"BOTTLENECK DETECTION AND MITIGATION IN SERIAL PRODUCTION SYSTEMS"

Subodh Kumar[1], Arun Kumar[2] and Amit Kumar[3]

Department of Mechanical Engineering
Rajeev Gandhi Proudyogiki Mahavidhyalaya, Bhopal, India

ABSTRACT:
Process improvement programs and reengineering have taken a front seat due to a significant increase in competition amongst the giants in manufacturing industry. These programs require an accurate estimate of performance metrics such as throughput capacity of a given layout to justify projects and make valid comparisons between various restructuring options which would cost millions of dollars to implement. The focus of this research is on the analysis of serial production lines which are characterized by capacitated buffers, stochastic processing times, unreliable machines, rework loops, maintenance and operator issues. This constitutes a complex manufacturing system for which we have adopted discrete event simulation as a tool to predict the performance metrics. As a part of this thesis, a VBA project was undertaken in an effort to automate the process of building a simulation model in Arena 7.01 from an Excel template. We have also developed an algorithm which will automatically detect the bottleneck in a serial production system and suggest appropriate changes to the analyst with an objective of increasing the throughput of the system. These techniques are embedded in a ten step methodology presented in this thesis

KEYWORDS: bottleneck, inventory, Supply chain, buffers, serial production

INTRODUCTION:
A technique to detect bottleneck resources in complex serial production systems with the help of discrete event simulation is proposed in this thesis. A procedure for mitigating the bottleneck of any given serial production line is also developed. This will provide decision support to the analyst in the form of addition of buffers, reallocation of operators, or addition of parallel bottleneck resources. The objectives of this thesis are to identify the line bottlenecks and to use an iterative procedure to maximize line throughput. Our bottleneck detection technique is then contrasted with a traditional bottleneck detection method with the help of case studies.

Discrete Event Simulation:
Discrete-event simulation consists of a collection of techniques that, when applied to the study of a discrete-event dynamical system, generates sequences called sample paths that characterize its behavior" (Fishman, 2001). The term simulation in this thesis refers to discrete-event simulation. Simulation has evolved into a powerful decision support tool for manufacturing industries which is dominated by dynamic and stochastic variables. A serial production line can be viewed as a system that has resources (machines and operators) arranged in some predetermined order, which processes entities (units).

Formalization of a Typical Station:
A serial production line is characterized as *K* work stations arranged in series and each station is labeled as *k* where *k* = 1 to *K*. Let *N* be the number of products being processed in the line and each product is marked as *n* where *n* = 1 to *N*. The product mix is denoted by M_n, where M_n is the proportion of production allocated to product n. Then,

$$\sum_{n=1}^{N} M_n = 1.00$$

The processing time per unit (P) for the n^{th} product at station *k* is denoted as $P_k(n)$.
Some of the notations used in this research are:
T is the total production time
F_k is the expected time lost due to repair
G_k is the expected time lost due to replenishment
B_k is the % of time spent by station *k* in blocked state

Processing Parameters:
Table 1.

Some other notations are:

ISSN: 2278 – 7798

Attribute	Expression	Symbol
Processing time	Stochastic	$P_k(n)$
Time to failure	Stochastic	L_k
Cycles to exhaust	Stochastic	C_k
Time to repair	Stochastic	R_k
Time to replenish	Stochastic	H_k

I_k is the Maximum units that can be produced in station k without considering failures and exhausts

$U_k = T - F_k$ is the available processing time in station k after the expected total repair time has been removed.

$V_k = T - F_k - G_k$ is the available processing time in station k after the expected total repair and replenish times are removed.

J_k is the maximum units that can be produced in station k by considering repair and replenishment.

Methodology :

Data Collection

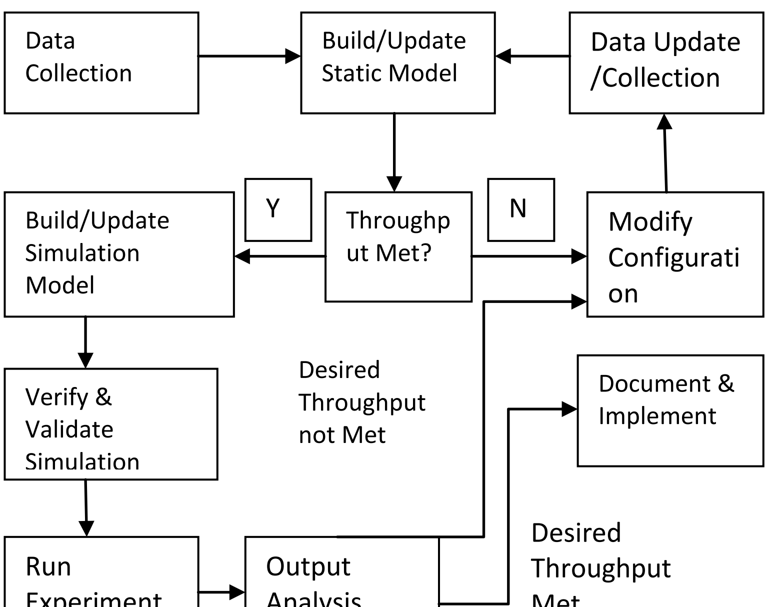

Fig1. Methodology for Simulation Modeling and Analysis of Serial Production Systems

Output Analysis of the System :

The simulation results thus obtained will reveal the performance level of each station with respect to the seven states we have defined. The results have to be analyzed and effective measures are to be undertaken to enhance the efficiency of the system. To do this the analyst will have to identify the station which has the greatest impact on the performance of the whole system, which will be the bottleneck resource.

When a machine is in blocked or idle state it is ready to accept parts for processing but is unable to do so due to complex inter-process interactions. The machine which is least affected by the inter-process interactions will be the bottleneck resource. This transforms to the fact that the machine which has the least value for the % of time spent in idle and blocked states is the bottleneck resource.

The states of a machine are represented in the resource state graph which is generated from Arena with the help of VBA (Jadhav and Smith, 2005 and Jadhav, 2005). Three possible changes can be made in the system to counter the bottleneck problem. They are listed as follows:

a. Add buffers before or after the bottleneck process
b. Re-allocate the workforce to reduce the time spent in failure/exhaust
c. Consider parallel processing option for the bottleneck resource.

The decision to select the options stated above is not straightforward and hence we have developed an algorithm which will be helpful for the analyst in this regard. The processing steps are as follows:

Step 0: If Exhaust/failure state for any station is >= Z %, consider reallocating the operator for those stations

Step 1: If Blocked state of the bottleneck is > 0 %, then add buffers right after the bottleneck resource

Step 2: If Idle state of the bottleneck is >= Y %, then add buffers appropriately for machines upstream and stop

Step 3: If idle state of the bottleneck is > 0 %, add buffers right before the bottleneck resource

Step 4: If (Idle + Blocked) states of bottleneck resource are =< X %, add parallel resource and stop .

The values that the parameters X, Y and Z will assume are decided by the analyst and the management, which is going to be subjective and it will depend on the objective of the analysis that is being conducted. The values of X, Y and Z

ISSN: 2278 – 7798

parameters used in this thesis are assumed to be 5, 25 and 2 respectively.

CASE STUDIES:

The case studies conducted on four sets of production lines are presented in this chapter. We analyze ten station lines and fifty station lines with low and high variability in processing parameters. The production lines listed here can be thus be categorized into:

i) Long line with low variability in processing parameters , ii) Long line with high variability in processing parameters, iii) Short line with low variability in processing parameters and iv) Short line with high variability in processing parameters.

Experimental Setup:

In this thesis, we have conducted ten sets of experiments for each category of production line mentioned above. An experiment is setup by arbitrarily assigning mean values to the processing parameters. The length of the line and coefficient of variation (C.V.) of processing parameters are decided upon by the category in which it falls.

Data for the Stations:

Resource	Attribute				
	Processing Time	Time to Failure	Cycles to Exhaust	Time to Repair	Time to Replenish
Machine A	16.72	84.75	25	313.01	97.07
Machine B	17.86	82.07	25	326.76	81.98
Machine C	16.62	65.12	31	270.59	87.65
Machine D	19.54	72.54	23	302.26	98.05
Machine E	17.13	68.58	33	287.75	83.23
Machine F	17.06	88.33	34	307.07	87.24
Machine G	16.25	74.72	28	282.19	98.58
Machine H	18.87	84.68	40	288.56	74.11
Machine I	18.63	65.3	27	298.04	94.51
Machine J	16.61	72.73	37	292	80.45

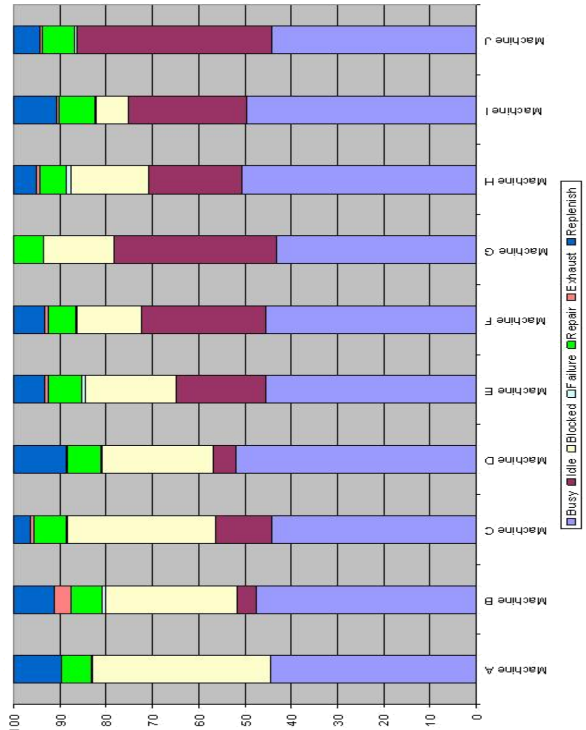

Fig 2 : Base Scenario

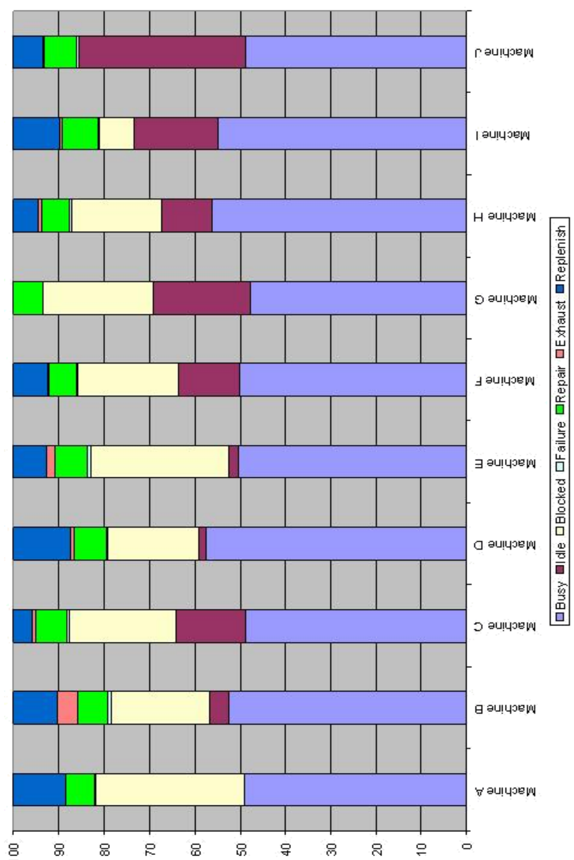

Fig 3: Improved Scenario

Fig 2 represents the base scenario and Fig 3 represents the improved scenario after calculating the different utilizations of machines and finding the bottleneck. In the graphs 0-100 represents the % ulitization.

ISSN: 2278 – 7798

RESULTS:

CASE (LONG LINE LOW VARIABILITY)

CONFIGURATION			
Attribute	**Mean Value**	**Unit**	**C.V B/W Process**
Processing Time P_k	18	Sec	0.08
Time to Failure L_k	65.5	Min	0.11
Cycles to Exhaust C_K	130	Parts	0.09
Time to Repair R_k	120	Sec	0.13
Time to Replenish H_k	140	Sec	0.12
Results			
Initial throughput:114,569			

	Bottleneck Index Method	**Highest Utilization Method**
Bottleneck	Machine AQ	Machine AQ
Mitigation	Add 10 buffers in front of machine AQ and 5 behind it	Add 10 buffers in front of machine AQ and 5 behind it
Throughput	**127,498**	**127,498**

Conclusion: We have developed a technique (Bottleneck Index Method) to detect the bottleneck resource in any serial production system by effectively using the resource state statistics, which records the % of time spent by a station in the seven different states discussed in this thesis. This technique of ours is compared to that of the traditional bottleneck detection method, where in the resource with the highest utilization is considered to be the bottleneck. We analyze ten station lines and fifty station lines with low and high variability in processing parameters. The production lines analyzed in this thesis can be thus be categorized into:

i) Long line with low variability in processing parameters , ii) Long line with high variability in processing parameters, iii) Short line with low variability in processing parameters and iv) Short line with high variability in processing parameters.

CASE(LONG LINE HIGH VARIABILITY)

CONFIGURATION			
Attribute	**Mean Value**	**Unit**	**C.V B/W Process**
Processing Time P_k	14	Sec	0.31
Time to Failure L_k	60.85	Min	0.29
Cycles to Exhaust C_K	120	Parts	0.33
Time to Repair R_k	130	Sec	0.32
Time to Replenish H_k	100	Sec	0.29
Results			
Initial throughput:92,533			

	Bottleneck Index Method	**Highest Utilization Method**
Bottleneck	Machine X	Machine AT
Mitigation	Add 15 buffers in front of machine X and 20 behind it	Add 25 buffers in front of machine AT and 15 behind it and 4 buffers in front of and behind machine AR
Throughput	**111,221**	**107,092**

REFERENCES:

1. Blumenfeld, D. E. (1990). A simple formula for estimating throughput of serial production lines with variable processing times and limited buffer capacity, *Int. J. Prod. Res*., Vol. 28, No. 6, 1163-1182.
2. Burman, M.H. (1995). New results in flow line analysis, MIT, Thesis.
3. Chen, C.T., Yuan, J., and Lin, M.H. (2003).Transient Throughput Analysis using Sample Path Method for Serial Unreliable Work Centers, *International Journal of The Computer, The Internet and Management*, Vol. 11, No.1, 30 – 41.
4. Chiang, S.Y., Kuo, C.T., and Meerkov, S.M. (2000). DT-Bottlenecks in Serial Production Lines: Theory and Application, *IEEE Transactions on Robotics and Automation*, Vol. 16, No. 5, 567- 579.
5. Chiang, S.Y., Kuo, C.T. and Meerkov, S.M. (2001). C-Bottlenecks in Serial Production Lines" Identification and Application, *Mathematical Problems in Engineering* Vol. 7, 543-578.
6. Choong, Y.F., and Gershwin, S.B. (1986). A decomposition method for the approximate evaluation of capacitated transfer lines with unreliable machines and random processing times, MIT Laboratory for Information and Decision Systems Report LIDS-P-1476.

ISSN: 2278 – 7798