $\overline{p}$ are the decision time and the average processing time, respectively. The scaling parameter k is suggested to be set at 2 in a static flow shop and a single machine shop. The unprocessed job with the largest index value is selected.

Considering the sequence dependent setup, the Apparent Tardiness Cost with Setups (ATCS) rule (Lee *et al.* 1997, Lee and Pinedo 1997) adds the shortest setup rule to the ATC rule.

$$I_{ATCS}(t,i,j) = \frac{w_j}{p_j}\exp(-\frac{\max(d_j - p_j - t,0)}{k_1\overline{p}})\exp(-\frac{s_{ij}}{k_2\overline{s}}) \tag{2}$$

where i and $\overline{s}$ are the last scheduled job and the average setup time, respectively, and k_2 is the scaling parameter of the setup term.

Vepsalainen and Morton (1987) extend the ATC rule for a job shop.

$$I_{VM}(t,i,j) = \frac{w_j}{p_{ij}}\exp(-\frac{\max(d_i - p_{ij} - t - \sum_{q=j+1}^{m_i}(W_{iq} + p_{iq}),0)}{k\overline{p}}) \tag{3}$$

W_{iq} and p_{iq} are the estimated waiting time and the processing time of the unprocessed operation q of job i, respectively, and m_i is the number of remaining operations of job i. In the index, W_{iq} is estimated by bp_{iq} (Carroll 1965, Conway *et al.* 1967), where the lead time estimation parameter b reflects the anticipated machine utilization. In the index of $I_{VM}(t,i,j)$, b is set at 2 and k is suggested to be 3.

Morton and Pentico (1993) propose the X-RM heuristic to schedule the single machine problems with future job ready time, r_j. A ready time term is added to ATC rule.

$$I_{X-RM}(t,j) = \frac{w_j}{p_j}\exp(-\frac{\max(d_j - p_j - t,0)}{k\overline{p}})(1-\frac{B}{p_{\min}}\max(r_j - t,0)) \tag{4}$$

In the index, p_{min} is the minimum processing time of all waiting jobs, and B is a scaling parameter to adjust the ready time term. In the index of $I_{X\text{-}RM}(t,j)$, B is set at 1.3 + ρ, where ρ is the machine utilization.

X-RMmod (Pfund *et al.* 2008) is a modification of X-RM and considers the sequence dependent setup.

$$I_{X-RM\,\mathrm{mod}}(t,i,j) = \frac{w_j}{p_j}\exp(-\frac{\max(d_j - p_j - t,0)}{k_1\overline{p}})\exp(-\frac{s_{ij}}{k_2\overline{s}})(1-\frac{B}{p_{\min}}\max(r_j - t,0)) \tag{5}$$

Apparent Tardiness Cost with Setups and Ready times (ATCSR) (Pfund *et al.* 2008) is one of the most effective dispatching rules to solve a dynamic problem with sequence dependent setup in minimizing the total weighted tardiness. A new slack term and a new ready time term are proposed in the index. ATCSR uses three scaling parameters, k_1, k_2 and k_3.

$$I_{ATCSR}(t,i,j) = \frac{w_j}{p_j}\exp(-\frac{\max(d_j - p_j - \max(r_j,t),0)}{k_1\overline{p}})\exp(-\frac{s_{ij}}{k_2\overline{s}})\exp(-\frac{\max(r_j - t,0)}{k_3\overline{p}}) \tag{6}$$

In batch production where several jobs are processed together at the same time by a machine, Mason *et al.* (2002) propose Batch Apparent Tardiness Cost with Setups (BATCS). At the decision time, the jobs with the highest index values are selected together to form a batch.

$$I_{BATCS}(t,i,j) = \frac{w_j}{p_j}\exp(-\frac{\max(d_j - p_j + r_j - t,0)}{k_1\bar{p}})\exp(-\frac{s_{ij}}{k_2\bar{s}}) \tag{7}$$

Pfund *et al.* (2008) modify the slack term of BATCS. The index of BATCSmod is:

$$I_{BATCS\mathrm{mod}}(t,i,j) = \frac{w_j}{p_j}\exp(-\frac{\max(d_j - p_j + \max(r_j - t,0),0)}{k_1\bar{p}})\exp(-\frac{s_{ij}}{k_2\bar{s}}) \tag{8}$$

Table 1 summarizes indexes of ATC based dispatching rules. All rules with setup time s_{ij} consider only the continuous sequence dependent setup; none of them considers the separable sequence dependent setup.

Table 1. ATC based dispatching rules

Authors	**Rule name**	**Future ready Time**	**Number of parameters**	**Environment**
Rachamadugu and Morton (1982)	ATC	No	1 (k)	Single machine
Lee *et al.* (1997) Lee and Pinedo (1997)	ATCS	No	2 (k_1 and k_2)	Single/Parallel machines
Vepsalainen and Morton (1987)	MATC	No	2 (b and k)	Flow shop , Job shop
Morton and Pentico (1993)	X-RM	Yes	2 (B and k)	Single machine
Pfund *et al.* (2008)	X-RMmod	Yes	3 (B, k_1 and k_2)	Parallel machines
Mason *et al.* (2002)	BATCS	Yes	2 (k_1 and k_2)	Batch machine
Pfund *et al.* (2008)	BATCSmod	Yes	2 (k_1 and k_2)	Batch machine
Pfund *et al.* (2008)	ATCSR	Yes	3 (k_1, k_2 and k_3)	Parallel machines

4. Modified Apparent Cost Tardiness with Setup and Ready time (MATCSR)

This section introduces the new index of MATCSR to minimize the total weighted tardiness on the single machine with continuous sequence dependent setup and future ready time.

The proposed index of MATCSR is:

$$I_{MATCSR}(t,i,j) = \frac{w_j}{p_j + s_{ij} + \max(r_j - t,0)} \exp(-\frac{\max(d_j - p_j - s_{ij} - \max(r_j,t),0)}{k_1\bar{p}}) \exp(-\frac{s_{ij}}{k_2\bar{s}}) \exp(-\frac{\max(r_j - t,0)}{k_3\bar{p}}) \qquad (9)$$

This index has four terms. The first term is the modified WSPT term. A job with a larger job weight or shorter processing time is given a higher priority. In this term, we include the sequence dependent setup time and the time from the decision time to the job's ready time as if they are a part of the processing time because, if we process job *j* next, other jobs' earliest possible start times are delayed by p_j+s_{ij}+max(r_j-t,0). The second term is the slack term. The slack is the maximum time by which the start of the setup of a job can be delayed from its earliest possible start time, max(r_j,t), without violating its due date. The last two terms, the setup term and the ready time term, are the same as those of ATCSR.

5. Performance evaluation of MATCSR

This section evaluates the performance of the proposed MATCSR for the continuous sequence dependent setup.

5.1. Experiment design

The evaluation uses the grid approach, which is also used by Park *et al.* (2000), Pfund *et al.* (2008) and Rene and Lars (2009). The grid approach of the ATC based rules generates schedules by using a dispatching index for given combinations of *k* values that correspond to the grid points, and then chooses the best schedule among them as the final schedule. The grid method gives a good solution in a given range of the scaling parameters; its disadvantage is in deciding the search range and grid size. Also, the searching time can be long if a large search space and a small grid size are used.

The performance of MATCSR is compared with those of BATCS, BATCSmod and ATCSR. When BATCS and BATCSmod are used as benchmark methods, their batch size is set at 1. Both ATCSR and MATCSR use three *k* values, while BATCS and BATCSmod use only two *k* values, k_1 and k_2. In the experiment, we adopt the grid values of k_1, k_2 and k_3 suggested by Pfund *et al.* (2008).

Set k_1: {0.2,0.6,0.8,1,1.2,1.4,1.6,1.8,2,2.4,2.8,3.2,3.6,4,4.0,4.4,4.8,5.2,5.6,6,6.4,6.8,7.2}

Set k_2: {0.1,0.3,0.5,0.7,0.9,1.1,1.3,1.5,1.7,1.9,2.1}

Set k_3: {0.001,0.0025,0.004,0.005,0.025,0.04,0.05,0.25,0.25,0.4,0.6,0.8,1,1.2}

For each scheduling problem, each of MATCSR and ATCSR makes 3146 (22 x 11 x 13) grids and as many schedules. For BATCS and BATCSmod, 242 (22 x 11) grids and schedules are made. The schedule with the smallest total weighted tardiness among 3146 or 242 schedules is selected as the corresponding rule's final schedule of the problem.

The test data of each problem is generated by the method of Pfund *et al.* (2008). In the experiment, we use all factors of Pfund except for the job machine factor μ, which is meaningful only for problems of parallel machines. Each of the remaining five factors has three levels. The experiment is a 3^5 experiment, which has 243 (3^5) scenarios. In each scenario, seven cases, or problems, are randomly generated. In total, 1701 (243 x 7) problems are considered in the performance evaluation. Each problem has 40 jobs. Table 2 shows the factors and levels of the experiment. We perform two types of tests, *the best of the best test* and *the territory test*, which are explained in more detail in sections 5.3 and 5.4.

Table 2. Factors and levels of the performance test

Factor Number	Notation	Factor name	Low level	Center level	High level
1	η	Setup severity factor	0.02	1.01	2
2	ז	Due date tightness factor	0.3	0.6	0.9
3	R	Due date range factor	0.25	0.63	1
4	J_a	Job availability factor	0.2	0.5	0.8
5	r_ז	Ready time factor	1	5.5	10

5.2. *The best of the best test*

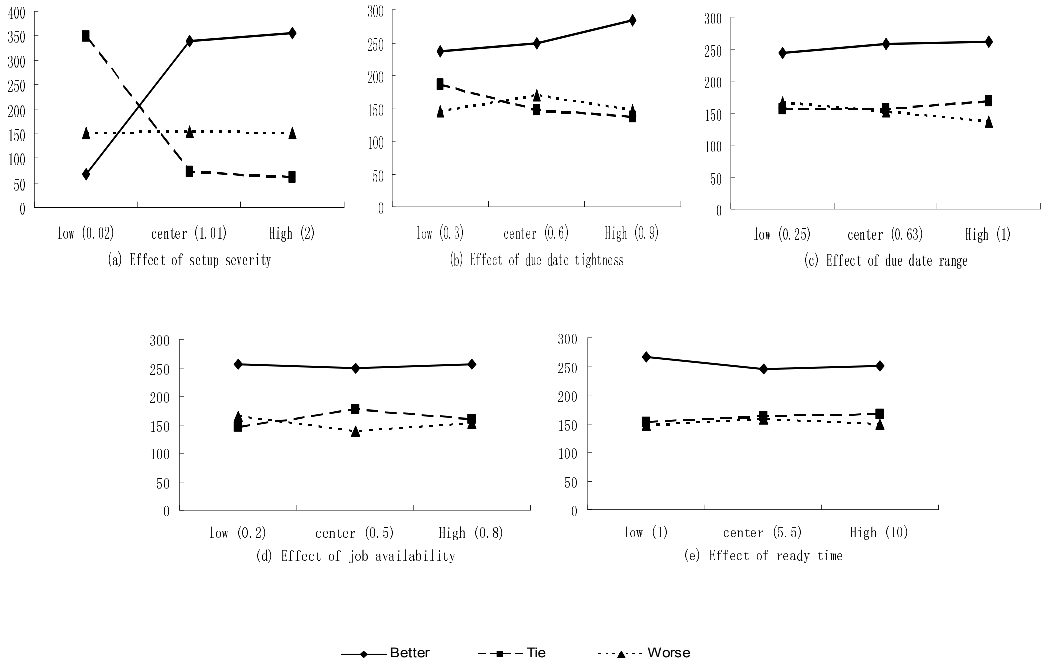
The grid approach generates multiple schedules for each problem and selects the best one as the final schedule. We compare the selected best schedules, one from each scheduling method, and check which method gives the best schedule among the selected best schedule. We call this test *the best of the best test*.

Table 3 compares BATCS, BATCSmod, ATCSR and MATCSR rules for the 1701 problems. The sum of percentage in the table is more than 100% because of ties. It shows MATCSR yields more best cases than ATCSR, which significantly outperforms BATCS and BATCSmod.

Table 3. Performance comparison for the continuous setup case: the best of the best test

Method	Number of best cases	Percentage
MATCSR	1220	71.72
ATCSR	916	53.85
BATCS	89	5.23
BATCSmod	2	0.12

We next compare the two best methods in more detail. The result shows MATCSR outperforms ATCSR at all levels of all factors except for factor 1, setup severity. (figure 1) When the level of factor 1 is low (the average setup time is short, *i.e.*, 2% of the average processing time), ATCSR outperforms MATCSR and shows mostly the same tardiness level as MATCSR.



(a) Effect of setup severity

(b) Effect of due date tightness

(c) Effect of due date range

(d) Effect of job availability

(e) Effect of ready time

Figure 1. Effect of each factor by MATCSR (compared to ATCSR, y axis: number of cases)

5.3. *The territory test*

To compare MATCSR and ATCSR, we perform *the territory test* as follows: for each problem, the results from MATCSR and ATCSR are compared at each of all 3146 grid points and the percentages of grid points at which MATCSR performs better than, equal to and worse than ATCSR are recorded. The test uses the average values of the percentages over the seven problems of each scenario. This analysis is important in relation to the regression method because the *k* values estimated by the regression method are somewhat away from the best grid point in a random fashion. When the best *k* values are not estimated very accurately, the scheduling method with a larger favorable territory is likely to give a better schedule. (To estimate the values of ATC based rules, the regression approach (Kim *et al.* 1995, Lee and Pinedo 1997, Park and Lee 2000, Pfund *et al.* 2008) develops regression equations to estimate the best *k* values by using certain known parameters of the scheduling problems.)

Figure 2 shows that when setup severity is at the high or center levels (scenarios 1 to 81 or scenarios 82 to 161, respectively), MATCSR performs significantly better than ATCSR. When setup severity level is low (scenarios 163 to 243), many grid points yield ties. It is also noticed that when the setup severity level is low and due date range tightness is central or low (scenarios 190 to 243), MATCSR outperforms ATCSR; only when the setup severity level is low and due date tightness level is high, ATCSR slightly outperforms MATCSR (scenarios 163 to 189).

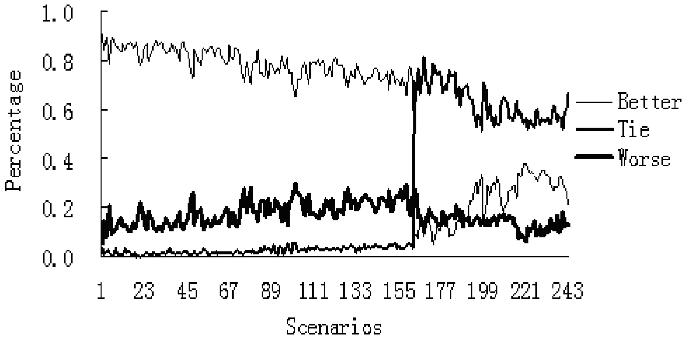


Figure 2. Territory test (MATCSR vs. ATCSR)

5.4. The effect of the modified WSPT term in MATCSR

Equation (9) introduced a new term, s_{ij}+max(r_j-t,0), in the WSPT term. To test the significance of this term, two versions of MATCSR with and without this new term are compared with ATCSR for the same 1701 problems used in section 5.1. The result of *the best of the best test* summarized in figure 3 shows the importance of the new term. Without it, the number of better cases is reduced and the number of tie cases is increased significantly.

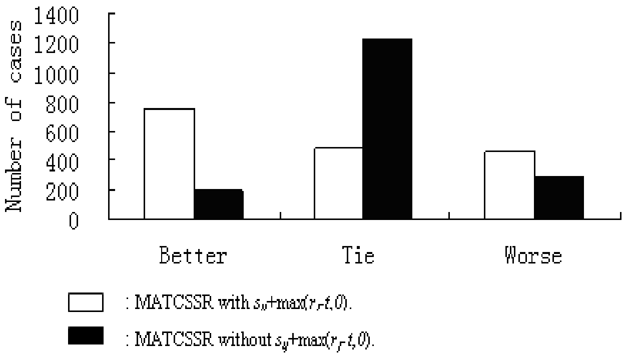


Figure 3. Significant of new term, s_{ij}+max(r_j-t,0), in MATCSR

6. Apparent Tardiness Cost with Separable Setup and Ready Time (ATCSSR)

This section introduces the new ATCSSR scheduling index to minimize the total weighted tardiness on the single machine with separable sequence dependent setup and unequal future ready time.

The proposed index of ATCSSR is:

$$I_{ATCSSR}(t,i,j) =$$

$$\frac{w_j}{p_j + \max(s_{ij}, r_j\text{-}t)} \exp(-\frac{\max(d_j - p_j - \max(r_j, t + s_{ij}), 0)}{k_1\bar{p}}) \exp(-\frac{s_{ij}}{k_2\bar{s}}) \exp(-\frac{\max(r_j - t - s_{ij}, 0)}{k_3\bar{p}})$$

(10)

This index has four terms. The first term is the modified WSPT term. A job with larger job weight or shorter processing time gets a higher priority. In this term, we include the machine idle time (r_j-t) or the sequence dependent setup time (s_{ij}) as a part of the processing time because, if we process job j next, other jobs' earliest possible start times are delayed by p_j+max(s_{ij}, r_j-t). The term max(s_{ij},r_j-t) is the time from the current time (t) to the earliest possible start time of job j after the setup. The second term is the slack term. A job with a smaller slack gets a higher priority. The third term is the setup term, which is the same as that of MATCSR. The last term is the ready time term. If a job's setup is finished after its ready time ($r_j<t+s_{ij}$) the job gets the highest priority value from the ready time term. Otherwise, the priority is reduced by the difference between the ready time (r_j) and earliest setup completion time ($t+s_{ij}$).

7. Performance evaluation of ATCSSR

The experiment design of ATCSSR is the same as that of MATCSR except that the setup is the separable type. Due to the lack of benchmark methods for the separable sequence dependent setup case, BATCS, BATCSmod and ATCSR are modified and compared with ATCSSR: if a future job is selected next, its setup is allowed to start as soon as the last scheduled job is finished.

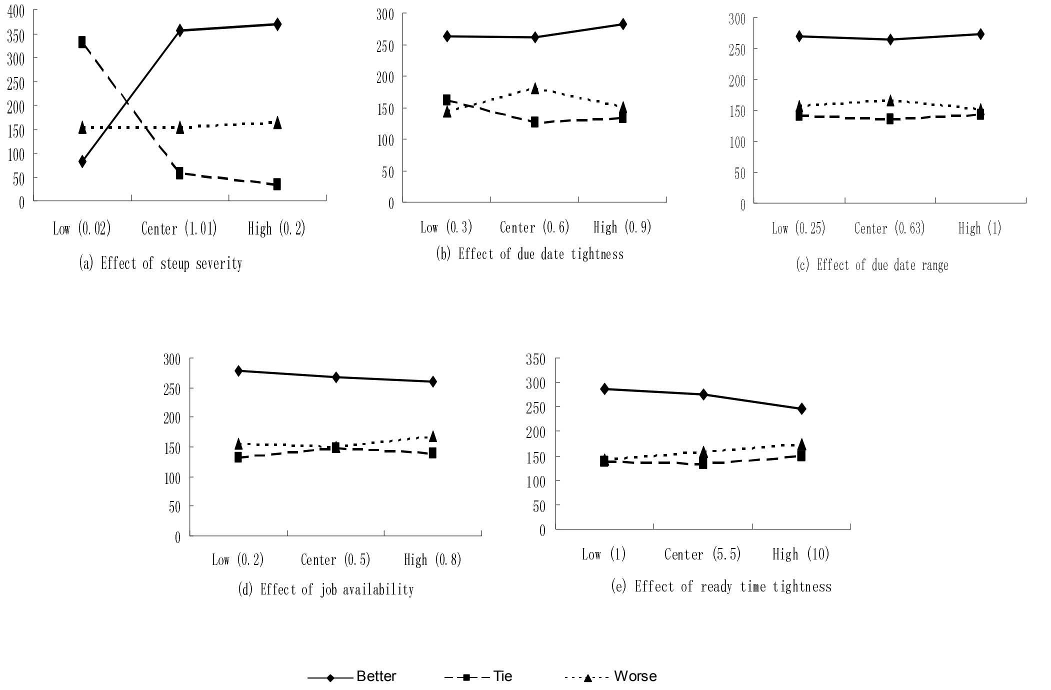
7.1. *The best of the best test*

Table 4 compares BATCS, BATCSmod, ATCSR and ATCSSR for the 1701 problems mentioned above. It shows ATCSSR yields more number of best cases than ATCSR, which significantly outperforms BATCS and BATCSmod.

Table 4. Performance comparison for the separable setup case: the best of the best test

Method	**Number of best cases**	**Percentage**
ATCSSR	1204	70.78
ATCSR	871	51.21
BATCS	89	5.23
BATCSmod	1	0.06

We next compare the two best methods in more detail. The result shows ATCSSR outperforms ATCSR at all levels of all factors except for factor 1, setup severity (figure 4). When the level of factor 1 is low (average setup time is short, *i.e.*, 2% of the average processing time), ATCSR slightly outperforms ATCSSR and shows mostly the same tardiness level as ATCSR.



(a) Effect of steup severity

(b) Effect of due date tightness

(c) Effect of due date range

(d) Effect of job availability

(e) Effect of ready time tightness

Figure 4. Effect of each factor by ATCSSR (compared to ATCSR, y axis: number of cases)

7.2. *The territory test*

We also perform *the territory test*. Figure 5 shows that when setup severity is at the high or center levels (scenarios 1 to 81 or scenarios 82 to 162, respectively), ATCSSR performs significantly better than ATCSR. When setup severity level is low (scenarios 163 to 243), many grid points yield ties. It is also noticed that when the setup severity level is low and due date tightness level is central or low (scenarios 190 to 243), ATCSSR outperforms ATCSR; only when the setup severity factor level is low and due date tightness level is high, ATCSR outperforms ATCSSR (scenarios 163 to 189).

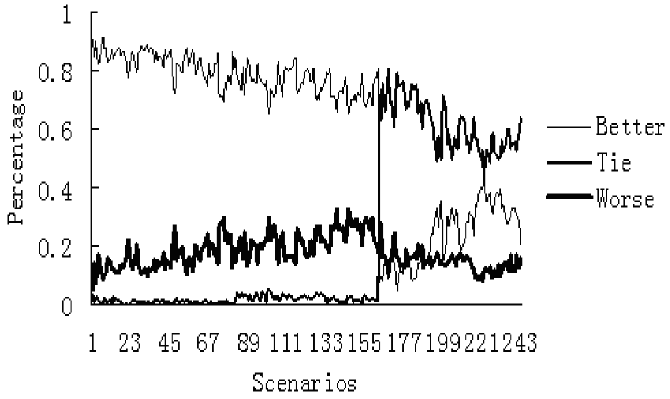


Figure 5. Territory test (ATCSSR vs. ATCSR)

7.3. The effect of the modified WSPT term in ATCSSR

Equation (6) introduced a new term, max(s_{ij},r_j-t), in the WSPT term. To test the significance of this term, two versions of ATCSSR with and without this new term are compared with ATCSR for the same 1701 problems generated in section 5.1. The result of *the best of best test* summarized in figure 6 shows the importance of the new term. Without it, the number of better cases is reduced and the number of tie cases is increased significantly.

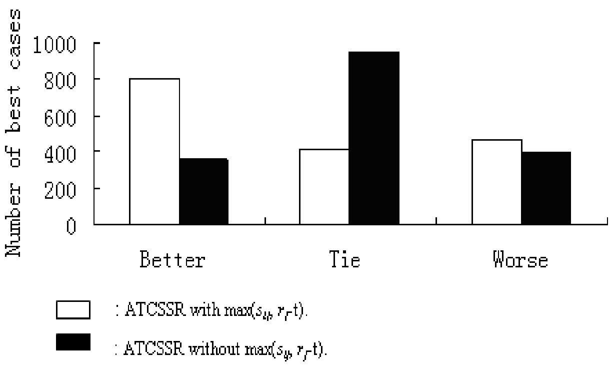


Figure 6. Significant of new term, max(s_{ij},r_j-t), in MATCSR and ATCSSR

8. Conclusions and future research

The sequence dependent setup is classified into *CSDS* and *SSDS* depending on whether or not setup is allowed to start before the ready time of a part. MATCSR is proposed for the *CSDS*. Two types of performance tests, *the best of the best test* and *the territory test*, show MATCSR outperforms other ATC based rules, BATCS, BATCSmod and ATCSR, in reducing the total weighted tardiness on a single machine. The term s_{ij}+max(r_j-t,0) in the WSPT term of MATCSR is shown to be important. ATCSSR is proposed for the *SSDS*. It outperforms benchmark methods in minimizing the total weighted tardiness on a single machine. The term max(s_{ij},r_j-t) in the WSPT term of ATCSSR is also shown to be important. For both *the best of the best test* and *the territory test*, the improvement of MATCSR (for continuous setup) and ATCSSR (for separable setup) over ATCSR shows a similar pattern.

One of directions for future research is study the performance of proposed rules under more complicated environments, such as parallel machines, flow shop, or job shop. Also, development of regression equations of the scaling parameters is another topic for the future study.

References

Arroyo, J. E. C., Nunes, G. V. P., and Kamke, E. H., 2009. Iterative local search heuristic for the single machine scheduling problem with sequence dependent setup times and due dates. *Proceedings of the Ninth international conference on hybrid intelligent systems* (Shenyang, China, Aug 2009), **104**, 505-510.

Chang, T. Y., Chou, F. D., and Lee, C. E., 2004. A heuristic algorithm to minimize total weighted tardiness on a single machine with release dates and sequence-dependent setup times, *Journal of the Chinese institute of industrial engineers*, **21** (3), 289-300.

Carroll, D. C., 1965. *Heuristic sequencing of jobs with single and multiple Components*. PhD Thesis, Sloan school of management, MIT.

Conway, R. W., Maxwell, W. L., and Miller, L. W., 1967, *Theory of Scheduling*, (Reading, MA: Addison Wesley).

Kim, S.Y., Lee, Y. H., and Agnihotri, D., 1995. A hybrid approach to sequencing jobs using heuristic rules and neural network. *Production planning and Control*, **6** (5), 445–454.

Lee, Y. H., Bhaskaran, K., and Pinedo, M., 1997. A heuristic to minimize the total weighted tardiness with sequence-dependent setups. *IIE Transactions,* **29**, 45–52.

Lee, Y. H., and Pinedo, M., 1997. Scheduling job on parallel machines with sequence dependent setup time. European Journal of Operational Research 100, 464-474.

Luo, X., and Chu F., 2006, A branch and bound algorithm of the single machine schedule with sequence dependent setup times for minimizing total tardiness. *Applied mathematic and computation*, **183**, 575-588.

Mason, S. J., Fowler, J. W., and Carlyle, W. M., 2002. A modified shifting bottleneck heuristic for minimizing total weighted tardiness in complex job shops. *Journal of Scheduling,* **5** (3), 247–262.

Morton, T. E., and Pentico, D., 1993. *Heuristic Scheduling Systems: With Applications to Production Systems and Project Management*, (New York: Wiley).

Park, Y., Kim, S., and Lee, Y. H., 2000. Scheduling jobs on parallel machines applying neural network and heuristic rules. *Computers & Industrial Engineering,* **38**, 189–202.

Pfund, M. E., Fowler, J. W., Gadkari, A., and Chen, Y., 2008. Scheduling jobs on parallel machines with setup times and ready time. *Computers & Industrial Engineering,* **54** (4), 764–782.

Rachamadugu, R.V., and Morton, T. E., 1982. Myopic heuristics for the single machine weighted tardiness problem. Working Paper, Carnegie Mellon University, Pittsburgh, PA 30-82-83.

Rene, D., and Lars, M., 2009. Scheduling jobs on parallel machines with sequence dependent setup times, precedence constraints, and ready times using variable neighborhood search. *IEEE*, 273-278.

Vepsalainen, A., and Morton, T. E., 1987. Priority rules for job shops with weighted tardiness costs. *Management Science,* **33**, 1035–1047.

Wodecki, M., 2008. A branch and bound parallel algorithm for single machine total weighted tardiness problem. *International journal of advance manufacturing technologies*, **37**, 996-1004.

Wisner, J. D., and Siferd, S. P., 1995. A survey of LIS manufacturing practices in make-to-order machine shops. *Production and Inventory Management Journal,* **1**, 1-7.