APPLYING THEORY OF CONSTRAINT AND BOTTLENECK SCHEDULING APPROACH TO SOLVE PRODUCTION CAPACITY PROBLEM

Sumiharni Batubara, Rahmi Maulidya, Mega Rahma Pertiwi

Industrial Engineering Department, Trisakti University, Jakarta, Indonesia
E-mail: mimi_batubara@yahoo.com

ABSTRACT

Research is done on manufacturing of some automotive components made from rubber, which are produced based on order using flow shop process. At present the company faced a problem of unfulfilled daily production target. Based on initial research, there was unbalanced work station capacity and the company have not implemented scheduling system. The objective of the research is to apply Theory of Constraint in order to identify the work station facing difficulty, to provide time buffer to resolve lack of capacity and schedule the product using Bottleneck Scheduling method that minimizes makespan. The data processing outcome found one bottleneck work station where the required capacity is larger than the available capacity. The solution given is to provide time buffer carried out after working hours. Subsequently, order scheduling is done using Bottleneck Scheduling, generating a different work order sequence than the present work order sequence presently applied by the company. Scheduling using Bottleneck Scheduling require a smaller makespan than the makespan needed using the scheduling currently used by the company. If the order is completed using smaller makespan, then the production capacity can increase.

***Key words**: Theory of Constraint, Time Buffer, Bottleneck Scheduling, Makespan Minimization*

1. INTRODUCTION

Based on the initial research of the company, it is recognized that the problem facing the company presently is the inability to achieve daily production target to fulfill the orders. The cause of this problem is the needed machine capacity is bigger than the available capacity, creating *bottleneck* at a work station. The company have not utilized scheduling system.

Starting off with the current problem, the company wanted to make improvement in order to achieve the specified production target. The objective of the research is to apply Theory of Constraint to identify bottleneck work station and to provide time buffer to resolve the lack of production capacity and to apply Bottleneck Scheduling to minimize makespan and therefore can increase production target.

2. THEORETICAL BACKGROUND

2.1 Theory of Constraint

The theory of constraint approach is used to analyze the required capacity on work stations. *Theory of Constraint* is first developed by Eli Goldratt in the mid 1980. According to Goldratt, TOC is a theory covering all aspects of running an organization. The viewpoint of the theory is the performance of sistem constraints. The key definition stated by Goldratt is *“Constraints is anything that limits a system from achieving higher performance versus its goal”* (Dettmer, 1997).

Every corporation faces limited resources in the production of their product. These limited resources is defined as *constraint*. The theory of constraint acknowledge that company performance is restrict constraints. To improve company performance, to identify the constraints, to exploit the constraints in the long term, and find the solution to resolve it. In the following are the steps carried out (Bhardwaj, 2010) :

1. Identify the constraints
2. Determine how to exploit the constraints to improve performance.
3. Subordinate all parts of the manufacturing system to the support of step 2.
4. Carry out the steps necessary to improve the performance;
5. If in the previous step, a constraint has been broken or a new constraint develops, go back to step 1.

To identify the station experiencing constraint it is necessary to know bottleneck resource. Bottleneck resource is resource with capacity smaller than demand. Meanwhile Non-bottleneck Resource is resource with capacity larger than demand (Umble, 1996).

After identifying a bottleneck station, the remedial step is to add buffer to that particular station. This buffer also function to prevent the rate of production is not disrupted by the disturbance in the production system, therefore this buffer is also known as protective buffer. There are 2 types of buffer, that is:

1. Time buffer, which is the time used as buffer with the intent to safeguard the production system throughput from disturbance always happening in a production system.
2. Stock buffer, which is the end product or intermediate product used as buffer with the aim to fix the ability to fulfill the demand at production line therefore making the system able to finish the product in less time than the normal finish time.

The buffer used in this research is time buffer. Following is the calculation of time buffer needed :

$$TB = K - Ki \quad (1)$$

Where:
TB= time buffer
K = needed capacity
Ki = available capacity

2.2 Scheduling

Scheduling comprise of *job sequencing*, work start time and work finish time (*release and finish*), and *job routing*.
Objective of scheduling (Bedworth, 1987) is:

1. To increase utilization of resource and decrease idle time, in order to increase productivity.
2. To reduce job tardyness that has finsh time limit to minimize cost due to delay.
3. Reduce average job wait interval when available machines are performing other job.
4. To assist the decision making process on capacity planning and type of capacity needed to avoid costly additions.

Flow Shop Scheduling

Production process using flow shop has the same flow pattern from one machine to the other. Production flow shop is differentiated into 2 types:

a) Pure Flow Shop

The jobs follow the same production process flow.

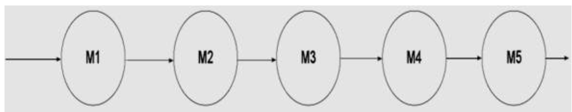

Figure 1. Pure Flow Shop

b) General Flow Shop

When production floor performs a variety of jobs but the job does not have to be done on all machines..

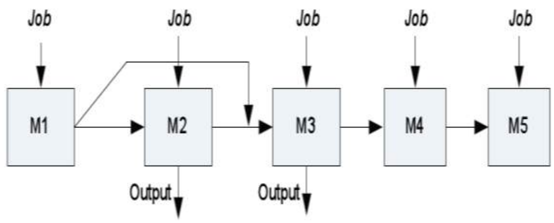

Figure 2. General Flow Shop

2.3. *Bottleneck Scheduling* Machine

Denote the bottleneck machine by b, and let j(b) be the operation oj job i done on b. We know the processing time of job i on machine b is $p_{IJ(b)}$; for convenience, let p'i = $p_{ij(b)}$. We also need to woory about what happens to i upstream and downstream of machine b. It is the release time of job i plus the time it takes i, r_i^b be the time job i arrives at machine b. It is the release time of job i plus the time it takes job i to get to the bottleneck machine; this includes processing and wait times for previous operations on upstream machines. Initially, we asume that there is no waiting, so we have r_b^i.=r_i+ $\sum_{l=1}^{j(b)-1} pil$ (2)

Estimate of waiting times from historical data or queueing results in the equation. Define a bottleneck due date for job i, d_i^b, that reflects when the operation on the bottleneck should be completed. To complete job i by its due date, it must be finished on the bottleneck by a time at least the sum of downstream operations processing times before the due date.
This again assumes there is no wait downstream.

$$d_b^i = d_i - \sum_{l=j(b)+1}^{m} pil \quad (3)$$

Schedule the bottleneck as a single machine with non zero release times. Let U be the set of unscheduled jobs and t be the current time. The procedure is
Set U = { 1,2,...,n} ; p'i = pij(b);i = 1,2,...,n ; dan t = min i ∈U r_b^i

1. Set S = {i I r_b^i ≤ t, i ∈U} be the available jobs. Schedule job i^* on b , where i^* has the best priority among jobs in S.
2. Set U ← U – {i*}. If U = Ø. Stop ,all jobs have been scheduled. Otherwise, set t = max{min i ∈U r_b^i ,t+p'i*} and go to step 1.

There are several priority rules that can be used to the algorithm. If the measure is makespans (Cmax), choose the available job with the most work remaining to LPT.

3. METHODOLOGY

Methodology for this paper is divided into three steps :

a. Defining the bottleneck machine
Theory of constraint can help analysing the bottleneck machine by checking the needed capacity and the available capacity. Needed capacity is obtained from the number of demand for each product times processing time for each product in each workstation divided by the number of machine. The equation for searching the needed capacity is :

$$\sum_{j=1}^{8} \frac{a_{nj} * D_j}{l_n} \quad (4)$$

Where :
a_{nj}= processing time of 1 unit product j in machine n; n = {1,2,3,4,5}; j = {1,2,3,…,8}
D_j= number of demand for product j (batch)
l_n= number of machine on workstation n

b. Adding time buffer
Time buffer is added for machine with available capacity smaller than needed capacity.
c. Schedule the bottleneck using bottleneck Scheduling and comparing with the result with Campbel Dudek Smith algorithm.
Bottleneck scheduling algorithm is divided into backward and Forward Scheduling and used bottleneck machine as a starting point to define a sequence. The steps are (Sipper, 1997) :

- Determine the release date
Release date (ri) come from the cumulative time after the bottlenech machine.
- Define the due date
Due date (di) is the total time for bottleneck minus the time of workstation after passing through the bottleneck machine..
- Define the processing time
Process time (pi) is the time for processing product n on the bottleneck machine.
- Define the sequence
Sequence is selected from the smallest release date. Release date can be used to find out the fastest release of a product and can improve the next process.

4. RESULT

This research is done in a manufacturing company which produce automotive component from rubber. The production of 5 product (j=1..5) in batch process, where 1 batch is equal to 200 kg. The process will be delivered from workstation 1 until workstation 10 (n=1..10) as shown in Production Flow (Figure 3). The processing time for each bacth in each workstation and the demand for each product in batch is given in Table 1.

4.1 Identify the constraint

Theory of constraint analyze the required capacity on workstation. The required Capacity for M7 is :

- Needed capacity for M7 :

$$= \frac{(25 \text{ x } 3508{,}12) + (35 \text{ x } 2916{,}18)) + \dots + (34 \text{ x } 3215{,}75)}{2}$$

$$= 242.14 \text{ second}$$

- Available capacity for M7 :

$$22{,}5 \times 3600 \times 2 = 162.000 \text{ second}$$

The result in Table 2 shows that M7 is a bottleneck machine. M7 with 2 machine, can't fullfill the capacity demanded.

Table 2. Required capacity for each workstation

Machine	Available Capacity (sec)	Needed Capacity (sec)	Valid
M1	81.000	29.038,64	Yes
M2	81.000	30.683,56	Yes
M3	81.000	30.577,66	Yes
M4	81.000	45.244,06	Yes
M5	81.000	38.761,57	Yes
M6	81.000	80653,24	Yes
M7	**162000**	**242142**	**No**
M8	162.000	73.048,06	Yes
M9	81.000	62.659,53	Yes
M10	81.000	30.078,26	Yes

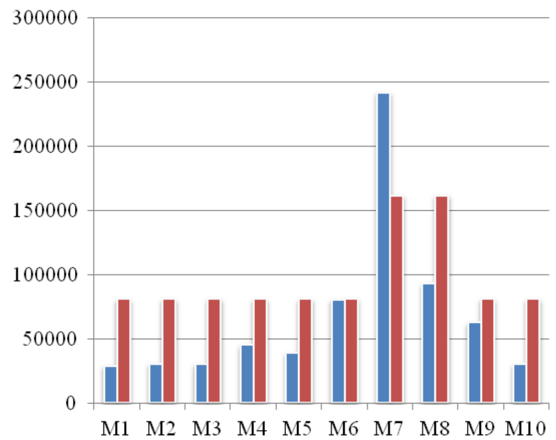

Figure 4. Available capacity (right-bar) and needed capacity (left-bar)

All workstation is categorized into Bottleneck and CCR (Capacity Constraint Resource) or both in Table 3.

Table 3. CCR-Bottleneck category

	Bottleneck	Non-Bottleneck	
CCR	M7	-	
Non-CCR	-	M1 M2 M3 M4 M5	M6 M8 M9 M10

4.2 Adding the Time Buffer

Time buffer is obtained using equation (1), where the needed capacity for M7 is 242.142 sec and the available capacity is 162.000 sec. So, the time buffer has to be 22,262 hour. Time buffer for M7 will be put in front of M7 as in Figure 5. The time buffer is allocated on Sunday, by adding one shift of production.

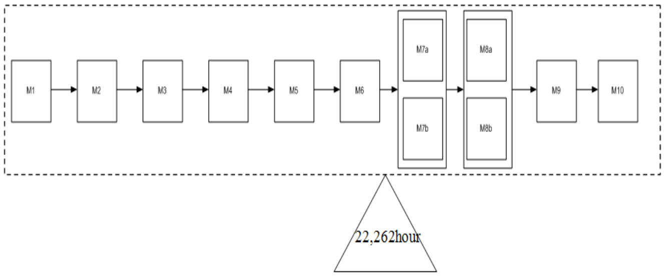

Figure 5. Time Buffer position

4.3 Schedule the batch using Bottleneck Scheduling

The bottleneck machine M7, will become the center of starting point for backward and forward scheduling. The result for backward scheduling in table 4

Table 4. Result for Backward scheduling

job i	1	2	3	4	5
ri	1968,601	1385,504	1705,108	1275,639	1694,81
di	13198,61	13781,71	13462,11	13891,58	13472,4
pi	3508,122	2916,182	3106,622	2420,536	3215,752

Where :
ri = release date
di = due date
pi = the processing time

Bottleneck scheduling gave result of a sequence J4-J2-J3-J5-J1. The sequence is scheduled for the batch with the smallest release data. The schedule is started from the bottleneck machine as shown in Table 5.

Table 5 Schedul Machine Bottleneck

Scheduling machine 7b		
t	U	(S.C)
0	4	(0.2420,536)
2520,536	2	(2520,536.2916,182
3016,182	3	(3016,182.3106,622
3206,622	5	(3206,622.3215,752
3315,752	1	(3315,752.3508,122

After backward scheduling, then it continue with forward scheduling. Forward scheduling use data after bottleneck machine in Table 6. The result from forward scheduling is compared with the recent sequence and choosed the sequence of J4-J5-J2-J3-J1

Table 6. Hasil Perhitungan forward

job i	1	2	3	4	5
ri	2214,198	1658,75	1858,922	1481,089	1569,814
di	12953,02	13508,46	13308,29	13686,13	13597,4
pi	3508,122	2916,182	3106,622	2420,536	3215,752

In Table 7, it shows the result of recent scheduling for M10. And in table 8, it shows the result of bottleneck scheduling for M10.

Table 7. Schedule of Recent Method for M10

Mesin	Job	Waktu (Detik)		
		Start	Total Waktu Proses	End
M10	1	7501,26	189,659	7690,92
	2	9859,81	191,845	10051,7
	3	10333,5	180,658	10514,1
	4	12309,1	175,891	12485
	5	15597,7	191,724	15789,4

Table 8. Schedule from bottleneck scheduling for M10

Mesin	Job	Waktu (Detik)		
		Start	Total Waktu Proses	End
M10	4	5001,37	175,891	5177,26
	2	8079,26	191,845	8271,11
	3	8552,92	180,658	8733,58
	5	11206,2	191,724	11397,9
	1	11852,6	189,659	12042,3

The sequence for recent method, J1-J2-J3-J4-J5 have makespan of 15789 sec, while the sequence of bottleneck scheduling is J4-J2-J3-J5-J1 and have makespan of 12042 sec. So, the percentage of makespan improvement is 37,4%.

In this paper, we compare the algorithm of FCFS, bottleneck scheduling and CDS algorithm. As shown in Tabel 9, the criteria of comparison is effect of scheduling for applying time buffer, CCR-Bottleneck, non CCR non bottleneck and makespan.

Table 9. Summary of Scheduling Algorithm

Scheduling algorithm	FCFS	Bottleneck scheduling	CDS
Adding time buffer	Yes	No	No
CCR Bottleneck	Yes	No bottleneck	No bottleneck
Non CCR non Bottleneck	Yes	Yes	Yes
Makespan (sec)	12.383,40	11.465	12.081

5. CONCLUSION

This research show that the improvement of adding the time buffer must be followed by defining the schedule. Therefore, the comparison of three different scheduling algorithm can show the effect for adding time buffer.
Bottleneck scheduling is more dominan in minimizing the makespan than CDS algoritm. In this case, it can improve 37,4% of makespan.

6. REFERENCES

(a) Bedworth, David D, and James E. Bailey. (1987). *Integrated Production, Control Systems: Management, Analysis and Design*. Wiley, John & Sons, Incorporated. San Frasisco.

(b) Bhardwaj, Arvind, Gupta, Ajay and Kanda, Arun (2010). "*Drum-Buffer-Rope: The Technique to Plan and Control Using Theory of Constraints*". World Academy of Science, Engineering and Technology, No. 69.

(c) Dettmer, H. W. (1997). *Goldratt's Theory of Constraints: A Systems Approach to Continuous Improvement*. ASQC Quality Press, Wisconsin

(d) Pinedo, Michael. (2002). *Scheduling: Theory, Algorithms, and Systems*. New Jersey: Prentice Hall Inc. 2nd Edition.

(e) Sipper, Daniel and Bulfin, L Robert. (1997). *Integrated Production Planning and Control*. Mcgraw-Hill College.

(f) Umble, Michael and Mokshagundam L. Srikanth. (1996). *Principles for World-Class Excellence Synchronous Manufacturing*. The Spectrum Management Group: Connecticut.

AUTHOR BIOGRAPHY

Sumiharni Batubara is a lecturer in the Industrial Engineering Department, Trisakti University, Jakarta, since 1993. She received her Master degree in Industrial Engineering from Institut Technology Bandung. Her research interest are in the area of Production Planning & Control, Inventory Control and Just In Time. Now, she is a head of the Production System Laboratory, Industrial Engineering Department, Trisakti University. She is also a member of the Society of Industrial Engineers. Her email address is mimi_batubara@yahoo.com

Rahmi Maulidya is a lecturer in the Industrial Engineering Department, Trisakti University, Jakarta. She received her Master degree in Industrial Engineering from Institut Technology Bandung in 2005. Her research interest are in the area of Production Planning & Control. She is a member of Production System Laboratory, Industrial Engineering Department, Trisakti University. She is also a member of the Society of Industrial Engineers and Indonesian Ergonomic Society. Her email address is rmauli@yahoo.com.

Mega Rahma is an alumnus from undergraduate at Industrial Engineering Department, Trisakti University in 2012. She is also an Assistance of Production System Laboratory at Industrial Engineering Department. Trisakti University. Her research area are in production planning & inventory control, and scheduling.

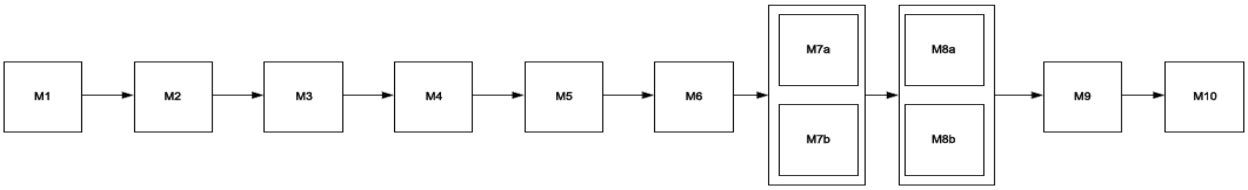

Figure 1. Production Flow

Table 1. Data Demand and processing time each station

j	*Dj*	a_{nj}									
		M1	M2	M3	M4	M5	M6	M7	M8	M9	M10
1	25	193,82	211,98	204,61	336,43	358,10	663,64	3508,12	1565,61	458,93	189,66
2	35	247	254,38	141,76	245,46	0	496,89	2916,18	1003,77	463,14	191,85
3	30	0	249,27	232,74	302,15	331,2	589,70	3106,62	1204,61	473,66	180,66
4	38	202,38	0	149,04	263,87	260,43	399,90	2420,54	992,04	313,16	175,9
5	34	231,11	264,76	231,01	269,12	293,40	405,39	3215,75	1117,3	260,79	191,7
I_n		1	1	1	1	1	1	2	2	1	1

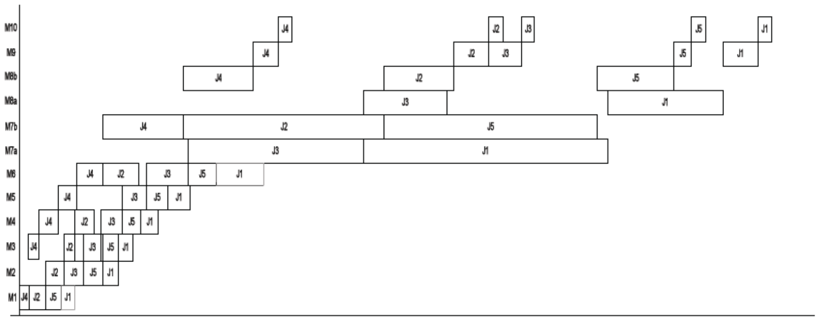

Figure 6. Gantt Chart after applying bottleneck scheduling