International Journal of Production Research
Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/tprs20

Heuristic dispatching rule to maximize TDD and IDD performance

T. F. Ho [a] & R. K. Li [a]

[a] Department of Industrial Engineering and Management , National Chiao Tung University , 1001 Ta Hseuh Road, Hsinchu, Taiwan 30050, ROC

[b] Department of Industrial Engineering and Management , National Chiao Tung University , 1001 Ta Hseuh Road, Hsinchu, Taiwan 30050, ROC E-mail:

Published online: 22 Feb 2007.

To cite this article: T. F. Ho & R. K. Li (2004) Heuristic dispatching rule to maximize TDD and IDD performance, International Journal of Production Research, 42:24, 5133-5147

To link to this article: http://dx.doi.org/10.1080/00207540420002704021

INT. J. PROD. RES., 15 DECEMBER 2004,
VOL. 42, NO. 24, 5133–5147

Heuristic dispatching rule to maximize TDD and IDD performance

T. F. HO† and R. K. LI†*

Although mean flow time and tardiness have been used for a long time as indicators in both manufacturing plants and academic research on dispatching rules, according to Theory of Constraints (TOC), neither indicator properly measures deviation from production plans. TOC claims that using throughput dollar-day (TDD) and inventory dollar-day (IDD) can induce the factory to take appropriate actions for the organization as a whole, and that these can be applied to replace various key performance indices used by most factories. However, no one has studied dispatching rules based on TDD and IDD performance indicators. The study addresses two interesting issues. (1) If TDD and IDD are used as performance indicators, do those dispatching rules that yield a better performance in tardiness and mean flow time still yield satisfactory results in terms of TDD and IDD performance? (2) Does a dispatching rule exist to outperform the current dispatching rules in terms of TDD and IDD performance? First, a TDD/IDD-based heuristic dispatching rule is developed to answer these questions. Second, a computational experiment is performed, involving six simulation examples, to compare the proposed TDD/IDD-based heuristic-dispatching rule with the currently used dispatching rules. Five dispatching rules, shortest processing time, earliest due date, total profit, minimum slack and apparent tardiness cost, are adopted herein. The results demonstrate that the developed TDD/IDD-based heuristic dispatching rule is feasible and outperforms the selected dispatching rules in terms of TDD and IDD.

1. Introduction

Due date and cycle time (or work-in-progress (WIP), because cycle time and WIP are positively correlated) are two crucial elements in production schedule planning and control (Tsai *et al*. 1997, Tseng *et al*. 1999, Philipoom 2000). Dispatching rules such as shortest processing time (SPT) and earliest due date (EDD) seek to minimize mean flow time and tardiness. Although mean flow time and tardiness have been adopted as indicators for a long time in both manufacturing plants and academic research on dispatching rules, according to Theory of Constraints (TOC), neither indicator can properly measure deviation from production plans (Goldratt 2000).

TOC claims that missing an order for which a customer is willing to pay US$10 000 does not have the same effect as missing an order for which a customer is willing to pay $100, and that missing by one day is not equivalent to missing by one month. Therefore, using the tardiness performance index neglects the value of the missing order and the number of days by which the order is missed. TOC implies that both the dollar value and the number of days by which the order is missed

Revision received July 2004.

†Department of Industrial Engineering and Management, National Chiao Tung University, 1001 Ta Hseuh Road, Hsinchu, Taiwan 30050, ROC.

*To whom correspondence should be addressed. e-mail: rkli@cc.nctu.edu.tw

International Journal of Production Research ISSN 0020–7543 print/ISSN 1366–588X online © 2004 Taylor & Francis Ltd
http://www.tandf.co.uk/journals
DOI: 10.1080/0020754042000270421

should be considered. Based on this concept, TOC proposes a new due date performance indicator, throughput dollar-day (TDD).

Using the TDD, a unit such as a plant can measure its performance with respect to meeting due dates of orders by assigning to every missed order a value equal to its selling price multiplied by the number of days by which the shipment is already late. A summation over all the missed orders gives the plant an objective measure of its performance, at any time. This measure forces a plant to concentrate on the extremely high TDD orders. A high TDD order implies either the ordinary price orders are extremely tardy or the high price orders are only slightly tardy. The higher is the TDD summation, the worse is the performance in factory. The target TDD is zero.

TOC also claims that when the mean flow time or days of WIP are used to measure the cycle time or the WIP, either the average cycle time or the average WIP only are considered, and the value of the WIP is missed. TOC suggests that the value of the WIP and the time since the order entered the plant should be considered. TOC proposes a new WIP or cycle time performance indicator, inventory dollar-day (IDD). IDD is the summation of the dollar value of the WIP multiplied by the time since the WIP entered the plant. The target here is not zero, but a lower value is preferred.

In fact, TDD and IDD have been extensively used in TOC application environments and both have been proven to have advantages over tardiness and mean flow time indicators (Goldratt 2000). TDD and IDD can be adopted to replace various key performance indices used by most factories. However, as far as we know, no one has yet studied dispatching rules in terms of TDD and IDD performance indicators. Two interesting issues remain to be addressed. (1) Are dispatching rules that claim better performance in terms of tardiness and mean flow time still better in terms of TDD and IDD performance? (2) Does a dispatching rule exist to outperform current dispatching rules in terms of TDD and IDD performance?

First, a TDD/IDD-based heuristic dispatching rule is developed to answer both questions. Second, a computational experiment with six simulation examples is developed to compare the present TDD/IDD-based heuristic dispatching rule with selected current dispatching rules. Five dispatching rules — SPT, EDD, total profit, minimum slack and apparent tardiness cost (ATC) — are considered.

2. TDD/IDD-based dispatching rule

Let:

TDD_i index represents throughput dollar-day of order i,
IDD_i index represents inventory dollar-day of order i,
W_iT_i index represents the weighted tardiness of order i,
W_iF_i index represents the weighted flow time of order i,
P_{ij} Process time of order i on machine j,
ID_{ij} idle time of order i on machine j,
T_i tardiness of order i,
C_i complete time of order i,
F_i flow time of order i,
D_i due date of order i,
A_i start time of order i,

SL_i slack time of order i,
S_i sale value of the order i,
M_i material cost of order i,
CCR capacity constraint resource or bottleneck of system,
$TH_{(i)}$ throughput$_{(i)}$, contribution margin order i, it is equal to $S_i - M_i$

As defined above, the TDD is the summation of the value of the orders multiplied by the number of days by which their delivery is late, and IDD is the summation of the dollar value of the WIP multiplied by the time since the WIP entered the plant. A review of the literature revealed that the concepts of ΣTDD_i and ΣIDD_i are similar to those of ΣW_iT_i (total weighted tardiness) and ΣW_iF_i (total weighted flow time) (Rajendran and Ziegler 1997a, b, 1999, Holsenback *et al.* 1999, Volgenant and Teerhuis 1999). Among those dispatching rules, weighted shortest processing time (WSPT) (Bedworth and Bailey 1987) and ATC (Pinedo 2002) perform well in achieving the minimum ΣW_iF_i and ΣW_iT_i, respectively, in a single machine.

Since the dollar amounts in TDD and IDD represent the sale price of orders 'S_i' and the value of material 'M_i' respectively. Further decomposition (Liaw 1999) reveals that TDD and IDD can be expressed by the following equations (it is assumed that k different orders are processed by n machines):

$$\begin{aligned}\min Z_1 &= \sum_{i=1}^{k} \mathrm{TDD}_i = \sum_{i=1}^{k} S_iT_i = \sum_{i=1}^{k} S_i \times \max\,[0, C_i - D_i] \\ &= \sum_{i=1}^{k} \max\left\{0, S_i\left[\sum_{j=1}^{n}(P_{ij} + \mathrm{ID}_{ij}) - D_i\right]\right\} \\ &= \sum_{i=1}^{k} \max\left\{0, S_i\left[\sum_{j=1}^{n}\mathrm{ID}_{ij} - \left(D_i - \sum_{j=1}^{n} P_{ij}\right)\right]\right\} \end{aligned} \quad (1)$$

$$\begin{aligned}\min Z_2 &= \sum_{i=1}^{k} \mathrm{IDD}_i = \sum_{i=1}^{k} M_iF_i = \sum_{i=1}^{k} [M_i \times (C_i - A_i)] \\ &= \sum_{i=1}^{k}\left[M_i \times \left(\sum_{j=1}^{n} P_{ij} + \sum_{j=1}^{n}\mathrm{ID}_{ij} - A_i\right)\right] \\ &= \sum_{i=1}^{k}\left\{M_i \times \left[\sum_{j=1}^{n}(P_{ij} + \mathrm{ID}_{ij}) - A_i\right]\right\} \end{aligned} \quad (2)$$

If S_i and M_i of equations (1) and (2) are treated as being equal to the W_i in ΣW_iT_i and ΣW_iF_i respectively, then the concept of minimization of ΣW_iT_i and ΣW_iF_i can be applied to minimize ΣTDD_i and ΣIDD_i. In this study, for simplicity, TDD is assumed to be as important as IDD. All orders are available when the scheduling begins, i.e. $A_i = 0$ for all i. Therefore, the flow time of each order equals its completion time and equations (1) and (2) can then be combined as follows:

$$\min Z = Z_1 + Z_2 = \sum_{i=1}^{k} \max\left[0, S_i\left(\sum_{j=1}^{n}\mathrm{ID}_{ij} - \mathrm{SL}_i\right)\right] + \sum_{i=1}^{k}\left[M_i \sum_{j=1}^{n}(P_{ij} + \mathrm{ID}_{ij})\right] \quad (3)$$

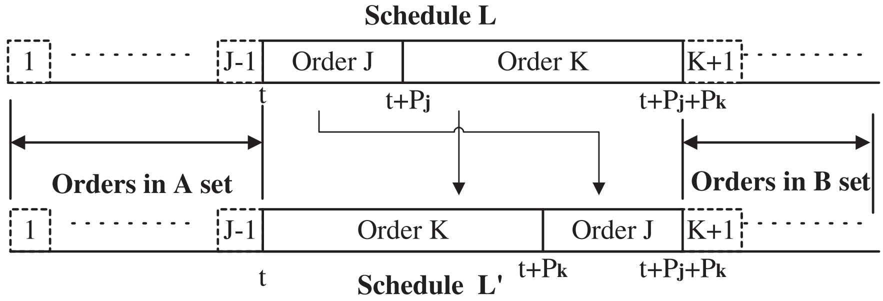

Figure 1. Adjacent pair interchanges method.

where $\mathrm{SL}_i = D_i - \sum_{j=1}^{n} P_{ij}$.

The parameters S_i, M_i, P_{ij} and D_i in equation (3) are as given; therefore, the only way to minimize Z is to control the variable ID_{ij}. However, according to TOC, in the job shop environment, the bottleneck machine determines the throughput of the plant. Scheduling the orders of the bottleneck machine can yield the best performance to meet due dates. It becomes a single machine-scheduling problem. The variable ID_{ij} depends on the dispatching rule used, so only reducing the idle time of resources and orders in the bottleneck can minimize Z. How can this conclusion be confirmed to be correct? The adjacent pair-wise interchange method (figure 1) can be used to prove that reducing the idle time of resources and orders in the bottleneck will minimize Z_1 and Z_2 in a single-machine environment.

2.1. *Minimization of ΣTDD_i*

Minimizing $\Sigma\mathrm{TDD}_i$ for a single machine is actually an NP-hard problem. Although no efficient algorithm is available for minimizing $\Sigma W_i T_i$ with arbitrary weight, such an algorithm does exist when Rule 1 applies (Pinedo 2002).

Rule 1: If two orders J and K satisfy $D_j \le D_k$, $P_j \le P_k$, $S_j \ge S_k$, an optimal sequence exists in which order J precedes order K. (Assume both are tardy and that neither can be completed early, regardless of the order in which they are received.)

Proof: In figure 1, a so-called adjacent pair-wise interchange method is applied to orders J and K. Under the original schedule L, order J begins to be processed at time t, and is followed by order K, whereas under the new schedule L′, order K begins to be processed at time t, and is followed by order J. All other orders in sets A and B remain in their original sequence of processing. Sets A and B are constituted by a series of orders. The orders in sets A and B are started and completed at the same times in both sequences; hence, their $\Sigma\mathrm{TDD}_i$ values are the same. The only difference in the completion times in the two sequences concerns orders J and K; therefore, the calculation of $\Sigma\mathrm{TDD}_i$ is only considered for orders J and K.

Given: $P_j \le P_k,\ S_j \ge S_k \Rightarrow S_k \times P_j \le S_j \times P_k$

Schedule L′ (order K before order J):

$$\begin{aligned}\Sigma\mathrm{TDD}_{(i\,\mathrm{in}\,L')} &= S_k \times \max[0, (t + P_k) - D_k] + S_j \times \max[0, (t + P_j + P_k) - D_j] \\ &= \max[0, S_k \times (t + P_k - D_k)] + \max[0, S_j \times (t + P_j + P_k - D_j)]. \quad (4)\end{aligned}$$

Schedule L (order J before order K):

$$\begin{aligned}\Sigma\text{TDD}_{(i\,\text{in}\,L)} &= S_j \times \max[0, (t + P_j) - D_j] + S_k \times \max[0, (t + P_j + P_k) - D_k] \\ &= \max[0, S_j \times (t + P_j - D_j)] + \max[0, S_k \times (t + P_j + P_k - D_k)]. \quad (5)\end{aligned}$$

Equations (4) and (5) are transformed into equations (6) and (7), respectively, because the two adjacent orders J and K are both tardy and cannot be completed early.

$$\text{Schedule L}': \ \Sigma\text{TDD}_{(i\,\text{in}\,L')} = S_k t + S_k P_k + S_j t + S_j P_j + S_j P_k - S_k D_k - S_j D_j \quad (6)$$

$$\text{Schedule L}: \ \Sigma\text{TDD}_{(i\,\text{in}\,L)} = S_j t + S_j P_j + S_k t + S_k P_j + S_k P_k - S_j D_j - S_k D_k \quad (7)$$

Let $\delta = \Sigma\text{TDD}_{(i\,\text{in}\,L')} - \text{TDD}_{(i\,\text{in}\,L)} = S_j P_k - S_k P_j$.

If schedule L$'$ is optimum, then δ must not exceed zero. Restated, $\delta \leq 0$. then:

$$\delta = \text{TDD}_{(i\,\text{in}\,L')} - \text{TDD}_{(i\,\text{in}\,L)} = S_j P_k - S_k P_j \leq 0 \Rightarrow S_j P_k \leq S_k P_j. \quad (8)$$

Function (8) is violated under the given $S_k P_j \leq S_j P_k$, and so the assumption that schedule L$'$ is optimum is incorrect. The idle times of orders J and K in schedule L and schedule L$'$ are compared. The former, $t + t + P_j$, does not exceed the latter, $t + t + P_k$.

The ATC rule is better for solving most $\Sigma W_i T_i$ problems in single machine. The heuristic rule uses the ranking index below, in which θ is a scaling parameter that can be determined empirically and $\bar{P}$ is the average of the processing times of the remaining jobs:

$$I_i(t) = \frac{W_i}{P_i} \exp\left[-\frac{\max(D_i - A_i - P_i, 0)}{\theta \bar{P}}\right]. \quad (9)$$

In summary, in a single machine environment W_i, P_i and $\exp[-\max(D_i - A_i - P_i, 0)]$ are three ample parameters that impact the scheduling of $\Sigma W_i T_i$.

2.2. *Minimization of ΣIDD_i*

The problem of $\Sigma W_i F_i$ gives rise to one of the better known rules in scheduling theory — the WSPT rule. As the 'M_i/P_i' value of order I becomes greater, order I is processed earlier. For ΣIDD_i, the holding cost per unit processing time thus increases, such that the WSPT rule prevents ΣIDD_i from increasing on the single machine as proven below.

Rule 2: The WSPT rule is optimal for ΣIDD_i $(1\|\Sigma W_i F_i)$. When n orders are scheduled on a single machine, where each order I has a material cost of M_i, the sum of inventory dollar-day is minimized by processing of the orders as follows:

$$\frac{M_1}{P_1} \geq \frac{M_2}{P_2} \geq \cdots \geq \frac{M_n}{P_n}. \quad (10)$$

Proof: Consider two arbitrary sequences, L and L$'$ (figure 1) of the same set of orders. These sequences are identical except in the order of two consecutive orders J and K, which are reversed in L$'$, and $M_j/P_j \geq M_k/P_k$. The orders before J and K constitute set A and those after J and K constitute set B. The orders in sets A and B start and are completed at the same times in both sequences; thus, their ΣIDD_i

are the same. The only difference between the flow times of the two sequences refers to orders J and K. The ΣIDD_i for each sequence is given by:

$$\text{Schedule L } \Sigma IDD_{(i \text{ in } L)} = \Sigma IDD_{\text{set A}} + (t + P_j) \times M_j + (t + P_j + P_k) \times M_k + \Sigma IDD_{\text{set B}}$$

$$\text{Schedule L}' \ \Sigma IDD_{(i \text{ in } L')} = IDD_{\text{set A}} + (t + P_k) \times M_k + (t + P_k + P_j) \times M_j + IDD_{\text{set B}}$$

$$\text{Let } \Sigma IDD_{(i \text{ in } L)} - \Sigma IDD_{(i \text{ in } L')} = (t + P_j)M_j + (t + P_j + P_k)M_k - (t + P_k)M_k - (t + P_k + P_j)M_j = P_jM_k - P_kM_j. \quad (11)$$

Recall $M_j/P_j \geq M_k/P_k$; thus, $P_jM_k - P_kM_j \leq 0$ and $\Sigma IDD_{(i \text{ in } L)} \leq \Sigma IDD_{(i \text{ in } L')}$. Notably, $\Sigma IDD_{(i \text{ in } L)}$ of schedule L was less than $\Sigma IDD_{(i \text{ in } L')}$ of schedule L′ because the M_j/P_j of order J is larger than the M_k/P_k of order K in schedule L. Restated, the idle time multiplied by material cost in schedule L′ exceeds that in schedule L.

2.3. *Establishing the heuristic dispatching PI index*

That ΣTDD_i and ΣIDD_i can be separately minimized was proven above, but how can they be minimized simultaneously? Equation (3) states that the simultaneous minimization of both ΣTDD_i and ΣIDD_i is affected by those given parameters S_i, M_i, SL_i, D_i and P_{ij}. What are their compositions about those parameters to affect Z ($= \Sigma TDD_i + \Sigma IDD_i$) seriously? In WSPT and ATC rules, $M_i/\Sigma P_{ij}$ and SL_i more strongly impact Z by means of heuristic inference. $M_i/\Sigma P_{ij}$ represents the cumulative cost per unit processing time and is directly proportional to Z. Where SL_i represents the tightness of the due date and is inversely proportional to Z. The sequence of orders entering the bottleneck is initially followed by equation (12):

$$\left(\frac{M_i}{\sum_{j=1}^{n} P_{ij}}\right) \Big/ \exp\left[\max\left(D_i - A_i - \sum_{j=1}^{n} P_{ij}, 0\right)\right]. \quad (12)$$

Again, according to TOC, the profit velocity (rate of profit-making) is more important than the unit profit. Profit velocity is defined as the order throughput (TH; marginal contribution) divided by the capacity constrained resource (CCR) time unit. That is TH/CCR. Here, the throughput profit of an order is defined as the sale price minus material cost ($S_i - M_i$). The higher is the TH/CCR, the higher will be the priority. By incorporating the TH/CCR factor into equation (12), the PI index of the heuristic dispatching rule is expressed in equation (13). The priority of an order on the bottleneck machine depends on its PI index. Restated, a higher PI of an order corresponds to its earlier processing on the bottleneck machine. This is called the TDD/IDD dispatching rule.

$$\begin{aligned} PI &= \frac{TH_{(i)}}{P_{iCCR}}\left(\frac{M_i}{\sum_{j=1}^{n} P_{ij}}\right) \Big/ \exp\left[\max\left(D_i - A_i - \sum_{j=1}^{n} P_{ij}, 0\right)\right] \\ &= \frac{(S_i - M_i)}{P_{iCCR}}\left(\frac{M_i}{\sum_{j=1}^{n} P_{ij}}\right) \Big/ \exp\left[\max\left(D_i - A_i - \sum_{j=1}^{n} P_{ij}, 0\right)\right]. \end{aligned} \quad (13)$$

2.4. *Procedure for applying TDD/IDD-based dispatching rule*

The algorithm that determines the priorities of orders is implemented as follows.

Step 1. Identify the bottleneck, if no bottleneck exists (meaning that sufficient capacity is available to process the orders), then go to Step 2; otherwise go to Step 3.

Step 2. Set the ΣTDD_i index to zero and compute the $M_i/\Sigma P_{ij}$ ratio. Rank the orders according to the $M_i/\Sigma P_{ij}$ ratio: a larger $M_i/\Sigma P_{ij}$ ratio corresponds to a higher priority. Compute ΣIDD_i. The algorithm stops.

Step 3. Compute PI. Rank the orders according to the PI; a larger PI corresponds to a higher priority. Create a scheduling Gantt chart and compute the ΣTDD_i and ΣIDD_i. The algorithm stops.

3. Example

A job shop example is presented to demonstrate the feasibility of TDD/IDD dispatching rule. The shop consists of six machines (G, P, Q, R, S, T) and processes five different products (U, V, X, Y, Z). Table 1 summarizes order quantity, routing, price, material cost, due date and other data.

Calculating the loading of each machine reveals that machine R is the bottleneck (3015 > 2400 min). Table 2 illustrates the computed results.

A bottleneck machine is present, so Step 3 is implemented and the PI for each order is computed. Table 3 shows the computed PI, according to which the order processing sequence is Y-Z-U-X-V.

Figures 2 and 3 present the throughput chain and the scheduling Gantt chart, respectively. Table 4 summarizes the idle times of five orders in the job shop, obtained by applying TDD/IDD dispatching rule; and table 5 shows the TDD and IDD values.

The calculation was performed as follows.

Idle time: 22.09 + 17.23 + 11 + 3.25 + 0 + 0 = 53.57
Cycle time: when the start time are zero, the cycle time equal to last order V leave the job shop. It is 77.27 h
Number of tardy 3 (orders U, X and V are tardy)
Mean FT (23.75 + 6.27 + 42.45 + 64.6 + 77.27)/5 = 42.87
Maximum tardiness 12.27 (order V)

(1)	(2)	(3)	(4)	(5)	(6)							(7)
					Process time (min/unit)							
Order	Routing	Order quantity	Unit price ($)	RM cost ($)	G	P	Q	R	S	T	Total (min)	Due day (h)
U	G-P-R-S-T	70	70	35	1	5		15	5	2	28	35
V	G-Q-R-S-T	125	30	10	2		6	8	5	1	22	65
X	G-P-Q-R-S	165	50	15	2	2	4	5	6		19	55
Y	G-P-Q-S-T	95	60	20	3	4	3		3	2	15	39
Z	P-Q-R-T	20	100	35		6	5	7		1	19	10

Table 1. Five orders processed by six various machines in a job shop.

Order	Order quantity	Marginal (throughput)	Profit	Capacity required by each machine process order (min)						
				G	P	Q	R**	S	T	Total
U	70	70 − 35 = 35	70 * 35 = 2450	70 * 1 = 70 (1.17 h)	70 * 5 = 350 (5.8 h)		70 * 15 = 1050 (17.5 h)	70 * 5 = 350 (5.8 h)	70 * 2 = 140 (2.3 h)	32.67 h
V	125	30 − 10 = 20	125 * 20 = 2500	125 * 2 = 250 (4.17 h)		125 * 6 = 750 (12.5 h)	125 * 8 = 1000 (16.67 h)	125 * 5 = 625 (10.4 h)	125 * 1 = 125 (2.18 h)	45.83 h
X	165	50 − 15 = 35	165 * 35 = 5775	165 * 2 = 330 (5.5 h)	165 * 2 = 330 (5.5 h)	165 * 4 = 660 (11 h)	165 * 5 = 825 (13.75 h)	165 * 6 = 990 (16.5 h)		52.25 h
Y	95	60 − 20 = 40	95 * 40 = 3800	95 * 3 = 285 (4.75 h)	95 * 4 = 380 (6.3 h)	95 * 3 = 285 (4.75 h)		95 * 3 = 285 (4.75 h)	95 * 2 = 190 (3.2 h)	23.75 h
Z	20	100 − 35 = 65	20 * 65 = 1300		20 * 6 = 120 (2 h)	20 * 5 = 100 (1.67 h)	20 * 7 = 140 (2.3 h)		20 * 1 = 20 (0.3 h)	6.33 h
Total capacity requirement				935 (15.6 h)	1180 (19.7 h)	1795 (29.9 h)	3015** (50.3 h)	2250 (37.5 h)	475 (7.91 h)	

**Bottleneck machine (normal capacity each week 5 day × 8 h × 60 min = 2400 min).

Table 2. Throughput and machine process time of each order.

(1) Order	(2) Marginal (Throughput)	(3) CCR time	(4) = (2)/(3) TH/CCR	(5) Factor 1 exp[max(slack$_i$, 0)] = exp($D_i - \Sigma P_{ij}$)	(6) Factor 2 $M_i/\Sigma P_{ij}$	(7) PI index = (4) × (6)/(5)
U	70 − 35 = 35	15	2.33	exp 2.33 = 10.28	75	17
V	30 − 10 = 20	8	2.5	exp 19.17* = 2×10^8	27.27**	3×10^{-7}
X	50 − 15 = 35	5	7	exp 2.75 = 15.6	19.15	8.6
Y	60 − 20 = 40	0	8	exp 15.25 = 4.2×10^6	35.35	∞
Z	100 − 35 = 65	7	9.3	exp 3.7 = 39.2	110.58	26.23

For order V, the slack time = ($D_i - \Sigma P_{ij}$) = 65 − 45.83 = 19.17*.
For order V, the $M_i/\Sigma P_{ij}$ = (125 × 10)/(45.83) = 27.27**.
Sequence of orders is 'Y-Z-U-X-V' according to the seventh column (PI index).

Table 3. Orders priorities according to the PI index.

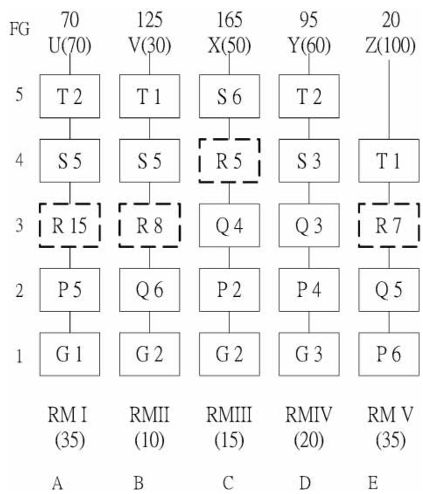

Figure 2. Throughput chain.

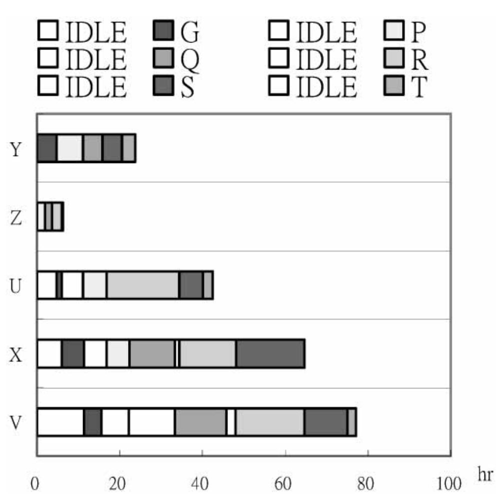

Figure 3. Gantt chart by the TDD/IDD rule.

	Idle	G	Idle	P	Idle	Q	Idle	R	Idle	S	Idle	T
Y	0.00	4.75	0.00	6.30	0.00	4.75	0.00	0.00	0.00	4.75	0.00	3.20
Z	0.00	0.00	0.00	2.00	0.00	1.67	0.00	2.30	0.00	0.00	0.00	0.30
U	4.75	1.17	5.13	5.80	0.00	0.00	0.00	17.50	0.00	5.80	0.00	2.30
X	5.92	5.50	5.43	5.50	0.00	11.00	1.00	13.75	0.00	16.50	0.00	0.00
V	11.42	4.17	6.67	0.00	11.00	12.50	2.25	16.67	0.00	10.40	0.00	2.10
Σ	22.09		17.23		11		3.25		0		0	

Table 4. Idle time about five orders happen under the TDD/IDD dispatching rule.

	(1)	(2)	(3) $=(1)\times(2)$	(4)	(5) $=(1)\times(4)$	(6)	(7)	(8)	(9) $=(3)\times(6)$	(10) $=(5)\times(8)$
	Order quantity	RM cost ($)	Total cost ($)	Price ($)	Sales ($)	Flow time	Due day	Tardiness	IDD	TDD
Y	95	20	1900	60	5700	23.75	39	0	45 125	0
Z	20	35	700	100	2000	6.27	10	0	4389	0
U	70	35	2450	70	4900	42.45	35	7.45	104 002	36 505
X	165	15	2475	50	8250	64.60	55	9.6	159 885	79 200
V	125	10	1250	30	3750	77.27	65	12.27	96 587	46 013

Table 5. Calculating performance indexes through the TDD/IDD dispatching rule.

ΣTardiness $(0 + 0 + 7.45 + 9.6 + 12.27) = 29.32$
ΣTDD $(0 + 0 + 36\,505 + 79\,200 + 46\,013) = 161\,718$
ΣIDD $(45\,125 + 4389 + 104\,002 + 159\,885 + 96\,587) = 409\,988$
$Z = 161\,718 + 409\,988 = 571\,706$

Five dispatching rules — SPT, EDD, total profit, minimum slack and ATC (shifting bottleneck heuristic shown in figure 4) — are compared with the TDD/IDD dispatching rule shown in table 6. Appendix 1 presents the computed results obtained by applying the selected dispatching rules. The following conclusions are drawn. (1) The performance of the TDD/IDD heuristic rule in terms of TDD, IDD, idle time and cycle time are best, as shown in italics in table 6. (2) From the perspective of traditional cost accounting, the orders are prioritized to maximize unit profit, which equals sale price minus material cost. However, from a TOC perspective, the throughput per CCR-minute is the main factor considered in decision-making: a larger value corresponds to earlier processing, as advocated by TOC.

4. Computational experiments

Although the foregoing example demonstrates that the TDD/IDD dispatching rule outperforms the other considered dispatching rules, it is still only an example. Here, experimental simulations of cases are run to make a more robust comparison. For example, in trial 1, five orders were processed in a job shop with four different machines. The measures of performance are the six traditional indices and two global indexes. The simulation experiments assume a job shop configuration. The routings of the orders may differ and are generated randomly. The algorithm is coded in visual basic and implemented on a Pentium II 333-MHz computer.

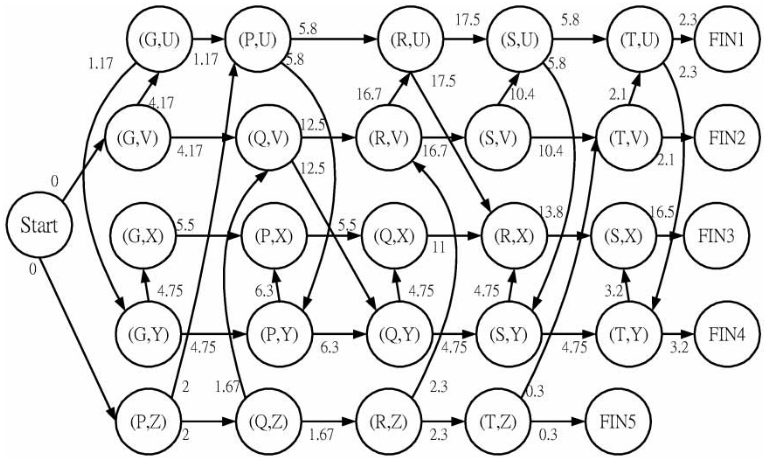

Figure 4. Directed graphs of the shifting bottleneck heuristic.

Rule \ Index	TDD/IDD heuristic	SPT	EDD	Total profit	Minimum slack	ATC
IDLE time	53.57	56.58	54.66	184.6	104.6	94.2
Cycle time	77.27	81.27	78.27	81.82	78.97	81.09
Numbers tardy	3	2	3	4	3	3
Mean FT	42.87	43.45	45.86	68.84	55.39	51.35
T_{max}	12.27	26.27	13.27	71.82	45.02	26.09
ΣTardiness	29.32	33.72	30.62	142.96	75.66	75.62
Global performance						
ΣTDD	161 718	253 233	176 326	505 253	251 097	478 412
ΣIDD	409 988	434 060	433 511	586 222	462 778	529 511
Z	571 707	687 292	609 837	1 091 475	713 875	1 007 922

Table 6. Performance indexes under various dispatching rules.

Table 7 compares the results. The TDD/IDD dispatching rule outperforms the other considered dispatching rules for the job shop in terms of the minimum values of TDD and IDD. In particular, when the number of orders exceeds the number of machines in the job shop very much, the performance of TDD and IDD becomes strongly significant.

5. Conclusions

This study proposes a novel TDD/IDD dispatching rule that can be applied in a job shop environment to maximize the performance of TDD and IDD, as advocated by the TOC. An example demonstrated the feasibility of the TDD/IDD dispatching rule and that its TDD and IDD performance is better than that of five traditional dispatching rules. An experimental simulation then proved that the developed TDD/IDD dispatching rule outperforms the five considered traditional dispatching rules in the given scenarios.

Trial	Number of jobs/orders	Number of machines	Rule	Idle	CT	Number tardy	$\overline{\text{FT}}$	$T_{\max}$	ΣT	ΣTDD	ΣIDD
1	5	4	TDD/IDD	37.30	63.40	3	37.40	15.50	25.30	312 453	405 234
			SPT	40.20	73.40	3	35.40	24.30	31.40	423 445	512 345
			EDD	39.20	76.30	2	38.30	31.30	26.30	545 435	634 643
			T (profits)	120.90	83.40	4	57.40	84.30	124.30	534 634	564 543
			Min (ST)	102.30	74.30	3	56.30	73.20	50.30	43 453	634 453
			ATC	49.30	71.60	2	33.30	25.30	27.40	432 952	534 678
2	5	5	TDD/IDD	38.20	53.43	3	37.29	10.91	24.52	193 523	546 434
			SPT	40.78	62.33	3	35.51	19.63	29.95	346 343	654 975
			EDD	39.89	64.91	2	38.09	25.86	25.41	353 456	615 453
			T (profits)	72.60	71.23	4	55.09	73.03	112.63	1 453 394	654 740
			Min (ST)	56.05	63.13	3	54.11	63.15	46.77	293 342	747 233
			ATC	57.78	50.62	2	35.54	29.42	23.19	234 644	600 471
3	6	4	TDD/IDD	36.83	83.52	4	44.50	16.39	43.42	573 532	601 430
			SPT	39.88	96.22	4	41.96	27.47	53.89	638 645	634 500
			EDD	38.83	99.90	3	45.64	35.38	45.14	565 877	613 433
			T (profits)	36.95	93.92	3	59.90	35.28	44.33	623 753	694 302
			Min (ST)	35.34	97.36	4	68.50	18.74	86.33	733 678	630 647
			ATC	36.02	86.63	3	51.99	19.90	81.35	593 244	654 480

4	7	4	TDD/IDD	100.71	171.18	3	100.98	39.15	68.31	627 345	735 946
			SPT	108.54	198.18	3	95.58	65.61	84.78	734 341	843 092
			EDD	105.84	206.01	3	103.41	84.51	71.01	743 423	834 743
			T (profits)	116.43	225.18	3	154.98	227.61	75.61	703 932	794 952
			Min (ST)	126.21	200.61	2	152.01	197.64	81.81	732 432	804 453
			ATC	97.11	160.32	3	116.91	95.31	127.98	691 342	852 504
5	7	5	TDD/IDD	115.63	196.54	3	115.94	44.95	78.43	394 644	795 430
			SPT	104.62	207.54	2	109.74	75.33	97.34	542 432	834 436
			EDD	121.52	236.53	3	118.73	67.03	81.53	493 345	825 044
			T (profits)	124.79	258.54	4	177.94	61.33	85.33	543 345	836 008
			Min (ST)	137.13	230.33	5	174.53	76.92	65.93	522 233	864 053
			ATC	128.03	202.96	3	134.23	59.43	86.94	523 243	854 984
6	10	4	TDD/IDD	130.55	221.90	3	130.90	50.75	88.55	704 564	810 004
			SPT	140.70	256.90	4	123.90	85.05	89.90	944 345	939 345
			EDD	137.20	267.05	3	134.05	59.55	92.05	903 324	1 164 033
			T (profits)	123.15	291.90	4	115.90	65.05	95.05	873 453	1 043 049
			Min (ST)	113.05	200.05	3	134.05	56.20	86.05	856 045	1 146 093
			ATC	147.55	245.60	4	141.55	61.55	89.90	869 335	993 093

Table 7. Indexes measuring six dispatching rules under various jobs and machines.

Acknowledgment

The authors thank the referees for valuable comments and suggestions.

Appendix

	$Cost_{raw}$ (\$)	Sales (\$)	Flow time	Due day	Tardiness	IDD	TDD
Z	700	2000	6.27	10	0	4389	0
Y	1900	5700	23.75	39	0	45 125	0
U	2450	4900	42.45	35	7.45	104 003	36 505
V	1250	3750	63.52	65	0	79 400	0
X	2475	8250	81.27	55	26.27	201 143	216 728

Order sequence (Z-Y-U-V-X).
$Z = \Sigma TDD + \Sigma IDD = 253\,233 + 434\,060 = 687\,292.$

Table A1. Indexes of performance by the SPT dispatching rule.

	$Cost_{raw}$ (\$)	Sales (\$)	Flow time	Due day	Tardiness	IDD	TDD
Z	700	2000	6.27	10	0	4389	0
U	2450	4900	33.4	35	0	81 830	0
Y	1900	5700	45.5	39	6.5	86 450	37 050
X	2475	8250	65.86	55	10.85	163 004	89 513
V	1250	3750	78.27	65	13.27	97 835	49 763

Order sequence (Z-U-Y-X-V).
$Z = \Sigma TDD + \Sigma IDD = 176\,326 + 433\,511 = 609\,837.$

Table A2. Indexes of performance by the EDD dispatching rule.

	$Cost_{raw}$ (\$)	Sales (\$)	Flow time	Due day	Tardiness	IDD	TDD
X	2475	8250	52.25	55	0	129 319	0
Y	1900	5700	60.2	39	21.2	114 380	120 840
V	1250	3750	68.42	65	3.42	85 525	12 825
U	2450	4900	81.52	35	46.52	199 724	227 948
Z	700	2000	81.82	10	71.82	57 274	143 640

Order sequence (X-Y-V-U-Z).
$Z = \Sigma TDD + \Sigma IDD = 505\,253 + 586\,222 = 1\,091\,475.$

Table A3. Indexes of performance by the total profit rule.

	$Cost_{raw}$ (\$)	Sales (\$)	Flow time	Due day	Tardiness	IDD	TDD
U	2450	4900	32.57	35	0	79 797	0
X	2475	8250	54.72	55	0	135 432	0
Z	700	2000	55.02	10	45.02	38 514	90 040
Y	1900	5700	62.67	39	23.67	119 073	134 919
V	1250	3750	78.97	65	6.97	89 963	26 138

Order sequence (U-X-Z-Y-V).
$Z = \Sigma TDD + \Sigma IDD = 251\,097 + 463\,778 = 713\,875.$

Table A4. Indexes of performance by the minimum slack rule.

	$Cost_{raw}$ (\$)	Sales (\$)	Flow time	Due day	Tardiness	IDD	TDD
Z	700	2000	6.27	10	0	4389	0
V	1250	3750	45.84	65	0	57 300	0
U	2450	4900	58.94	35	23.94	144 403	117 306
Y	1900	5700	64.59	39	25.59	122 721	145 863
X	2475	8250	81.09	55	26.09	200 698	215 243

Order sequence (Z-V-U-Y-X).
$Z = \Sigma\text{TDD} + \Sigma\text{IDD} = 478\,412 + 529\,511 = 1\,007\,922$.

Table A5. Index of performance by the ATC dispatching rule.

References

Bedworth, D. D. and Bailey, J., 1987, *Integrated Production Control Systems. Management, Analysis, Design*, 2nd edn (New York: Wiley), pp. 255–260.

Goldratt, E. M., 2000, *Essays On the Theory of Constraints* (North River, Great Barrington, MA).

Holsenback, J. E., Russell, R. M., Markland, R. E. and Philipoom, P. R., 1999, An improved heuristic for the single-machine, weighted-tardiness problem. *Omega, Int. J. Mgmt*, **27**, 485–495.

Liaw, C. F., 1999, A branch-and-bound algorithm for the single machine earliness and tardiness-scheduling problem. *Comput. Oper. Res.*, **26**, 679–693.

Philipoom, P. R., 2000, The choice of dispatching rules in a shop using internally set due-dates with quoted lead time and tardiness costs. *Int. J. Prod. Res.*, **38**, 1641–1655.

Pinedo, M., 2002, *Scheduling Theory, Algorithm, and System*, 2nd edn (Upper Saddle River: Prentice-Hall), pp. 53–54, 335–342.

Rajendran, C. and Ziegler, H., 1997a, An efficient heuristic for scheduling in a flow shop to minimize total weighted flow time of jobs. *Eur. J. Oper. Res.*, **103**, 129–138.

Rajendran, C. and Ziegler, H., 1997b, A heuristic for scheduling to minimize the sum of weighted flow time of jobs in a flow shop with sequence-dependent setup times of jobs. *Comput. Ind. Eng.*, **33**, 281–284.

Rajendran, C. and Ziegler, H., 1999, Heuristics for scheduling in flowshops and flowline-based manufacturing cells to minimize the sum of weighted flow time and weighted tardiness of jobs. *Comput. Ind. Eng.*, **37**, 671–690.

Tsai, C. H., Chang, G. T. and Li, R. K., 1997, Integrating order release control with due-date assignment rules. *Int. J. Prod. Res.*, **35**, 3379–3392.

Tseng, T. Y., Ho, T. F. and Li, R. K., 1999, Mixing macro and micro flow time estimation model: wafer fab example. *Int. J. Prod. Res.*, **37**, 2447–2461.

Volgenant, A. and Teerhuis, E., 1999, Improved heuristics for the *n*-job single-machine weighted tardiness problem. *Comput. Oper. Res.*, **26**, 35–44.