Copyright © 2002 IFAC
15th Triennial World Congress, Barcelona, Spain

AN APPLICATION OF DBR AND BUFFER MANAGEMENT IN SOLVING REAL WORLD PROBLEMS

Yajie Tian, [*] Tetsuo Sawaragi, [**] Qiang Wei [**] and Nobuo Sannomiya [***]

[*] *Graduate School of Informatics, Kyoto University, Japan*
[**] *Graduate School of Engineering, Kyoto University, Japan*

[***] *Kyoto Institute of Technology, Japan*

Abstract: Any real world system must have at least one constraint to limit the system from achieving its objective. How to make good use of constraints is of vital importance to increase the efficiency of the system. In this paper, an approximate algorithm consisting of two stages of approaches is proposed for solving the real world problems. The first stage of approach called Basic Approach (BA) is proposed for searching a suboptimal solution with non-relaxing resources. The second stage of approach called Buffer Management Approach (BMA) is proposed for improving the solution obtained from BA and excluding the situations of violating restrictions by relaxing resources. According to the complicacy, diversity and limited resources of real world problems, an idea of relaxing resources is embedded in the second stage of approach. The methods of Drum-Buffer-Rope (DBR) scheduling and buffer management are used in the approaches for excluding the constraints in the process and increasing the efficiency of the system. The proposed algorithm is applied to solving the Loading Allocation and Scheduling Problems (LASP) in real world. By combining the two stages of approaches, the proposed algorithm is considered to be effective and adaptive for solving a real world problem with complicated restrictions.

Keywords: Approximate algorithm, Bottleneck block, Drum-Buffer-Rope scheduling, Buffer management, Loading allocation and scheduling problems.

1. INTRODUCTION

The Theory of Constraints (TOC) is a management science invented by Dr. Eliyahu M. Goldratt (Goldratt and Cox, 1992). The core idea in the TOC is that every real system such as a profit-making enterpriser must have at least one constraint. Because a constraint is a factor that limits the system from getting more of whatever is strives for, then a business manager who wants more profits must manage constraints (Noreen *et al.*, 1995). The techniques illustrated in TOC such as DBR scheduling and the five-step process for continuous improvement by focusing on the constraints can be applied to capacity-constrained job shops with high product diversity.

In this paper, we pay a particular attention to the methods of DBR scheduling and buffer management. We then apply the idea to construct an approximate algorithm for solving a real world problem called Loading Allocation and Scheduling Problems (LASP). As we know, real world problems are usually restricted by many complicated conditions, which are often self-contradictory and are changed with dynamic environment. Therefore, the situations of violating restrictions often appear in solving the real world problems and we cannot find even a suboptimal solution for the problems.

For solving LASP, we have observed the dispatching processes in the warehouses of some companies in Japan. As we observe, when an emergent situation happens which cannot be controlled by the predictive scheduling, the skilled workers can handle it adaptively, even though sometimes it is not an optimal way. Common steps used in the site are to identify the constraint first which causes a bottleneck in the process, and then to make a decision to exclude the constraint by using different methods for the different situations. Not like the computer, skilled workers have a high degree of adaptability to the emergent situations by using their years of experience.

In this paper, we try to combine the DBR scheduling and buffer management with the experience of skilled workers in site and construct an approximate algorithm for solving LASP. The concept of DBR scheduling and buffer management is redefined according to the characteristics of practical problems. The ideas of buffer classification and constraint relaxation are used to construct the approximate algorithm. Then the proposed algorithm is applied to solving LASP in the site.

2. DBR SCHEDULING AND BUFFER MANAGEMENT

TOC builds on a practical platform for aligning key strategies with local actions and for maximizing the return on a given set of resources in both short and long run. TOC is based on the following five-step focusing process:

1. Identify the constraint
2. Decide how to exploit the constraint.
3. Subordinate everything else in the organization to the decision to exploit the constraint.
4. Elevate the constraint.
5. Start over by finding the new constraint.

The five-step process is used in strategic planning, project management, process improvement, manufacturing continuous improvement, and any day-to-day factory scheduling (McMullen, 1998). Most production processes were explained in TOC using the analogy of a scout troop on a hike as shown in Fig.1. The troop is to tie a rope between the leading scout and the slowest scout in the line (i.e., the constraint). The leading scout then can never get ahead up to the length of the rope and the other scouts can close up any gaps that might temporarily develop, because they are faster than the slowest scout. This solution called DBR, constraints the scouts in front of the slowest scout to walk no faster, on average, than the slowest scout.

The three key words in DBR scheduling are "Drum", "Buffer" and "Rope". A drum is a strategic operation that has limited resources and determines the flow of work through the system. A system can go only as fast as the slowest or the most overloaded resource. This is called *constraining resources*, and the second step of TOC (exploitation) is provoked at the drum. A buffer is a pocket of time represented by work in the process that is reserved in ahead of the drum, the constraining resource, and the shipping due date. The rope is the length of time necessary to accomplish processes in ahead of the drum or ship dates.

More important thing is how to determine the length of the rope between the leading scout and the slowest scout which is the size of the working-in-process from growing. Sizing the buffer is difficult because it has two risks. Under sizing the buffer will leave the constrained capacity resource open to starvation and lost throughput for the entire plant. Over sizing the buffer will increase operating expenses and cycle time,

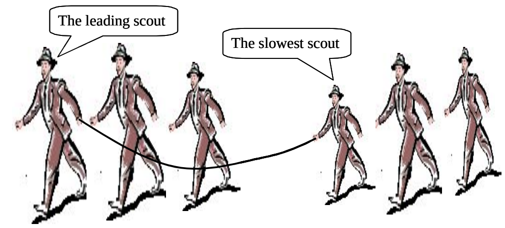

Fig. 1. Illustration of Drum-Buffer-Rope

and decrease inventory turns, resulting in decreased cash flow.

Smith (2000) divides the buffer into three zones, green, yellow and red zones (Smith, 2000). When a job does not enter the buffer on schedule, it creates a "hole" in the buffer. Holes in the green zone do not cause for concern. Holes in the yellow zone indicate an immediate need to locate the missing job and decide whether or not the job needs to be expedited. If the job does not arrive at the red zone before it at the constrained capacity resource, then the job needs to be expedited.

In the real world problems, a proper size of the buffer is usually determined by monitoring the buffer. During the initial estimation of part run-cycle times, being at least realistic and at best slightly inflated is preferred. In this paper, the idea presented by Smith is developed for constructing the approximate algorithm with a varying buffer. Because of the complexity of LASP, the size of the buffer is determined by at least three parameters, called relaxing parameters. The buffer is divided into three zones according to the requirements in the site and is adjusted by monitoring buffer.

3. DESCRIPTION OF LASP

In recent years, the software with artificial intelligence on allocating a truck or a container becomes very important and popular. The reasons are considered as follows: one reason is to decrease loading cost and the other is to increase loading speed. The skilled workers usually can allocate various large products to a container effectively and fast. But an unskilled worker cannot. However, companies usually make matters worse that the skilled workers are absent from work because of illness or some other things. On the other hand, when there are more than decades kinds of products with small sizes, even the skilled workers cannot handle them under a limited time and the result is not satisfactory sometimes.

The LASP described in this paper is to load and allocate many kinds of products with different sizes, weights and numbers into containers with many complicated restrictions as shown in Fig.2. We deal with such a data that all products have similar bottom areas and different heights and weights. According to the characteristics of problems, solving the LASP is divided into two processes. The first process is to allocate products into a series of blocks under satisfying the restrictions of loading blocks. The second process is to allocate the blocks into containers in two rows

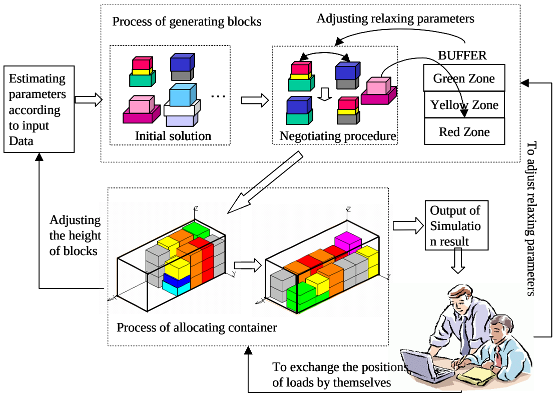

Fig. 2. Description of LASP

subject to the restrictions of allocating containers. The purpose of solving the problem is to achieve a high efficiency in the space and carrying capacity of a container subject to the restrictions. Because of the limited pages of the paper, we only discuss the first process in this paper, which is very important for the whole process of solving LASP.

In our earlier paper, an autonomous decentralized approach and extended Garbage Can Model (GCM) were proposed for solving LASP (Wei *et al.*, 2001). However, there was a problem not to be solved in that paper, that is, how to determine the extents of relaxing resources. In this paper, we try to obtain the optimal determination of relaxing resources so that we can make good use of the limited resources. The attention is focused to how to use of DBR scheduling and buffer management.

4. RESTRICTIONS AND DBR IN LASP

Since there are too many restrictions in the real world problems, it is necessary to divide them into hard and soft restrictions. The hard restrictions must be satisfied even though the loading efficiency is low. But the soft restrictions are allowed to be relaxed in different extents according to the different situations for increasing the loading efficiency.

4.1 *Restrictions*

$p_i^k(l_i^k, w_i^k, h_i^k, wg_i^k, n_i^k)$ expresses a kind of products in block b_k and the parameters l_i^k, w_i^k, h_i^k, wg_i^k and n_i^k are the length, width, height, weight and number of product p_i^k, respectively. $C_u(L_u, W_u, H_u, WG_u)$ expresses a container and the parameters L_u, W_u, H_u and WG_u are the length, width, height and the maximum carrying capacity of the container C_u, respectively. $b_k(bl_k, bw_k, bh_k, bg_k)$ expresses a block which consists of I_k products $p_1^k, p_2^k, \cdots, p_{I_k}^k$. The parameters $bl_k = \max_i\{l_i^k\}$, $bw_k = \max_i\{w_i^k\}$, $bh_k = \sum_{i=1}^{I_k} h_i^k$ and $bg_k = \sum_{i=1}^{I_k} wg_i^k$ are related to the length, width, height and weight of the block b_k.

4.1.1. *Hard restrictions*

$\mathbf{H_1}$: The height of a block cannot exceed the height of a container.

$\mathbf{H_2}$: The weight of a block cannot exceed the maximum weight restricted for a block.

4.1.2. *Soft restrictions*

$\mathbf{S_1}$: The products allocated vertically should satisfy

$$wg_{i+1}^k \leq wg_i^k \tag{1}$$

If Eq.(1) is not satisfied, the following relaxed restriction should be satisfied.

$$wg_i^k < wg_{i+1}^k \leq (1+\varepsilon_{wg})wg_i^k \tag{2}$$

$\mathbf{S_2}$: The products allocated vertically should satisfy

$$\frac{3}{4}l_i^k \leq l_{i+1}^k \leq l_i^k \tag{3}$$

$$\frac{3}{4}w_i^k \leq w_{i+1}^k \leq w_i^k \tag{4}$$

If Eq. (3) and (4) are not satisfied, the following relaxed restrictions should be satisfied.

$$(\frac{3}{4} - \varepsilon_l)l_i^k \leq l_{i+1}^k < l_i^k \quad \text{or} \quad l_i^k < l_{i+1}^k \leq \frac{l_i^k}{1-\varepsilon_l} \tag{5}$$

$$(\frac{3}{4} - \varepsilon_w)w_i^k \leq w_{i+1}^k < w_i^k \quad \text{or} \quad w_i^k < w_{i+1}^k \leq \frac{w_i^k}{1-\varepsilon_w} \tag{6}$$

where ε_l, ε_w and ε_{wg} are called the relaxing parameters related to the length, width and weight of products, respectively. By adjusting the values of the relaxing parameters, the situations of violating restriction are excluded. Product p_{i+1}^k is set to allocate on the product p_i^k. A rule in this paper is that the length l_i^k and width w_i^k of products p_i^k are set to correspond to the width W_u and the length L_u of container C_u, respectively.

$\mathbf{S_3}$: All the blocks should satisfy

$$\hat{H}_b - h_{av} \leq bh_k \leq \hat{H}_b + h_{av} \tag{7}$$

If Eq. (7) is not satisfied, the following relation should be satisfied

$$\frac{\hat{H}_b}{2} < bh_k < \hat{H}_b - h_{av} \tag{8}$$

where $\hat{H}_b$ is the estimated height of the block and $h_{av} = \frac{1}{2}\bar{h}$ and $\bar{h}$ is the average height of products. $\hat{H}_b = H_u - h_{av}$ is the initial value of $\hat{H}_b$.

$\mathbf{S_4}$: The same products should be allocated together.

4.2 *Objective function*

The objective function of a block is defined for evaluating the structure of block and judging which is the best negotiating procedure among blocks.

$$\begin{aligned} Z_k = \mu_1 \sum_{i=1}^{I_k-1} E_{li}^k + \mu_2 \sum_{i=1}^{I_k-1} E_{wi}^k + \mu_3 \sum_{i=1}^{I_k-1} E_{gi}^k \\ + \mu_4 \times E_h^k + \mu_5 \times E_v^k \end{aligned} \tag{9}$$

$$E_{li}^k = \begin{cases} \dfrac{l_i^k - l_{i+1}^k}{2\max\varepsilon_l \times l_i^k} & \text{for} \quad l_i^k \geq l_{i+1}^k \\ \dfrac{l_{i+1}^k - l_i^k}{\max\varepsilon_l \times l_i^k} & \text{for} \quad l_i^k < l_{i+1}^k \end{cases} \tag{10}$$

$$E_{wi}^k = \begin{cases} \dfrac{w_i^k - w_{i+1}^k}{2\max \varepsilon_w \times w_i^k} & \text{for } w_i^k \geq w_{i+1}^k \\ \dfrac{w_{i+1}^k - w_i^k}{\max \varepsilon_w \times w_i^k} & \text{for } w_i^k < w_{i+1}^k \end{cases} \quad (11)$$

$$E_{gi}^k = \begin{cases} 0 & \text{for } wg_i^k \geq wg_{i+1}^k \\ \dfrac{wg_{i+1}^k - wg_i^k}{\max \varepsilon_{wg} \times wg_i^k} & \text{for } wg_i^k < wg_{i+1}^k \end{cases} \quad (12)$$

$$E_h^k = \begin{cases} 0 & \text{for } \hat{H}_b - h_{av} \leq bh_k \leq \hat{H}_b + h_{av} \\ \dfrac{|\hat{H}_b - bh_k|}{\varepsilon_h \times bh_k} & \text{for } bh_k < \hat{H}_b - h_{av} \end{cases} \quad (13)$$

$$E_v^k = 1 - \frac{\sum_{i=1}^{I_k} v_i^k}{\max_i\{l_i^k\} \times \max_i\{w_i^k\} \times \max\{\hat{H}_b, bh_k\}} \quad (14)$$

where ε_h is the parameter related to the height. Both h_{av} and ε_h are determined by a large number of computational simulations. v_i^k is the capacity of products in block b_k. μ_i $(i = 1, 2, 3, 4, 5)$ are weights and $\mu_i = 1$ $(i = 1, 2, 3, 4, 5)$ are set in the approach.

4.3 *DBR in LASP*

The process of allocating blocks to containers requires that all blocks must satisfy the restrictions, at least the relaxed restrictions and have a similar height at the same time. If not, the restrictions of allocating container are difficult to be satisfied. The process can be explained in *The Goal* using the analogy of a scout troop on a hike. Each block can be considered as a scout and the slowest scout is considered as the bottleneck block that violates the restrictions and has the maximum objective value.

According to the real world problem, all blocks are divided into three classes. The first class consists of the blocks satisfying the non-relaxed restrictions given by Eqs. (1), (3), (4) and (7). It is called *Green Zone*. The second class consists of the blocks satisfying the relaxed restrictions given by Eqs. (2), (5), (6) and (8). It is called *Yellow Zone*. The third class consists of the blocks not satisfying even the relaxed restrictions. It is called *Red Zone*. The size of each zone varies with the relaxation of restrictions and with the property of the problem. The blocks in the *Red Zone* violate the restrictions, that is to say, bottleneck behaviors emerge in the process. The drum is set in the *Red Zone*. The buffer is a pocket of time represented by the blocks in the process and is considered as consisting of the above three zones in this paper. The rope is the length to link the *Green Zone* and *Yellow Zone* in front of the drum. The right length of the rope depends on the relaxation of restrictions and property of the problem.

In LASP, the relaxation of restrictions is based on the three parameters ε_l, ε_w and ε_{wg} which correspond to the length, width and weight of a product, respectively. Over sizing the values of parameters will cause the unnecessary relaxation of restrictions, which results in a low efficiency of blocks. Under sizing the values of the parameters will lose the stability of blocks and increase the number of blocks violating the restrictions.

5. APPROXIMATE ALGORITHM

5.1 *Basic Approach*

Assume ε_{wg}, ε_l and ε_w to be the initial values of relaxing parameters, respectively. t is the iteration number and t^* is the maximum iteration number. A block is called a manager if it is chosen as a negotiator and a block is called a contractor if it is chosen as the subject of the manager. The idea of negotiating procedure proposed in our earlier paper (Tian *et al.*, 2000) for solving production scheduling problems is used in this paper. S_t is the set of blocks and a block is deleted from S_t when all the negotiating procedures carried out in the block are failed. A_k is a temporary set of products chosen from manager b_k when the negotiation is failed. We have $A_k = \Phi$ only if the negotiating procedure is successful. The two sets are used in the algorithm for avoiding overlapping searches. $B = B_{green} \cup B_{yellow} \cup B_{red}$ is the buffer. B_{green}, B_{yellow} and B_{red} are the green, yellow and red zones, respectively.

Step1. Generate an initial solution $S_0 = \{b_1^0, b_2^0, \cdots, b_{N_b}^0\}$ and calculate the objective function Z_k^0 of each block and the sum of objective functions $Z_{sum}^0 = \sum_{k=1}^{N_b} Z_k^0$ of all blocks. Set $t = 1$.

Step2. If $S_{t-1} = \Phi$, then go to Step6. Otherwise, choose a block b_m^{t-1} with the maximum objective value as the manager. Three kinds of negotiating procedures (movement, insertion and exchange) are used in the approach.

Step3. In order to avoid the duplicated searches, a set A_m is set to record the products joined a failed negotiation. If $A_m = b_m^{t-1}$, then it indicates that any negotiating procedure cannot improve its solution in the current environment. In this case throw b_m^{t-1} to a zone of buffer according to its property. Let $S_{t-1} \Leftarrow S_{t-1} - b_m^{t-1}$ and go back to Step2. Otherwise choose a product $p_i^m \notin A_m$ from b_m^{t-1} as a negotiating origin and the contractors are chosen according to the property of p_i^m.

Step4. According to the behavior of the manager, the other agents judge whether or not they have eligibilities to join the negotiation according to the restrictions. Assume that R_m agents b_r^{t-1}, $r = 1, 2, \cdots, R_m, and r \neq m$ are eligible to cooperate with the manager b_m^{t-1}. If $R_m = 0$, then it indicates that there is no any agent to have the eligibility to cooperate with the manager in the current environment. If $A_m \neq b_m^{t-1}$, then go back to Step3. Otherwise throw b_m^{t-1} to a zone of buffer according to its property, then let $S_{t-1} \Leftarrow S_{t-1} - b_m^{t-1}$ and go back to Step2. If $R_m \neq 0$, judge whether or not there is a contractor to satisfy the following condition.

$$Z_{m,r} < Z_m^{t-1}, \text{ and } Z_{m,r} + Z_{r,m} \leq Z_m^{t-1} + Z_r^{t-1} \quad (15)$$

The condition requires that the agent taking a rope of the drum must be improved and the whole process does not become bad even though some agent becomes a little bad. If there are some agents to satisfy the above condition, then choose a

Table 1. Computational results

Buffer variation	Initial solution				Solution obtained from BA				Solution obtained from BMA			
	Z_{best}	G.Z.	Y.Z.	R.Z.	Z_{best}	G.Z.	Y.Z.	R.Z.	Z_{best}	G.Z.	Y.Z.	R.Z.
Data1	48.61	14	19	20	36.59	16	26	11	28.98	23	30	0
Data2	32.40	21	15	18	18.86	17	21	16	17.24	22	32	0
Data3	47.93	16	29	5	29.19	12	31	7	23.85	17	32	0
Data4	20.47	17	23	12	16.95	19	23	10	16.38	19	33	0

contractor satisfying the above equation and $Z_{m,r} = \min_{k \neq m}\{Z_{k,m}\}$ and go to Step5. Otherwise keep the last value of objective and the solution, i.e. let $Z_m^t = Z_m^{t-1}$, $Z_r^t = Z_r^{t-1}$ and let $t \Leftarrow t+1$. If $A_m \neq b_m^{t-1}$, go back to Step3, otherwise, go back to Step2.

Step5. Update the value of objective function $Z_m^t = Z_{m,r}$, $Z_r^t = Z_{r,m}$ and $S_t = S_{t-1}$. Let $t \Leftarrow t+1$ and $A_m = \Phi$. Go back to Step2.

Step6. If $S_t = \Phi$, and if $B_{yellow} \cup B_{red} = \Phi$, then the solution obtained from Step4 or Step5 is the best solution and end the negotiating procedure. If $B_{yellow} \cup B_{red} \neq \Phi$, it indicates that there are some agents violating restrictions and throw b_m^t into a zone of buffer according to its property and $A_m = \Phi$, then go to execute BMA.

5.2 *Buffer Management Approach*

BMA is different from BA in the following two aspects. One is that the relaxing parameters are increasing gradually within the monitoring buffer so that the freedoms in a manager's choice can be increased. The other is that the subjects of negotiation are limited only in the yellow and red zones of buffer, which depends on the available time of a user specified. The blocks in the red zone of the buffer are the bottleneck blocks, thus the efficiency of the whole process is increased with the improvement of the bottleneck blocks.

Step1. Get the initial values and the maximum values of relaxing parameters from skilled workers. Let $S_t = B_{yellow} \cup B_{red}$, and $B_{yellow} \cup B_{red} = \Phi$.

Step2. Let $\varepsilon_l \Leftarrow \varepsilon_l + \Delta\varepsilon_l$, $\varepsilon_w \Leftarrow \varepsilon_w + \Delta\varepsilon_w$ and $\varepsilon_{wg} \Leftarrow \varepsilon_{wg} + \Delta\varepsilon_{wg}$. The increments of $\Delta\varepsilon_l$, $\Delta\varepsilon_w$ and $\Delta\varepsilon_{wg}$ are set to be the one of fifth of each maximum value, respectively.

Step3. Execute the negotiating procedures similar to Step2$\sim$Step5 of BA. Since the relaxing parameters are enlarged, the choice opportunity is increased for each manager. Therefore, the success rate of the negotiating procedure is increased and the number of blocks in the red zone is reduced gradually.

Step4. If $S_t = \Phi$, and if $B_{red} = \Phi$, then the solution obtained from Step3 is the best solution and end the negotiating procedure. If $B_{red} \neq \Phi$, it indicates that there are some blocks violating restrictions and throw b_m^t into a zone of buffer according to its property and $A_m = \Phi$, go to Step5.

Step5. If $\varepsilon_l < \max \varepsilon_l$, then let $S_t \Leftarrow S_t \cup B_{red}$, $A_m = \Phi$ and $t \Leftarrow t+1$ go back to Step2. If $\varepsilon_l \geq \max \varepsilon_l$, but $B_{red} \neq \Phi$, it indicates that the maximum values of relaxing parameters cannot exclude the whole situations of violating restriction. Therefore, it needs to communicate with skilled workers for obtaining a new values of relaxing parameters or a solving indication.

Table 2. Relaxing parameters

Relaxing parameters	The second stage of approach		
	ε_l	ε_w	ε_{wg}
Data1	0.25	0.20	0.30
Data2	0.14	0.11	0.17
Data3	0.19	0.15	0.23
Data4	0.16	0.13	0.20

The values of relaxing parameters in the first stage of approach are given by:
$\varepsilon_l = \frac{\max \varepsilon_l}{5}$, $\varepsilon_w = \frac{\max \varepsilon_w}{5}$, $\varepsilon_{wg} = \frac{\max \varepsilon_{wg}}{5}$.

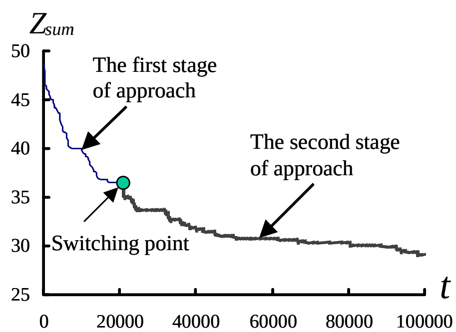

Fig. 3. Variation of the sum of objective values

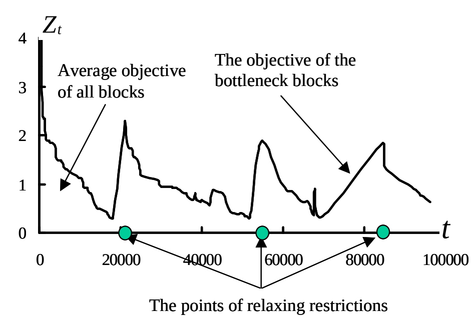

Fig. 4. The variations of average objective the objective for bottleneck blocks

6. COMPUTATIONAL EXPERIMENTS

Computational experiments are performed by using the proposed algorithm with the practical data obtained from the site. As examples, Table 1 shows the computational results of four kinds of data provided from users. We find from Table 1 that the block number in each zone varies in the two stages of approaches. In BA, the objective values reduce fast for all data, but the block number in the red zone does not reduce a lot. In BMA, the objective values reduce slowly, but the block number in the red zone reduces significantly. The reasons are considered as follows, one is that the relaxing parameters in BA is strict and with non-relaxing, but they are relaxed gradually in BMA; the other is that the negotiating procedures in BMA are focused on improving the yellow and red zones and some blocks located at green zone need to contribute their resources to improve the blocks located at the yellow and red zones. By doing so, the bottleneck blocks are excluded and the loading efficiency for the whole system is increased. The computation is over until the red zone is empty or the relaxing parameters reach their maximum values.

Table 2 shows the best values of relaxing parameters obtained from BMA. As we find from Table 2, the values of relaxing parameters vary with data and they do not need to relax to their maximum values given by the users for all data. The best values of the relaxing parameters can be obtained from BMA and within monitoring buffer.

Fig.3 shows the variation of objective sum of all blocks for data 1. We find from Fig.3 that the objective function converges fast in BA, but converges slowly in BMA. Even though only a local optimal solution is obtained from BA, but its computational time is much shorter than that of BMA. On the other hand, the situations of violating restrictions cannot be excluded completely only by using BA, but they can be excluded by using BMA. Therefore, the basic approach is not enough for solving the complicated problems, and the buffer management approach is essential. Users can determine which approach to be executed according to their time and purpose.

Fig.4 shows that the variations of the average objective for all blocks including the blocks in the buffer and the objective for the bottleneck blocks of data 1. The three peaks in Fig.4 indicate that a new relaxation of restrictions is started because all blocks cannot be improved under the current environment and there still exist bottleneck blocks violating restrictions in the buffer. In this case, the improving procedure is executed in the set $S_t = B_{yellow} \cup B_{red}$. By doing so, the blocks not to be improved in last relaxing situation may be improved in current relaxing situation. The procedure is repeated until the $B_{red} = \Phi$ or the relaxing parameters reach their maximum values.

7. CONCLUSION

An approximate algorithm consisting of two stages of approaches has been proposed for solving the real world problems with complicated restrictions, such as Loading and Allocation Scheduling Problems (LASP). A global negotiating procedure has been carried out in BA with non-relaxing the limited resources and a local negotiating procedure has been carried out in BMA with relaxing the limited resources. Both of the approaches have been constructed by using DBR scheduling and buffer management. Because the situation of violating restrictions exists in the real world problems, an idea relaxing the limited resources used in the site has been also embedded in the algorithm.

The methods of DBR scheduling and buffer management mentioned in this paper have been developed from those proposed by Noreen, et al. (1995), Smith (2000) and McMullen (1998). There are three parameters in the methods used to control the buffer size. The values of the parameters vary with monitoring buffer and the data. The maximum values of the relaxing parameters were given by the skilled workers. The optimal values of the parameters have been determined by monitoring buffer according to the properties of the real world problems.

The computational experiments show that BA with non-relaxing resources can only obtain a local optimal solution at a short time, but cannot deal with the situations of violating restrictions. BMA with relaxing resources can exclude the situations of violating restrictions by adjusting the size of buffer. By using the algorithm, the constrained bottleneck blocks have been improved gradually without increasing the size of buffer. The values of the relaxed parameters depend on the properties of data and the best values can be obtained by monitoring buffer gradually. This indicates that the proposed algorithm has presented a good method to determine the buffer size for solving LASP. Therefore, it is considered to be an effective and adaptive algorithm for solving LASP with complicated restrictions.

8. REFERENCES

Goldratt, E.M. and J. Cox (1992). *The Goal: A Process of Ongoing Improvement.* North River Press. USA.

McMullen, T. B. (1998). *Introductio to Theory of Constraints (TOC) Management System.* St Lucie Press. USA.

Noreen, E., D. Smith and J. T. Mackey (1995). *The Theory of Constraints and Its Implications for Management Accounting.* North River Press. USA.

Smith, D. (2000). *The Measurement Nightmare: How the Theory of Constraints Can Resolve Conflicting Strategies, Policies, and Measures.* St Lucie Press. USA.

Tian, Y., N. Sannomiya and Y. Xu (2000). A tabu search with a new neighborhood search technique applied to flow shop scheduling problems. *Proc. of the 39th IEEE Conf. on Decision and Control* pp. 4606–4611.

Wei, Q., Y.Tian and T.Sawaragi (2001). An extended organizational choice model applied to loading arrangement and scheduling problems. *Proc.of the IEEE International Conferenc on System, Man and Cybernetics* pp. 2605–2610.