Expert Systems with Applications 37 (2010) 3747–3754

Contents lists available at ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Research on enhancement of TOC Simplified Drum-Buffer-Rope system using novel generic procedures

Jun-Huei Lee [a], Jia-Ging Chang [b], Chih-Hung Tsai [c,*], Rong-Kwei Li [a]

[a] Department of Industrial Engineering and Management, National Chiao-Tung University, Hsinchu, Taiwan
[b] WTMicroelectronics, Taipei, Taiwan
[c] Department of Information Management, Yuanpei University, No. 306, Yuanpei Street, Hsin-Chu, Taiwan

ARTICLE INFO

Keywords:
Theory of Constraints
Simplified Drum-Buffer-Rope
Due-date Performance

ABSTRACT

The Theory of Constraints (TOC) Simplified Drum-Buffer-Rope (SDBR) system works effectively in typical job shop environments. However, according to the experiences implementing SDBR system in local companies, such environments have the following characteristics, which may block SDBR implementation: (1) capacity constraint resources (CCR) is not always located in the middle of the routing as assumed in SDBR. The CCR can be located in either the front or back end of routing; (2) multiple or interactive CCRs can exist rather than the assumption of just one CCR; (3) order insertion, including urgent orders, change in due-date (especially bringing it forward) is common. Two ways may be applied to overcome these characteristics: (1) move back to a traditional DBR; (2) address problems through the buffer management process. However, both these ways have limitations and complexities. This paper presents an alternative method enhancing SDBR system performance. With the enhancement, we expect that SDBR can be adopted by more companies, especially those that have the same characteristics.

1. Introduction

Winning businesses in make-to-order (MTO) environments repeatedly turn out to be those with strong ability to fulfill higher levels of reliable due-date performance (DDP). Therefore, numerous studies have attempted to improve DDP. For example Lu and Kumar (1991), Katcher, Arakawa, and Strosnider (1993), Grabot and Geneste (1994), Lin, Wang, and Yen (2001), Lin, Chiu, and Tsai (2008), and Dabbas and Fowler (2003) focused on determining right dispatching rules (working priorities) for different production environments. Graves and Milne (1997), Tsai, Chang, and Li (1997), Breithaupt, Land, and Nyhuis (2002), Nandi and Rogers (2003), Chung and Lai (2006), and Chung, Pearn, Lee, and Ke (2003) concentrated on studying rules for controlling order release, and Lozinski and Glassey (1988), Glassey and Petrakian (1989), Rippenhagen and Krishnaswamy (1998), Chiang, Kuo, and Meerkov (2000), Roser, Nakano, and Tanaka (2002), and Gorinsky and Jechlitschek (2007) dealt with bottleneck starvation issues. These studies demonstrated that DDP can be improved through careful management of order release, working priorities and bottleneck machines utilization. Despite these academic works, businesses have also employed numerous approaches (developed by industrial practitioners) including, but not limited to, JIT, advanced production scheduling system (APS), and Theory of Constraints (TOC) (Watson, Blackstone, & Gardiner, 2007). These efforts have all focused on improving DDP. Among these approaches, The TOC Drum-Buffer-Rope (DBR) system developed by Goldratt (1990, 1992, 1996), and Goldratt and Fox (1986) is one of the best known methods of improving DDP.

Previously, hundreds of DBR successful stories have been reported and these reports have claimed that highly reliable DDP can be rapidly achieved (Mabin & Balderstone, 2000; Umble, Umble, & Murakami, 2006). Traditional DBR uses a three-buffer system to protect both the due-dates and detailed finite capacity schedule of the capacity constraint resources (CCR). This approach offers far more protection than merely keeping the CCR from starvation as a result of delay on the non-constraint resources. DBR assumes an internal CCR is active but in reality this is not always the case. In most cases, a company's constraint is in the market which means even the CCR possesses sufficient protective capacity. Therefore, Schragenheim and Dettmer (2000), and Schragenheim, Weisenstern, and Schragenheim (2006) proposed a Simplified Drum-Buffer-Rope (SDBR) method.

The core of SDBR is planned load concept and order due-date setting method. The planned load is the accumulation of the load derived on the CCR (or a bottleneck machine) for all the firm orders requiring delivery within a certain time horizon. For example, suppose three orders are scheduled for delivery within the standard quoted lead-time (QLT) of 12 days. Order #1 requires 1 day of work

* Corresponding author. Tel.: +886 3 5927700x2953; fax: +886 3 5926848.
E-mail address: ietch@thit.edu.tw (C.-H. Tsai).

0957-4174/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2009.11.049

on the CCR, Order #2 requires 2 days and Order #3 requires 1 day. The planned load is then simply the total of 1 + 2 + 1 = 4 days (Fig. 1a). Naturally, the planned should always be less than the QLT offered to the market. The time difference between the QLT and the planned load represents the current protected capacity on the CCR. A minimum time difference is required between the planned load and the QLT to ensure that the last firm order has sufficient time from CCR until completion.

The front of the planned load indicates when, on average, the CCR will be able to work on the new order. Since just one production buffer is assumed, SDBR releases the materials half the production buffer earlier than the CCR is scheduled begin working on it. It is not a critical assumption since timing on the CCR is flexible. The SDBR also assumes that within half of the production buffer enough orders arrive at the CCR to prevent unnecessary starvation. The question thus arises of when orders should be promised. It seems obvious that a safe delivery time can be calculated by adding half the production buffer to the time new orders can work on CCR. Fig. 1b shows the planning logic.

Notably, if an order can reliably be delivered at the date we can committee, the next question becomes deciding whether to promise delivery on that date. It is important not to offer a very early delivery time, namely a time earlier than the QLT, even if such a delivery time may be safe. When the future objective is to offer rapid response (so we can get better price), it is better not to spoil future customers by providing excessively early delivery. Therefore, if the safe delivery time is shorter than the QLT the buffer time should be extended until the QLT and today + QLT should be taken as the promised delivery time. In the SDBR due-date setting method, any order for which the buffer time is extended is called that the order has slack time (Fig. 1c). The idea here is to release the order sufficiently early to ensure the CCR is not starved, in the hope that more demand will appear in the short term. If the material release is delayed to provide the regular production buffer length (sufficient to ensure on time completion of the order), the possible negative branch is a waste of the CCR capacity. This occurs when a low demand period is followed by a very high demand period. If the capacity of the CCR is properly used during the low demand period, then more can be done during the high period without threatening safe delivery within an acceptable lead-time.

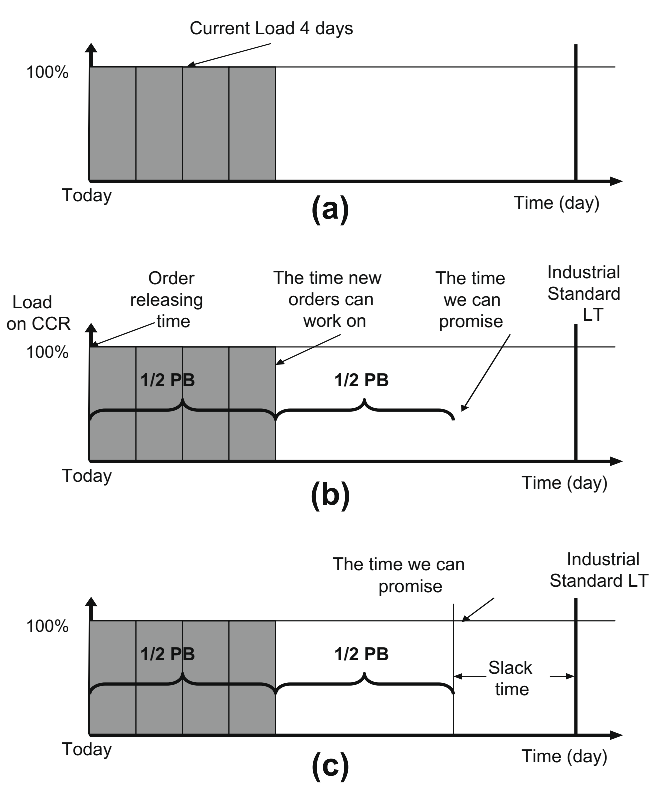

Fig. 1. (a) Four days planned load. (b) Order due date and releasing date setting. (c) Order slack time.

Now take Fig. 1a as an example. In this example the QLT is 12 days and assume the production buffer time of orders is 8 days. Initially three orders (#1, #2 and #3) are already on the floor. The accumulation of the load on the CCR for these three WIPs is 4 days (Fig. 1a). The question is when the reliable delivery dates can be offered for orders on day one. Order #2 is taken as an example. The planned load is 4 days from now and work on the new order can begin on CCR on day 5, meaning the reliable due-date is 5 plus half of the production buffer (plus 4 days). Therefore, the safe due-date is 9 days from the current date which means the order can be safely delivered on day 9. In this case sticks to a QLT of 12 days from now meaning delivery is promised for day 13 (current day 1 + 12 = 13). The order is released to the floor according to the time the new order can work on CCR minus half the production buffer (5 days minus 4 days equal to 1 day), meaning release the order at day one. Fig. 2a illustrates the due-date and release date setting for order #2. Since order #2 requires 2 days on CCR to complete, the planned load for the firmed orders now become 6 days (Fig. 2b). Continuing applies the same approach, it is possible to rapidly determine the safe due-dates and release dates for orders.

As we know, the main difference between SDBR and DBR is that SDBR does not need to determine the precise sequence of the CCR in advance. The actual sequence should be determined based on the order buffer status. Eq. (1) presents the buffer status formula.

$$\text{Buffer status (BS)} = \frac{(\text{production buffer (PB)} - \text{remain days to due-date})}{\text{production buffer(PB)}} \tag{1}$$

Buffer management sets priorities (three color code system) purely according to the degree to which it was consumed (buffer status). Fig. 3 presents an illustrative example. Although SDBR advocates buffer management as the only priority system suitable for use on shop floors, it also leaves flexibility to the shop floor foreman. For example, Fig. 3 shows that the two yellow colored orders (WO2 and WO3) should be prioritized over the green order (WO1). However, SDBR leaves the operator to determine the actual sequence of the two yellow color orders if setup time can be saved through adjusting their sequence. It is particularly true in complex operational environments where the sequence significantly influences quality, setup and process times.

Several success stories indicate that SDBR is as effective as DBR in general job shop environments (Lilly, 2004). Goldratt (2006) also developed a strategic and tactics tree to provide the sequence of implementing. Chang (2008) employed the simulator developed by Goldratt (1996) demonstrated that SDBR is as effective as DBR. Furthermore, Lee, Hwang, Wang, and Li (2009) conducted an experimental study to examine why it is so difficult to achieve high DDP. Thirty five teams participated in the experiment (involving a total of 245 people). Experimental results support the notion that in most cases, the method of managing production planning and execution is the root cause of poor DDP, including the following phenomenon:

- Over-promising, meaning setting order due-dates that fail to consider the planned load of CCR.
- No choking of order release, meaning too many orders occupy the shop floor due to excessively early release, masking priorities, promoting local optima behavior, and thus prolonging lead-time and significantly disrupts DDP.

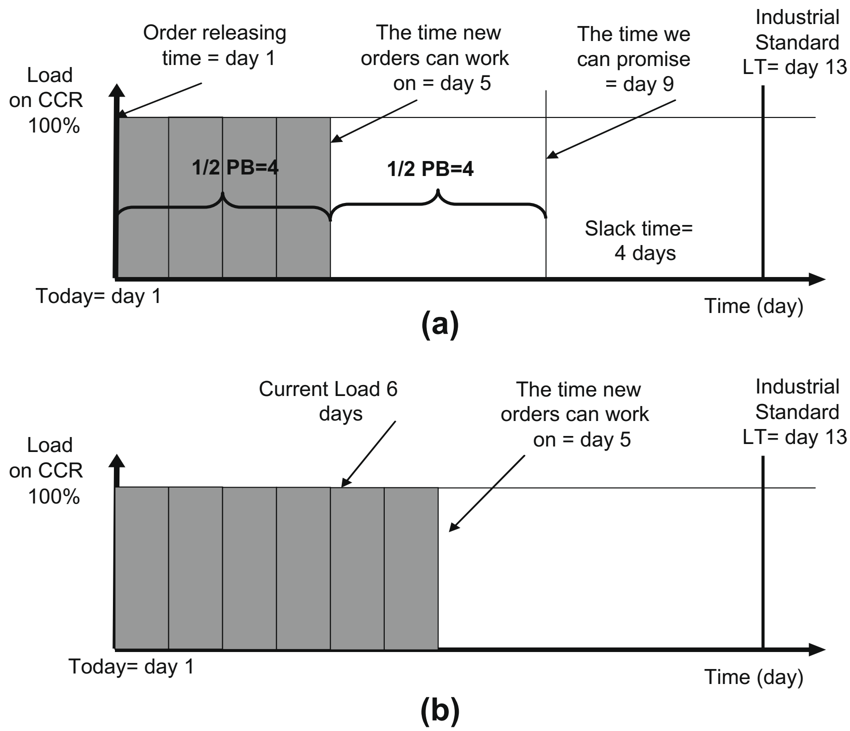

Fig. 2. (a) Due date setting of order #2. (b) Planed load update of order #2.

Today is day five

Work Order	WO1	WO2	WO3
Production Buffer Time	20 days	10 days	5 days
Order Releasing Date	At day 1	At day 1	At day 3
Committed Due Date	At day 20	At day 10	At day 7
Remain Days to Due Date	15 days	5 days	3 days
Buffer Status	(20-15)/20 25% Green	(10-5)/10 50% Yellow	(5-3)/5 40% Yellow

Fig. 3. An example of buffer status.

- Failure to manage priorities, meaning hectic priorities create chaos on the floor, and the lack of a priority system can lead to late orders.

Experimental results prove that by adopting SDBR planned load concept, order due-date setting method, choking of order release and buffer management to resolve the above phenomenon, high DDP can be achieved.

As its name implies, SDBR is easily implemented and significant results can be achieved within a short period (Lilly, 2004). However, according to the experiences implementing SDBRs at local companies, such companies have the following characteristics, which may block SDBR implementation:

- CCR is not always located in the middle of the routing, as assumed in SDBR. That is, CCR can be located in the front or back end of a routing. By ignoring this characteristic, order due-date promised may be overly optimistic or order may be released too late.
- Multiple or interactive CCRs can exist. Ignoring this characteristic can result in too many orders in the red zone or excessive delay.
- Order insertion, including that for urgent orders, changes due-dates (especially bringing them forward). Ignoring this characteristic is unrealistic. Schragenheim (2006) proposed a capacity reservation concept for handling this characteristic. However, Schragenheim assumed that offering urgent orders to clients is a promise rather than an option. A promise means a supplier cannot refuse to deliver an order regardless of capacity loading, meaning that reserve capacity is necessary; otherwise, promises would be impossible to meet. Again, in certain environments, urgent orders do not require promises, but rather an option, in which case an urgent order is accepted or refused depending on capacity load.

Two ways can be applied to overcome these characteristics: (1) move back to traditional DBR; (2) address them through the buffer management process. Returning to traditional DBR can resolve problems associated with characteristics 1 and 3 (SDBR does not address characteristic 3) but not 2. The reasons that returning to traditional DBR cannot resolve characteristic 2 are as follows:

- The complexity of planning CCR schedules for interactive CCRs becomes exponential compared with that for only one CCR. Notably, complexity is high not only when planning CCR schedules but also when determining the best product mixs (Barnard, 2006).
- Most current DBR softwares cannot handle CCR schedule that is complicated by dependent setups and other complexities such as interactive constraints (Schragenheim, 2006). This means

even when returning to traditional DBR, a heuristic algorithm for planning CCR schedule is also needed. Goldratt (2006) also pointed out that only for environments dominated by heavily dependent set-up matrixes, traditional DBR should be considered; otherwise, SDBR should be used.

The functions of buffer management are: (1) prioritizing orders and expediting orders in the red zone; (2) recording why an order is in the red zone of its buffer (recording what the order is waiting for), analyzing the frequency of orders waiting for the same reason (BM analysis) and launching improvement initiatives; (3) adjusting the production buffer accordingly. Buffer management is a proactive execution system and has two assumptions:

- Planning (order due-date determination) has been done properly. Since characteristics 1–3 are associated with more planning issues than execution issues. The planning issues without being solved by planning methods adds additional variability to a system; addressing them through the buffer management process will create a high percentage of orders in the red zone such that expediting orders becomes the norm or causes order delays due to insufficient time to expedite orders. The empirical study conducted by Lee et al. (2009) indicates that if setting order due-dates fails to consider the planned load of the CCR, even buffer management is active, DDP is still poor.
- The production buffer time is sufficiently large to absorb variability; however, based on experiences of implementing SDBR, this assumption is not always valid because of competition (particularly in the high-tech sectors).

Moving from an SDBR to a traditional DBR and addressing these problems using the buffer management process have their own limitations and complexities. This paper presents a novel method, an enhanced SDBR (an alternative), which can be adopted by more companies, especially those who have the same characteristics as those mentioned above.

2. Novel generic enhancement procedures

2.1. Location of CCR

SDBR assumes that CCR is located in the middle of the routing, but in reality this is not always true. CCR can be located at either the front or back end of the routing. If CCR is located in the front end, using half the production buffer to dictate the due-date and release time, the promises regarding the order due-date may be over optimistic (Fig. 4a).

However, if CCR is located in the back end of the routing, if half the production buffer is still used to determine the due-date and release time, the order will be released too late. In this situation, an excessively low planned load may cause the CCR to be idle (but no harm to order DDP) and potentially to lost sales opportunities (Fig. 4b). If the touch time of the plant is assumed to be a very small fraction (<10%) of the production lead time, there is no need to worry about the location of CCR, and the available buffer time is sufficient to handle the issues mentioned above. However, in certain situations the assumption may be invalid, overpromised may exists, which is one of the major causes of poor DDP (Lee et al., 2009). Consequently, using first available slot time on CCR plus/minus half the production buffer as a yardstick for setting the due-date and the release time must be revised. The revisions are:

- CCR location is in the front end of the routing:
  Order due-date = first available slot time on CCR plus $(1-\alpha)$PB.
  Order release date = first available slot time on CCR minus αPB.
- CCR location is in the back end of the routing:
  Order due-date = first available slot time on CCR plus $(1-\beta)$PB.
  Order release date = first available slot time on CCR minus βPB.

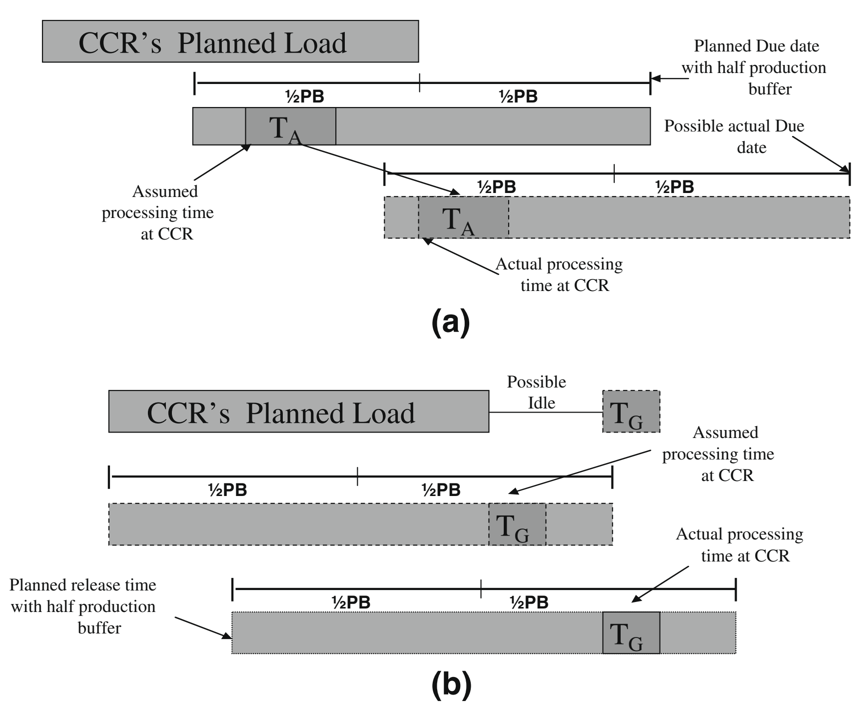

Fig. 4. (a) CCR location in front end of the routing. (b) CCR location in back end of the routing.

The value of α should be below 0.5 and β should be exceed 0.5. Fig. 5 shows the result. The revision is designed to ensure that an order is not released too early (in the case of the order whose CCR location is at the front end) or too late (in the case of the order whose CCR location is at the back end), allowing the order to be processed at CCR at the expected time.

The above revision is sufficient except in an environment that contains a mix of CCR locations, in which case the order buffer status needs to be adjusted, because the actual sequence of orders at CCR is determined by their buffer status. Orders with CCR location at the front end of the routing will be released late (compared with using half the production buffer to determine the release time). Following release the order will soon complete the operations before CCR and then waits in front of the CCR for processing. If most orders waiting in front of the CCR are those whose CCR location in the middle of the routing, then according to the original formula for calculating buffer status, their buffer status is higher and the wait for the new order is delayed until it is assigned a higher priority. Waiting too long in front of the CCR prolongs the expected processing time at CCR of the coming order, this will erode the buffer time and create a risk in meeting the promised due-date (Fig. 6a). The same problem exists for orders whose CCR location is at the back end of the routing, with such orders being released early (compared with using half the production buffer to determine the release time). Following order release, time is required to complete the operations before CCR then waits in front of the CCR for processing. If most orders waiting in front of the CCR are CCR locations in the middle of the routing, then according to the buffer status formula, their buffer status will be lower and newly arrived orders are prioritized for immediate processing. This prioritization accelerates the new order being worked at CCR early and may risk DDP of other orders (Fig. 6b). Delay or acceleration of orders being processed at CCR will lead to CCR work being wrongly prioritized. Moreover, incorrect prioritization of the CCR work will degrade the method for setting the SDBR due-date.

To resolve the above problem, it is necessary to adjust the buffer status for orders with CCR locations at either the front or back ends of the routing. The idea is for an order (with CCR location at either the front or back end of the routing) when it is in the shop floor, its buffer status is adjusted to as if its CCR is located in the middle of the routing. Fig. 7a shows the order whose CCR location is at the front end of the routing. For example, on times X and Y the buffer status should be adjusted as:

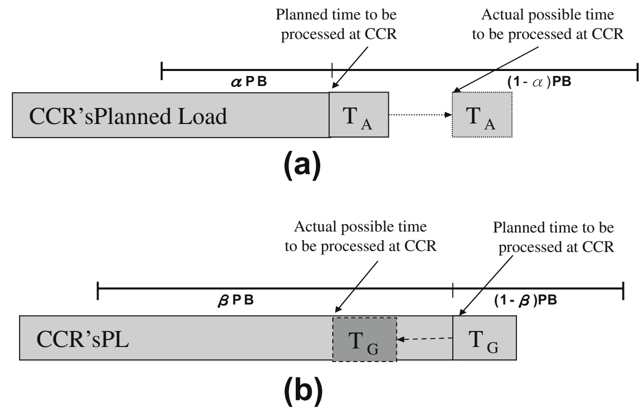

Fig. 6. (a) Delay the order's process time at CCR. (b) Escalate the order's process time at CCR.

$$BS_x = \frac{\frac{x}{\alpha PB}\times\frac{1}{2}PB}{PB} = \frac{\frac{x}{2\alpha}}{PB} = \frac{x}{2\alpha PB} \tag{2}$$

$$BS_y = \frac{\frac{(y-\alpha PB)}{(1-\alpha)PB}\times\frac{1}{2}PB}{PB}+0.5 = \frac{\frac{y-\alpha PB}{2(1-\alpha)}}{PB}+0.5 = \frac{y-\alpha PB}{2(1-\alpha)PB}+0.5 \tag{3}$$

The same adjustment should be made for the order whose CCR location is at the back end of the routing (Fig. 7b). For example, when the time is at X or Y the buffer status should be adjusted as:

$$BS_x = \frac{\frac{x}{\beta PB}\times\frac{1}{2}PB}{PB} = \frac{\frac{x}{2\beta}}{PB} = \frac{x}{2\beta PB} \tag{4}$$

$$BS_y = \frac{\frac{(y-\beta PB)}{(1-\beta)PB}\times\frac{1}{2}PB}{PB}+0.5 = \frac{\frac{y-\beta PB}{2(1-\beta)}}{PB}+0.5 = \frac{y-\beta PB}{2(1-\beta)PB}+0.5 \tag{5}$$

Comparing these equations, regardless of the CCR location, the adjusted buffer status is the function of α or β. Therefore, if θ is

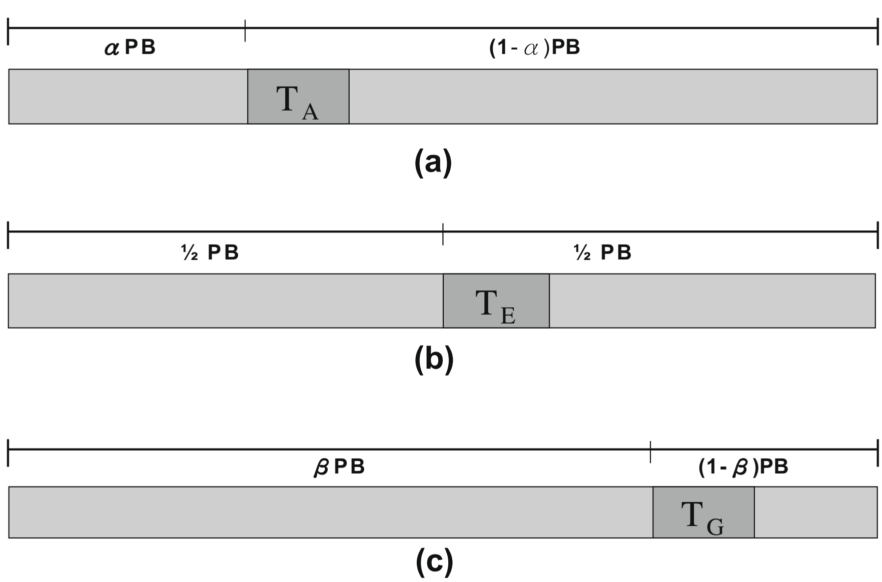

Fig. 5. (a) CCR location in the front end of routing. (b) CCR location in the middle of routing. (c) CCR location in the back end of routing.

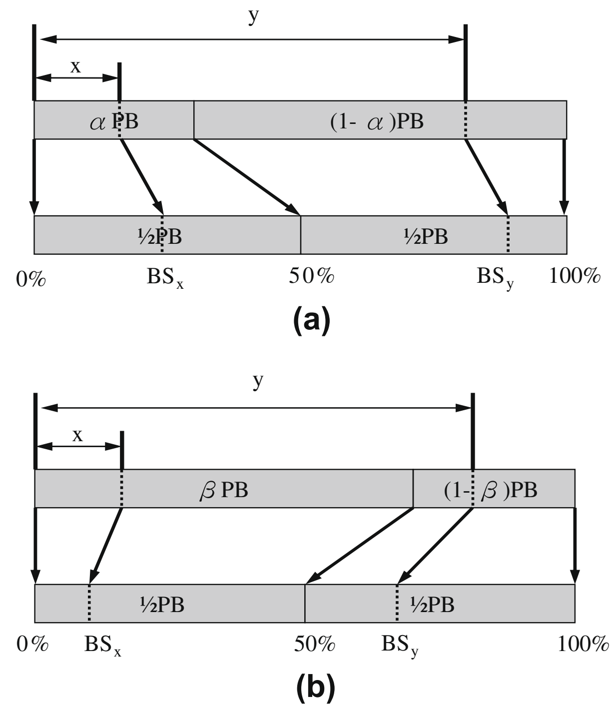

Fig. 7. (a) CCR location in the front end of routing. (b) CCR location in back end of routing.

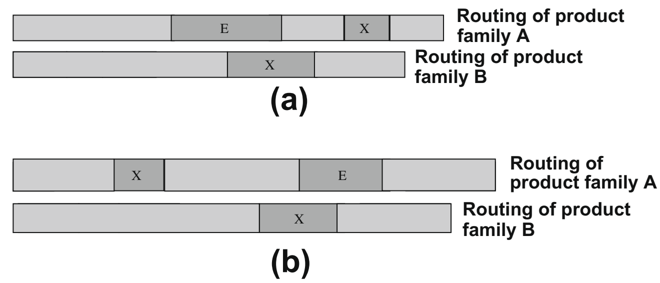

Fig. 8. (a) CCR E is in front of second CCR X. (b) Second CCR X is in front of CCR E.

substituted for α and β, Eqs. (2)–(5) can be combined into Eq. (6) as follows:

$$BS_A = \begin{cases} \frac{x}{2\theta PB}, & 0 \le x \le \theta PB \\ 0.5 + \frac{x-\theta PB}{2(1-\theta)PB}, & \theta PB < x \le PB \quad (0 < \theta < 1) \end{cases} \tag{6}$$

2.2. Multiple CCRs or interactive CCRs

Two types of multiple CCRs exist. In the first type, different product families have their own CCR, and these various CCRs are independent. The order due-date and release date can be based on its own CCR planned load. Unfortunately, this type is rare. In the second type, CCRs are interactive (or CCR shifting). For example, for two product families, A and B, machine E is the CCR of product family A, while machine X is that of product family B, but is also used by product family A, machine X serves as their second CCR for product family A. In this type, orders of product family B can meet their set due-dates based on the planned load of CCR X. However for product family A, due to change of product mixs, CCR may shift. Two cases thus may exist, either CCR E is in front of the second CCR X or the second CCR X is in front of CCR E.

Case 1: The CCR E is in front of second CCR X (Fig. 8a)If the order due-date is set according to the planned load of CCR E, the latest time for CCR E to process the order is approximately at planned time. However, when the order waits for the second CCR X to be processed, its buffer status already exceeds 50% compared with product family B, meaning product family A has higher priority. In this situation both product families can meet their set due-dates. However, if the product mixes changes and more product family B comes in, causing the planned load of machine X to be higher than that of machine E. For product family A, in this situation, CCR shifts from E to X, if their due-date is still set based on the planned load of machine E, product family A is not harmed because it has higher priority while waiting for machine X to complete processing. However, for product family B due-dates may be risked (creating processing delays at machine X). In this situation the due-date is set based on the planned load of machine X rather machine E.

Case 2: The second CCR X is in front of the CCR E (Fig. 8b)The planned load of machine X affects the timing of the processing of product family A at CCR E and causes CCR E to be idle. Because when the product mix changes, the planned load of machine X approaches or exceeds that of machine E. In this situation, the wait faced by product family A for processing by machine X reduces the priority and leaves CCR E idle. The idle time of CCR E increases if machine X becomes CCR. In this situation the due-date is set according to the planned load of machine X. However, the due-date of product family A still suffer if more orders arrive for product family A, even if the due-date is set according to the planned load of machine X. Interactive CCR is a complex situation and while the above method can resolve the problem, the radical solution is to control the load of the second CCR and expand the capacity of the second CCR in time.

2.3. Orders insertion

When urgent order is an option not a promise, it is feasible to employ capacity reservation (Schragenheim, 2006) to handle urgent orders. Doing so is costly, so most companies do not use this approach. Challenge the method used by SDBR to handle urgent orders or change the order due-date (bringing it forward) is raised. The industry terms this challenge "order insertion", and since the issue is extremely prevalent, the ability of SDBR to deal with it is important. Since we know that with the SDBR planned load due-date setting method, order due-date commitments are given according to the first available slot time on the CCR plus half the production buffer. If the committed due-date is earlier than that expected by the customer, SDBR suggests not giving (for free) commitments shorter than standard lead time. This will create a slack time for the committed order. With the slack time concept it becomes easy to determine whether urgent orders can be accepted or committed orders can be brought forward.

Solve the problem of order insertion, the remaining slack time for each committed order can be checked (the slack time is updated with each advance in the time). If the remaining slack time for any committed order is zero, any order insertion will possibly hurt the DDP of the committed orders. For example, take an environment with customer accepted quote lead time 50 h, production buffer 30 h, CCR loading (for a process order) 5 h, and current planned load of CCR 25 h. Suppose three orders (# 1, 2 and 3) arrive, according to the planned load, the three orders and their due-dates can be promised at 50th hours (Fig. 9a). WO#1 has 10 h slack time, WO#2 5 h, but WO#3 0 h. In this case, any order insertion will possibly hurt the DDP of the committed orders. Negotiation should be initiated or the order will be lost.

However, if the remaining slack time for an order exceeds zero which means it might be possible to insert the order without damaging the DDP of committed orders. Take the above example again but with current planned load of CCR 20 h. Fig. 9b shows the due-date and slack time for each order. Every order has slack time, order insertion is possible. Supposing a new order (order #4) arrives and delivery is requested at 40 h (10 h faster than the customer accepted QLT of 50 h, this 10 h is termed the time of ahead schedule), then the question is whether delivery of the order can be promised at 40 h without hurting the due-dates of the firmed orders. Three rules should be followed:

Rule 1: To check whether an order (for example order #1) whose remaining slack time exceeds or equals the time of ahead schedule of the insertion order. If the answer is no, the order cannot be accepted according to its required due-date, meaning that negotiation should be initiated or the order will be lost. However, if the answer is yes, the new order can be inserted into the slot of order #1.

Rule 2: To check whether the CCR loading of the new urgent order must not significantly exceeds that of the last committed order.

Rule 3: To check the derived release date of the new urgent order should not be before today.

Take the example shown in Fig. 9b again, new urgent order #4 is 10 h ahead of schedule, searching the three committed orders reveals that all three have slack time, and the slack time of orders #1 and 3 is greater than or equals 10 h, creating potential for inserting the order. Further checking the CCR loading of orders #4 and #3 (the last committed orders) reveals that they are identical. It then becomes possible to ensure that the order is inserted into the CCR slow of orders #1 or #2 without damaging the due-date performance of the committed orders. The third rule is also met; its release date is at day 10. Fig. 9c illustrates the result after order insertion.

3. Conclusions

This study examines three environments that must be handled with appropriate enhancement of SDBR solutions. Such environments include those with the following characteristics: (1) CCR is not always located in the middle of the routing as assumed in SDBR. CCR can be located in either the front or back ends of routing; (2) multiple or interactive CCRs can exist rather than the assumption of just one CCR; (3) order insertion, including urgent orders, change in due-date (especially bringing it forward) is common. Further study of these environments indicated that SDBR with the minor modification can be applied perfectly in these environments. The enhancement solution elements have been applied to a local company whose environment has the three characteristics. The application result demonstrates the enhancement is necessary.

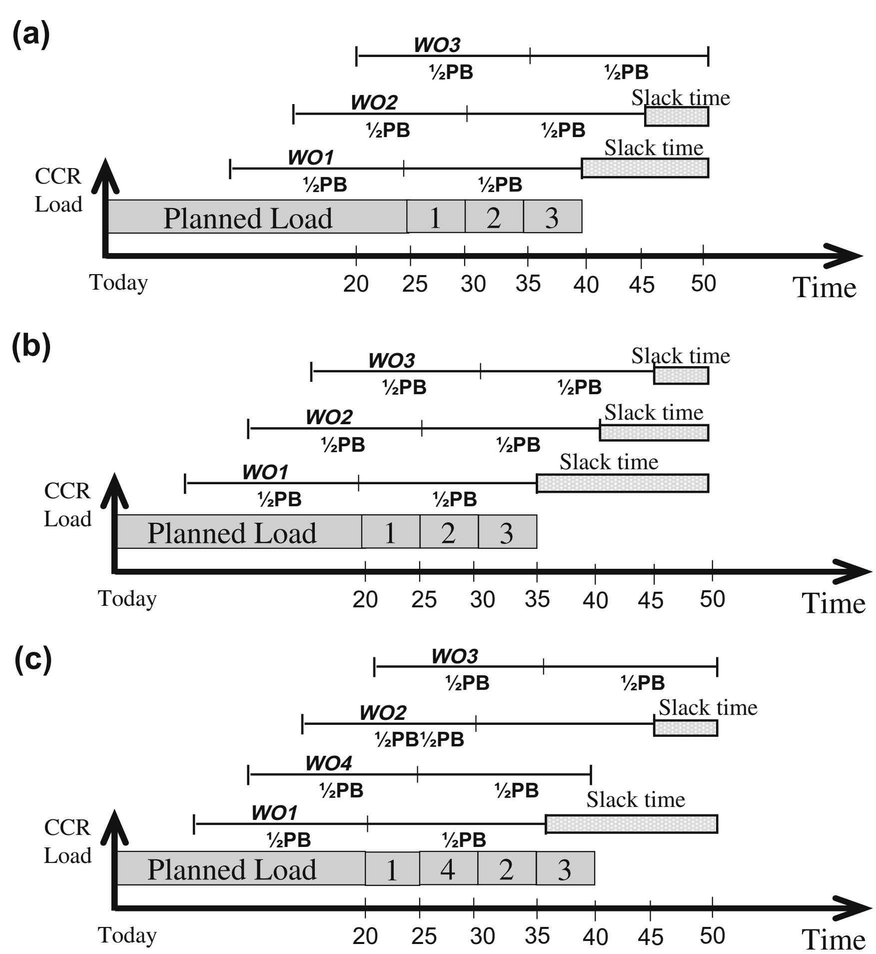

Fig. 9. (a) No slack time for inserting order. (b) With slack time for order insertion. (c) Result after order insertion.

References

Barnard, A. (2006). Challenge one of the basic laws of economics (&TOC). In *TOCICO international conference, Miami, FL*.

Breithaupt, J. W., Land, M., & Nyhuis, P. (2002). The workload control concept: Theory and practical extensions of load oriented order release. *Production Planning and Control, 13*(7), 625–638.

Chang, J. G. (2008). *Enhance the application of SDBR in a MTO environment*. Master thesis. Department of Industrial Engineering and Management, National Chiao-Tung Univeristy, Taiwan.

Chiang, S. Y., Kuo, C. T., & Meerkov, S. M. (2000). DT-bottlenecks in serial production lines: Theory and application. *IEEE Transactions on Robotics and Automation, 16*(5), 567–580.

Chung, S. H., & Lai, C. M. (2006). Job releasing and throughput planning for wafer fabrication under demand fluctuating make-to-stock environment. *International Journal of Advanced Manufacturing Technology, 31*(3/4), 316–327.

Chung, S. H., Pearn, W. L., Lee, A. H. I., & Ke, W. T. (2003). Job order releasing and throughput planning for multi-priority orders in wafer fabs. *International Journal of Production Research, 41*(8), 1765–1784.

Dabbas, R. M., & Fowler, J. W. (2003). A new scheduling approach using combined dispatching criteria in wafer fabs. *IEEE Transactions on Semiconductor Manufacturing, 16*(3), 501–510.

Glassey, C. R., & Petrakian, R. G. (1989). The use of bottleneck starvation avoidance with queue predictions in shop floor control. In *Proceedings of the 1989 winter simulation conference* (pp. 908–917).

Goldratt, E. M. (1990). *The haystack syndrome: Sifting information out of the data ocean*. Croton-Hudson, NY: North River Press.

Goldratt, E. M. (1992). *The goal*. Croton-Hudson, NY: North River Press.

Goldratt, E. M. (1996). *Production the TOC way: A self-learning kit*. Croton-Hudson, NY: North River Press.

Goldratt, E. M. (2006). *Reliable rapid response strategy and tactics tree*. Goldratt Group.

Goldratt, E. M., & Fox, R. E. (1986). *The race*. Croton-Hudson, NY: North River Press.

Gorinsky, S., & Jechlitschek, C. (2007). Fair efficiency or low average delay without starvation. In *Proceedings of 16th international conference computer communications and networks* (pp. 424–429).

Grabot, B., & Geneste, L. (1994). Dispatching rules in scheduling: A fuzzy approach. *International Journal of Production Research, 32*(4), 903–915.

Graves, R. J., & Milne, R. J. (1997). A new method for order release. *Production Planning and Control, 8*(4), 332–342.

Katcher, D. I., Arakawa, H., & Strosnider, J. K. (1993). Engineering and analysis of fixed priority schedulers. *IEEE Transactions on Software Engineering, 19*(9), 920–934.

Lee, J. H., Hwang, Y. C., Wang, M. T., & Li, R. K. (2009). Why is high DDP so difficulty to achieve – an experimental study. *Production and Inventory Management Journal, 45*, 30–43.

Lilly, M. (2004). Implementing simplified marketing pull. In *TOCICO international conference, Miami, FL*.

Lin, Y. H., Chiu, C. C., & Tsai, C. H. (2008). The study of applying ANP model to assess dispatching rules for wafer fabrication. *Expert Systems with Applications, 35*(3), 2148–2163.

Lin, J. T., Wang, F. K., & Yen, P. Y. (2001). Simulation analysis of dispatching rules for an automated inter bay material handling system in wafer fab. *International Journal of Production Research, 39*(6), 1221–1238.

Lozinski, C., & Glassey, C. R. (1988). Bottleneck starvation indicators for shop floor control in semiconductor manufacturing. *IEEE Transactions on Semiconductor Manufacturing, 1*(4), 147–153.

Lu, S. H., & Kumar, P. R. (1991). Distributed scheduling based on due-dates and buffer priorities. *IEEE Transactions on Automatic Control, 36*(12), 1406–1416.

Mabin, V. J., & Balderstone, S. J. (2000). *The world of the theory of constraints: A review of the international literature*. Boca Raton, FL: St. Lucie Press.

Nandi, A., & Rogers, P. (2003). Behavior of an order release mechanism in a make-to-order manufacturing system with selected order acceptance. In *Proceedings of the 2003 winter simulation conference* (pp. 1251–1259).

Rippenhagen, C., & Krishnaswamy, S. (1998). Implementing the theory of constraints philosophy in highly reentrant systems. In *1998 winter simulation conference proceedings* (pp. 993–996).

Roser, C., Nakano, M., & Tanaka, M. (2002). Shifting bottleneck detection. In *Proceedings of the 2002 winter simulation conference* (pp. 1079–1086).

Schragenheim, E. (2006a). *Operations planning and execution – the TOC way*. Goldratt Group.

Schragenheim, E. (2006b). *Using SDBR in rapid response projects*. Goldratt Group.

Schragenheim, E., & Dettmer, H. W. (2000). *Manufacturing at warp speed: Optimizing supply chain financial performance*. Boca Raton, FL: St. Lucie Press.

Schragenheim, E., Weisenstern, A., & Schragenheim, A. (2006). What's really new in simplified DBR. In *TOCICO international conference, Las Vegas, NV*.

Tsai, C. H., Chang, G. T., & Li, R. K. (1997). Integrating order release control with due-date assignment rules. *International Journal of Production Research, 35*(12), 3379–3392.

Umble, M., Umble, E., & Murakami, S. (2006). Implementing theory of constraints in a traditional Japanese manufacturing environment: The case of Hitachi tool engineering. *International Journal of Production Research, 44*(10), 1863–1880.

Watson, K. V., Blackstone, J. H., & Gardiner, S. C. (2007). The evolution of a management philosophy: The theory of constraints. *Journal of Operations Management, 25*(2), 387–402.