Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

A New Method of Dynamic Bottleneck Detection for Semiconductor Manufacturing Line

Zhongjie Wang*. Jun Chen, Qidi Wu***

** Department of Control Science and Engineering, Tongji University, Shanghai, 201804, China (Email: wang_zhongjie@mail.tongji.edu.cn)*
*** Shanghai embedded systems institute, east china normal university, Shanghai, 200241, China*

Abstract: A global definition of bottleneck degree is first proposed by introducing an exponential function, the value of bottleneck degree for each working station can be obtained through real time calculation and compared, based on which a new method of exponential dynamic bottleneck detection (EDBD) is presented. This method provides an efficient means for bottleneck detection. Experiments under light load and heavy load prove the validity of this method.

1. INTRODUCTION

The key instrument, e.g. lithography, on semiconductor manufacturing line is very expensive, so the enterprise will not buy many these machines, which leads to the appearance of bottleneck. Moreover, the WIP (working in process) level of different products is not the same, which results in local load imbalance, so bottleneck will also appear. If the imbalance can not be controlled, non-bottleneck would be the temporary bottleneck, which will reduce the productivity of system.

Bottleneck detection is quite important for improving the performance of semiconductor manufacturing line, but it is not an easy task, even for the definition of bottleneck there are different versions.

From the point view of the micro effect resulted from bottleneck, some scholars think bottleneck is a group of working stations or only one working station that restricts the yield (Goldratt, 1986, Goldratt, 1990). If time is lost at the bottleneck, then this lost can not be compensated. There are also some scholars deem that the bottleneck is the machine devoting the most to slowing or ceasing the system working (Lawrence, 1994, Lawrence, 1995).

From the point view of quantified targets, some scholars consider that the bottleneck is the machine with the highest utilization rate (Law, 2000). There are also some scholars think that the bottleneck is the machine with the most lots accumulating in its buffer (Lawrence, 1994).

Although the utilization rate of bottleneck is indeed high, there is more than one bottleneck on a actual product line, it is a difficult to describe the bottleneck. Furthermore, utilization rate depends on stochastic result, so the effect resulted from temporary production fluctuation can not be reflected, this method is not real time and is only suitable for steady system. Moreover, if WIP level suddenly increases, there must be a group of working stations with overload, then the utilization rate of the machines of this group of working stations is all 100 percent, so the overload degree can not be reflected.

Another method of bottleneck detection is based on the accumulating lots in buffers, i.e. buffer length, in front of the machine, in which the bottleneck is detected by calculating the total waiting time. The advantage of this method is that the change of dynamic bottleneck can be detected. But, the number of lots in buffers often fluctuates, so the bottleneck detected by this method often shift, it is difficult to find the long term stable characteristics of system (Roser, 2002).

The above methods explain the characteristics of the bottleneck from one aspect, but they all have some defects when employed. From the micro point view, it is difficult to demonstrate the validation of these methods. It needs quite much work to check the decrease of global performance by decreasing efficiency of some one machine for both simulation and practice. Furthermore, these methods may lead to wrong conclusion for products with different machining path.

Besides, some scholars deem that the bottleneck is the working station with the smallest machining capacity, i.e. the longest machining time. Two working stations will be in balance if their machining time is equal.

Obviously, if the machining time of different working station differs a lot, this method is quite valid for detecting steady bottleneck, but is unable to detect temporary bottleneck conduced by stochastic event. Even for simple Flow-Shop product line, this method can not detect the bottleneck correctly. Cox and Spencer (1972) pointed that the best way of detecting bottleneck is to inquire the workmen on the front line. Many factors lead to the uncertainty of bottleneck detection, but it can be affirmed that, it is far not enough to partition the load of working stations for semiconductor manufacturing line only by the definition of bottleneck.

Based on the literature, two targets are proposed to improve the bottleneck detection.

(1) Whether or not some parameters can be adjusted in bottleneck detection, so that the long term bottleneck or short term bottleneck is reflected and the result stability and validity is guaranteed.

(2) The traditional methods can not reflect the bottleneck degree for each working station, so whether or not feasible definition of bottleneck degree is presented to differentiate the light bottleneck and heavy bottleneck, the definition of bottleneck degree should be global.

In order to solve these problems, a new method of dynamic bottleneck detection based on exponential function (EDBD) is proposed.

2. THE ALGORITHM OF EXPONENTIAL DYNAMIC BOTTLENECK DEECTION (EDBD)

The EDBD algorithm is obtained by improving the method based on buffer length for multiple steps, so the characteristic of real time dynamic response of bottleneck detection is reserved. The main defect of the bottleneck detection method based on buffer length is that data fluctuation is not stable, so it is difficult to reflect the bottleneck degree of product line. The EDBD algorithm solved the problem with two steps.

Let QL_S denotes the buffer length of the working station s at time t.

Step 1. Exponential mapping of buffer length

Define the instantaneous bottleneck degree of the working station s is denoted by IBD, and introduce a function transform, i.e.

$$\mathrm{IBD} = E(QL_S) = 1 - e^{-QL_s/\mathrm{T}} \tag{1}$$

This transform function has the following characteristics.

(1) The function $E(x)$ converges quickly near the origin.

(2) The function $E(x)$ transforms all the number of accumulating lots into the range of (0, 1), which is benefit for presenting global standard bottleneck degree.

(3) The function $E(x)$ is stable quite well when the buffer length is long, sudden decrease of long buffer length will not affect bottleneck detection. However, it is sensitive to increase of short buffer length, which reflects the abrupt case of product line.

(4) The parameter T has engineering meaning. Let T is 70 to 80 percent of the upper limit of buffer length. If the buffer length is equal to or bigger than T, the corresponding working station is considered to be the bottleneck. The parameter T is called the bottleneck constant for the working station s, as shown in figure 2.1. For different working station T could be different according to practice.

Thus, the buffer length at any time is transformed to the instantaneous bottleneck degree IBD in the range of (0, 1). IBD is global, but not stable, and needs further management.

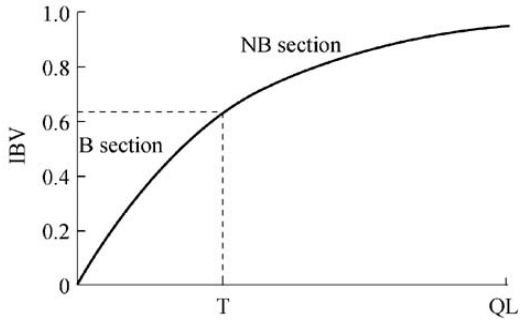

Fig.2.1 Mapping of the IBDV function and the meaning of T (B denotes bottleneck, NB denotes non-bottleneck)

Step 2. Smoothing of instantaneous bottleneck degree (IBD)

The transformed buffer length (IBD) has good transient dynamic characteristic, but it changes acutely on time sequence. If the traditional moving average method is employed to smooth IBD, time lag is prone to occur, which may destroy the dynamic performance of data. However, exponential smoothing is capable of adjusting historic and current data with different weight value, in which the newest data is considered. Therefore, in order to take account for both dynamic and steady performance, exponential smoothing method is employed for IBD, and the corresponding result is defined as the bottleneck degree (BD) of the working station s at time t.

$$BV_S(t) = \alpha \cdot IBV_S(t) + (1-\alpha) \cdot BV_S(t-1) \tag{2}$$

In equation (2), α is the smoothing factor, denotes the effect of historic and current data of BD. The value of α can be set according to data fluctuation on sequence. Initialize the buffer length for each buffer, and $BD_S(0)$ is equal to zero, then BD at any time is obtained.

The formalized definition of system bottleneck degree at time t can be given after the above two steps.

S_b is the current bottleneck of a system, if and only if,

$$BD_{S_b}(t) = \max_s BD_S(t) \tag{3}$$

It is worthy to point out that, the bottleneck constant T and smoothing factor α is closely pertinent to BD. If T is too big, the value of BD for each working station will approach to zero and there is no difference, otherwise, very acute temporary IBD will be prone to produce for some temporary bottleneck machine. Similarly, the value of α reflects the sensitivity of current BD to IBD and historic BD.

Besides, the EDBD algorithm has no too much demand for global restricts. The bottleneck constant for each working station is set individually according to practice, so that this method can be applied to actual semiconductor manufacturing line.

In addition, this method of bottleneck detection can also be applied to machine failure detection and machine maintenance.

3. SIMULATION MODEL

In order to demonstrate the validity of the EDBD algorithm, the product line model presented by Kumar (1972) is employed. The specific description of this model is shown in figure 3.1 and table 1.The working station number employed here is some what different from that presented by Kumar, which is in interest of consistency of programming.

Machine failure and maintenance is not considered in this model, the machining time of working stations is deterministic, there are no stochastic factors.

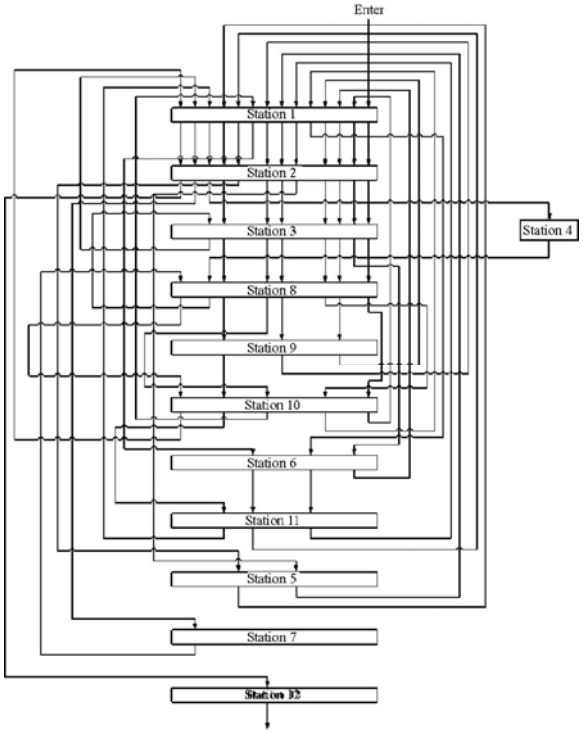

Fig.3.1 The machining technics of the model presented by Kumar

Table.1. The parameters of the model presented by Kumar (MN, VN, MT and LT denotes machine numbers, visiting numbers, machining time and load time respectively)

Station Name	number	MN	VN	MT	LT
Station 1	S0	4	14	30	105
Station 2	S1	3	12	22.5	90
Station 3	S2	10	7	150	105
Station 4	S3	1	1	108	108*
Station 5	S4	1	2	54	108*
Station 6	S5	2	3	72	108*
Station 7	S6	1	1	108	108*
Station 8	S7	4	8	48	96
Station 9	S8	1	3	36	108*
Station 10	S9	9	5	180	60
Station 11	S10	2	3	72	108*
Station 12	S11	2	1	150	75

4. EXPERIMENTS

Let all the initial buffer length are equal to zero, WIP level is 39, the simulation duration is 700000 time units. Lots are released into the product line with fixed WIP level, FIFO is employed for scheduling rule.

4.1 Calculation of bottleneck degree

The utilization rate of heavy load machine is shown in table 2.

Table.2 The utilization rate of heavy load machines

Working stations	S0	S2	S4	S5	S6	S8	S10
Utilization Rate (%)	96.52	96.88	96.69	95.88	94.03	95.33	95.15

It can be seen that, the utilization rate of the machines with same machining time is nearly equal to each other, so the dynamic performance can not be reflected. It can be seen from the next section the bottleneck obtained by utilization rate is not necessarily true.

Next the EDBD algorithm is employed to identify the bottleneck. Let the bottleneck constant of all working stations is equal to 5, and the smoothing factor is 0.05. It can be found from table 1 that, the machining time of the working station $S_3, S_4, S_5, S_6, S_8, S_{10}$ is the largest and same, the machining time of the working station S_0, S_2 is not but approaches the biggest. Therefore, the working station $S_0, S_3, S_4, S_5, S_6, S_8, S_{10}$ is selected for investigation.

Through calculation with EDBD algorithm, the value of BD for concerned working stations is obtained and denoted by $BD_0, BD_3, BD_4, BD_5, BD_6, BD_8, BD_{10}$, plotted as a curve, shown in figure 4.1 to figure 4.7 (the horizontal axis denotes the simulation time, the vertical axis denotes the bottleneck degree).

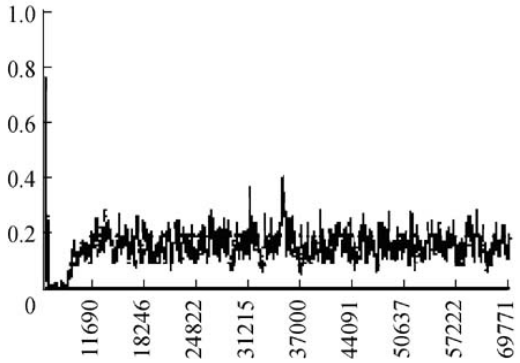

Fig.4.1 BD_0 of the working station S_0

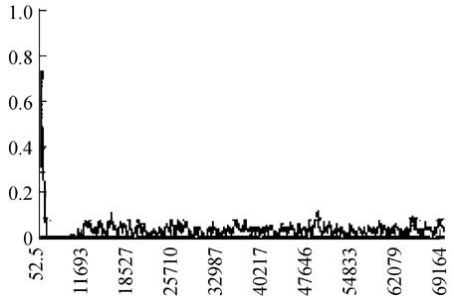

Fig.4.2 BD_3 of the working station S_3

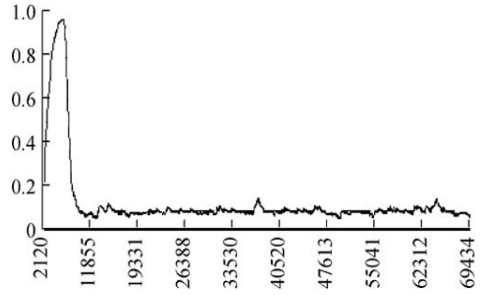

Fig.4.3 BD_4 of the working station S_4

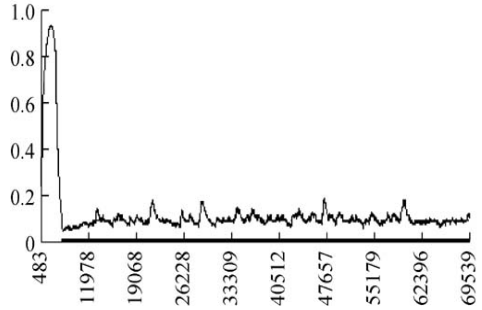

Fig.4.4 BD_5 of the working station S_5

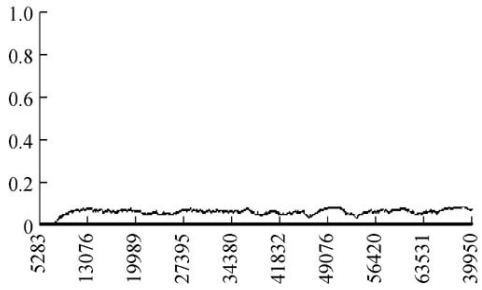

Fig.4.5 BD_6 of the working station S_6

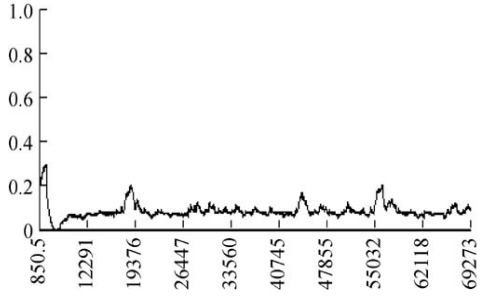

Fig.4.6 BD_8 of the working station S_8

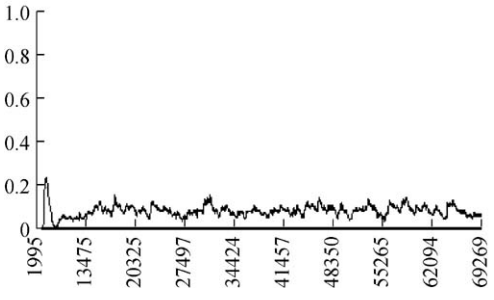

Fig.4.7 BD_{10} of the working station S_{10}

The above figures reflect the response rate of bottleneck degree under light load. It can be seen that, although the load time of each bottleneck is same, the dynamic response curve is different from each other. The working station S_4 and S_5 is very sensitive to lots inburst, the peak of bottleneck degree exceeds 0.9, but the working station S_6 and S_{10} are nearly not affected. Furthermore, the stability degree of each bottleneck degree is different. The bottleneck degree of the working station S_6 is very low and stable, while the working station S_0 has bigger bottleneck degree and the fluctuation of bottleneck degree is big too, although the load time of S_6 is not the biggest.

For comparison, the number of accumulating lots in buffer of the working station S_0 and S_3 is given individually, as shown in figure 4.8 and figure 4.9.

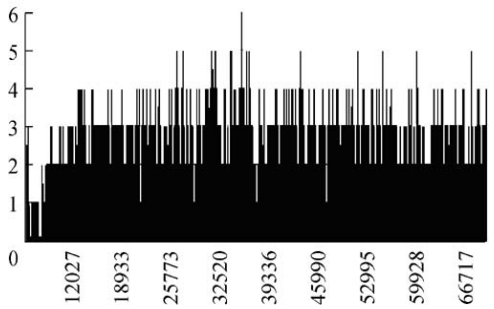

Fig.4.8 The number of lots in the buffer of S_0

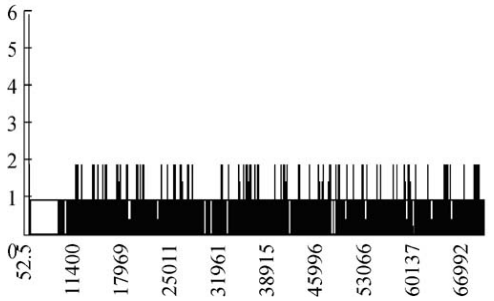

Fig.4.9 The number of lots in the buffer of S_3

4.2 Bottleneck shift detection

Based on the EDBD algorithm, real-time comparison of the bottleneck degree for each working station is done and then

the bottleneck shift at each time interval with current parameter setting is obtained, as shown in figure 4.10 (the horizontal axis denotes simulation time, the vertical axis denotes the number of working stations).

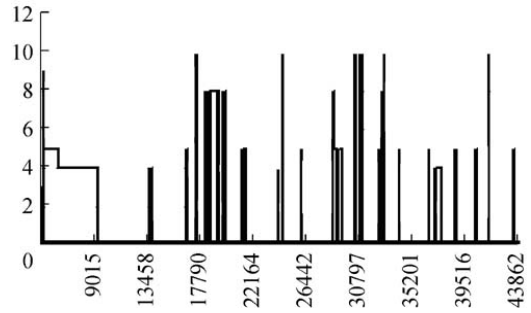

Fig.4.10 Bottleneck shift obtained by EDBD

It can be found from the above curve that, the working station S_4 and S_5 are in bottleneck for much longer time at the initial stage of production. Thereafter, the working station S_0 is often bottleneck.

It could be consider that, if the WIP level of the product line is 39 and CONWIP is employed for releasing policy, the long term bottleneck of the system is the working station S_0, while S_4, S_5, S_8 and S_{10} are prone to be temporary working stations.

For comparison, the detection result based on buffer length is given, as shown in figure 4.11 (the meaning of the axis is the same as that in figure 4.10).

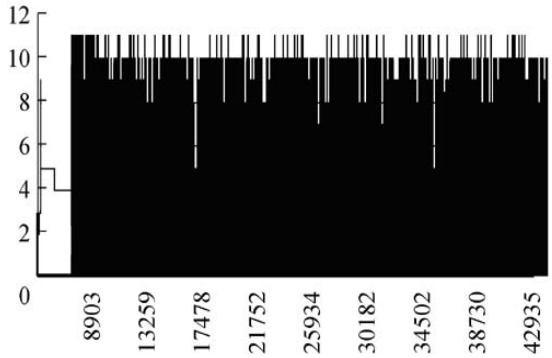

Fig.4.11 Bottleneck shift obtained by buffer length

Figure 13 shows that the bottleneck shift based on buffer length changes abruptly, so it is difficult to implement efficient long term optimization strategy for bottleneck.

3.3 Validation of EDBD

The method of validation for the EDBD algorithm is to improve the machining capacity at possible bottleneck and observe the performance improved of system. If the improved productivity of the system is the most after the machining capacity of some one working station is improved, then this working station could be considered to be the bottleneck. This method will be compared with the EDBD algorithm.

In order to verify the detection result obtained by EDBD algorithm under light load, a machine is added to the working station $S_0, S_3, S_4, S_5, S_6, S_8$ and S_{10} individually. The lots finished (LF) and average cycle time (ACT) is compared, as listed in table 3. The result of the original system is also listed in the last column.

Table.3 Comparison of ACT and LF before and after improvement of the machining capacity of working stations (AI denotes after improvement, BI denotes before improvement)

AI	S0	S3	S4	S5
ACT	4403.16	4436.84	4430.47	4420.35
LF	500	496	497	497
AI	S6	S8	S10	BI
ACT	4421.72	4426.18	4408.76	4442.13
LF	498	498	500	496

Through comparison we find that, the productivity improves the most after the machining capacity of the working station S_0 is improved, *so S_0 is the long term bottleneck under this case. Obviously, the actual result is same as that obtained by the EDBD algorithm*.

Next, the detection result under heavy load based on the EDBD algorithm is presented simply.

3.4 Bottleneck detection under heavy load

Supposed that a batch of lots with the size of 60 are released suddenly into the product line, and the WIP level is kept fixed, then figure 4.12 shows the detection result of bottleneck shift based on the EDBD algorithm under heavy load.

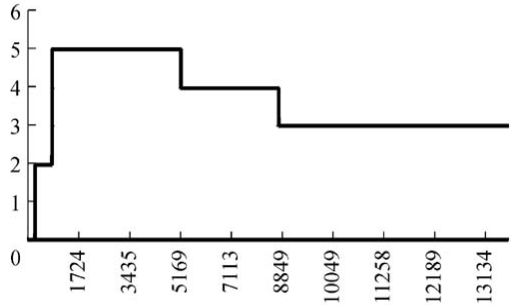

Fig.4.12 Bottleneck shift obtained by EDBD under heavy load

It is clearly seen that, the lots first impact the working station S_2 and S_5, then accumulate to a certain amount at the working station S_3 and does not increase any more at last. So, the working station S_3 is the long term bottleneck, the performance of the system will be improved the most if the working station S_3 is optimized. However, the utilization rate of more than one working station reaches 100 percent at this time, based on which it is incapable of identifying the real bottleneck.

Under the same circumstance, the detection result based on buffer length is presented, as shown in figure 4.13.

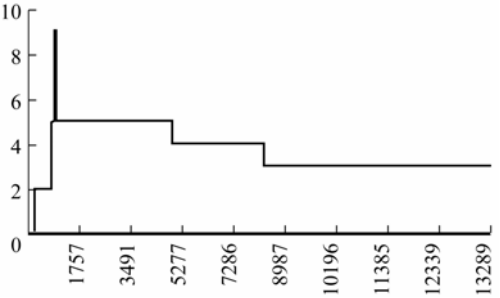

Fig.4.13 Bottleneck shift obtained by buffer length under heavy load

It can be seen from figure 4.13 that, the bottleneck shift between the working station S_5 and S_9 , S_4 and S_5 respectively, which is not benefit for bottleneck detection and further measurement. Therefore, good result of dynamic bottleneck detection under heavy load can be obtained by the EDBD algorithm.

4. CONCLUSIONS

For actual semiconductor manufacturing line, there may be more than one bottleneck. If the load time of different working stations is nearly equal to each other, the position of bottleneck will vary with the scheduling rules, releasing policy and WIP level. Furthermore, the bottleneck will change during transient process or its position will shift come and go between two machines. So, Bottleneck detection is not an easy task. However, it is very essential to improve the performance of semiconductor manufacturing line.

A global definition of bottleneck degree is first proposed, the value of bottleneck degree for each working station can be obtained by real time calculation and compared, based on which a new method of exponential dynamic bottleneck detection (EDBD) is presented. This method provides an efficient means for bottleneck detection. Experiments under light load and heavy load prove the validity of this method.

ACKNOWLEDGEMENT

The author and co-authors are appreciated the support of Project of Fok Ying Tong Education Foundation (No. 104030), Key Project of National Natural Science of Foundation of China (No. 70531020), Project of New Century Excellent Talent (No. NCET-06-0382) and Key Project of Education Ministry of China (No. 306023).

REFERENCES

Cox, J.F.I, Spencer, M.S. (1997). *The Constraints Management Hand book*, Boca, Florida: CRC Press-St. Lucie Press.

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

Goldratt, E.M.(1990). *Theory of Constraints*, Croton-on-Hudson, NY North River Press.

Kumar, S.E., Jones, L.Q. (1972). Scheduling Semiconductor Manufacturing Plants. *IEEE Control Systems,* **14(6),** p33-40.

Lawrence, S.R., Buss, A.H. (1994). Shifting Production Bottlenecks: Causes, Cures, and Conundrums. *Journal of Production and Operation Management*, **3(1),** p21-37.

Lawrence, S.R., Buss, A.H. (1995). Economic Analysis of Production Bottlenecks. *Mathematical Problems in Engineering*, **1(4),** p341-369.

Law, A.M., Kelton, D.W. (2000). *Simulation Modeling & Analysis*, McGraw Hill.

Roser, C.; Nakano, M.; Tanaka, M. (2002). Shifting bottleneck detection, *Proceedings of the Winter Simulation Conference*, 2:*1079*-1086.