

Simulation study of a bottleneck-based dispatching policy for a maintenance workforce

Rochak Langer^a, Jingshan Li^{a*}, Stephan Biller^b, Qing Chang^b, Ningjian Huang^b and Guoxian Xiao^b

^aDepartment of Electrical and Computer Engineering and Center for Manufacturing, University of Kentucky, Lexington, KY, USA; ^bManufacturing Systems Research Laboratory, General Motors Research & Development Center, Warren, MI, USA

(Received 30 May 2008; final version received 10 October 2008)

Maintenance is important for production operations and for continuous improvement. Appropriate dispatching of the maintenance workforce to quickly respond to equipment failures and carry out preventive services can improve system productivity. The first-come-first-served policy is typically used in many manufacturing industries. In this paper, we present a priority-based dispatching policy, a dynamic bottleneck policy, based on the analysis of real-time data. In such a policy, priority is assigned to the bottleneck machine after a fixed time period, and the maintenance worker will service the high-priority machine (i.e. bottleneck machine) first when multiple service requests are received. It is shown by extensive simulation experiments that this policy can lead to a greater improvement in system throughput compared with the first-come-first-served policy. To implement such a policy, the appropriate time period for data collection and the frequency for carrying out bottleneck analysis are investigated. In addition, a sensitivity study suggests that the results obtained are insensitive to machine downtime, efficiency, and reliability models.

Keywords: bottleneck; dispatching; maintenance; priority; throughput

1. Introduction

1.1 Motivation

This work is devoted to the study of dispatching policies for the maintenance workforce in manufacturing systems. Maintenance is critical for all manufacturing facilities. It plays an important role in a company to increase productivity, improve product quality and reduce costs.

In the manufacturing environment, all equipment is subject to random machine breakdowns. Frequent failures will lead to poor utilisation of the equipment, more scrapped parts, and loss of production. It may also call for early replacement of the machine due to its shorter life, which incurs higher capital costs. Moreover, in a manufacturing system, such a breakdown will not only result in disruption of a single machine, but will also propagate to other machines through blockages and starvation, leading to interruption of

*Corresponding author. Email: jingshan@engr.uky.edu

the production line, a delayed production schedule and unsatisfied customer demand. Thus, appropriate maintenance is needed for repair to return machines back to their working state, improve availability (which is typically referred to as *reactive maintenance* (RM)) and prevent potential failures by extending the machine life (referred to as *preventive maintenance* (PM)).

Due to the importance of maintenance activities, planning and scheduling play an important role in manufacturing systems. In all maintenance scheduling studies, workforce dispatching is of significant importance. In the modern lean manufacturing environment, each maintenance worker needs to be responsible for many stations and jobs. It is not unusual that one may have multiple tasks to be performed at the same time. Since the machines in the system interact with each other, different dispatching schedules will affect the overall system performance. This makes the scheduling of these tasks very important. The development of modern technology (such as sensors, RFID, etc.) provides the possibility of collecting production information in real time. Therefore, appropriate maintenance and repair assignments based on real-time information may lead to an improvement in system throughput and reduce system downtime and costs.

Currently, the most commonly used scheduling policy is the first-come-first-served (FCFS) policy, i.e. the maintenance worker will repair machines that have broken down based on the order of breakdowns. This implies that the machine that breaks down first is given priority and the others have decreasing priority in order of failure. However, in a manufacturing system, there is always a machine in the system that hampers system performance to a greater extent during a breakdown than the other machines. Typically, such a machine can be viewed as a bottleneck machine. In other words, a bottleneck machine is the machine that impedes system performance in the strongest manner, which implies that reducing the downtime (or waiting time) of a bottleneck machine will lead to the greatest improvement in system throughput compared with reducing the downtime of all other machines. Thus, prioritised maintenance and repair for the bottleneck machine can reduce the repair time of the machine (by reducing or eliminating its waiting time for the maintenance worker) and will lead to an improvement in system performance.

Although maintenance scheduling has attracted significant research attention, study related to the maintenance workforce dispatching policy is limited. The goal of this work is to present a dynamic bottleneck-based dispatching policy, investigate the impact of priority rules on the system throughput and provide insights into maintenance scheduling. Specifically, in this work, we study a serial production line using simulations. Three dispatching policies, first-come-first-served, constant bottleneck (CBN) and dynamic bottleneck (DBN) policies, are studied. The constant bottleneck approach involves the identification of the bottleneck machine in a selected time period and assigning priority to this machine to be repaired/maintained first throughout the complete operation or simulation period. The dynamic bottleneck policy is similar to the CBN except that the bottleneck is identified multiple times, i.e. after fixed, but much shorter, time periods, and priority is reassigned based on the identified bottleneck each time. Identification of the bottleneck machine is based on collected real-time data for each machine in the system. Thus, the DBN policy represents the maintenance response to real-time information in the system.

The main contribution of this paper is to present a dynamic bottleneck-based dispatching policy for scheduling the maintenance workforce and comparing it with the traditional first-come-first-served policy. By analysing the simulation results, it is concluded that the dynamic bottleneck policy is the most effective policy for all

the systems considered. We then examine the appropriate time period over which data should be collected and when the bottleneck analysis should be carried out to reassign priorities. It is shown that with a roughly daily frequency for identifying the bottleneck based on the data for the last one or two days, a better system throughput can be achieved. Moreover, sensitivity studies with respect to the effects of machine downtime, buffer capacity and coefficients of variation of machine up- and downtimes suggest that the system with the DBN policy is robust to such variations.

The remainder of the paper is structured as follows. Related literature is reviewed in Subsection 1.2. Section 2 formulates the problem. Section 3 describes the dispatching policies under consideration. A bottleneck identification method is introduced in Section 4. Analysis of the impact of the dispatching policies is presented in Section 5. Finally, Section 6 formulates the conclusions of the paper.

1.2 Literature review

A substantial amount of research has been devoted to the field of maintenance planning and scheduling (see, for instance, reviews by Scarf (1997), Dekker and Scarf (1998), Wang (2002), Garg and Deshmukh (2006) and Parida and Kumar (2006)). Much of the work concentrates on reliability studies of the machines, and planning or scheduling of maintenance activities from the equipment point of view. Some work is devoted to addressing the problem in a system environment, i.e. to integrate the maintenance and operational activities by considering the interactions of machines and work-in-process in a production line. For example, Schouten and Vanneste (1995) study the PM policy based on information concerning the age of the device and also the level of inventory in the downstream buffer. Boukas and Yang (1996) address the problem of the planning of the production and maintenance rates of a failure-prone manufacturing system to minimise costs. A two-machine line with a buffer between them is considered by Meller and Kim (1996) to determine the optimal buffer level that triggers preventive maintenance. Das and Sarkar (1999) study a production inventory system to investigate the PM policy with an (S, s) policy for inventory control. Marquez *et al.* (2003) discuss the maintenance policies for a production system constrained by its production rate and buffer capacity. Yao *et al.* (2005) study the joint preventive maintenance and production policies for an unreliable production-inventory system.

Simulation is one of the major tools used for the investigation of the impact of different policies (see the review on the use of simulation in maintenance by Andijani and Duffuaa (2002)). A simulation model for a just-in-time production system is introduced by Savasar (1997). Azadivar and Shu (1998) present a methodology for obtaining the optimal preventive maintenance policy using simulation-based optimisation. Gharbi and Kenne (2000) consider a multiple identical machine manufacturing system with random breakdowns, repair and preventive maintenance activities using simulation-based statistical tools. Cavory *et al.* (2001) study the optimisation of the schedule of the maintenance tasks of all machines in a single product line using simulation.

The workforce assignment problem has been studied in many areas. However, only a limited number of studies are related to maintenance workforce allocation. For example, Mosley *et al.* (1998) propose an assignment rule based on a priority value for each machine's downtime. Scheduling of the PM in a workforce-constrained environment, studied using evolution strategies, is studied by Ashire *et al.* (2000) by using

evolution strategies. However, such work does not consider the dynamic nature of the system. Real-time information is not included in the consideration.

In spite of these efforts, to the best of our knowledge the current literature does not provide any references related to the problem of maintenance staff dispatching using the bottleneck information of the production line. The goal of this work is to contribute to this end.

2. Simulation modeling and problem formulation

In this study, we consider a serial production line with both preventive and reactive maintenance operations. The maintenance staff (i.e. the repairman) is assumed to be responsible for both maintenance jobs. A simulation model using *Simul 8* (Haige and Paige 2001) is introduced as shown in Figure 1, where the square represents the work centre, consisting of two machines (represented by circles, with ‘R’ and ‘P’ denoting reactive and preventive maintenance operations, respectively), and the rectangles are the buffers. The following assumptions address the machines, buffers, maintenance worker and their interactions.

- (i) The serial production line consists of M work centres, and $M - 1$ in-process buffers separating each pair of work centres.
- (ii) For modeling purposes, each work centre consists of two machines, one being a regular machine subject to breakdown (i.e. having reactive maintenance operations), and the other is a maintenance machine with only preventive maintenance services. Each regular machine has a fixed cycle time, one part/time unit. Each maintenance machine has zero cycle time (for simulation modeling purposes, it is chosen as 0.01 time units per part).
- (iii) The regular machine is unreliable. Its up- and downtimes follow either exponential or non-exponential distributions. The average downtime, $T_{down,i}$, of regular machine i is randomly and equiprobably selected from $[1, 3]$ time units. The efficiency, e_i , of regular machine i is selected randomly and equiprobably from $[0.65, 0.95]$. The average uptime, $T_{up,i}$, is then calculated using

$$e_i = \frac{T_{up,i}}{T_{up,i} + T_{down,i}}$$

- (iv) The maintenance machine has a fixed preventive maintenance schedule, with each maintenance service taking 15 time units. The time between two preventive maintenance services is selected randomly and equiprobably from $[5000, 7000]$ time units, independent of the work it has performed.

Remark 1: It is typical that regular machine breakdowns have different characteristics than preventive maintenance services. In other words, the repair or service

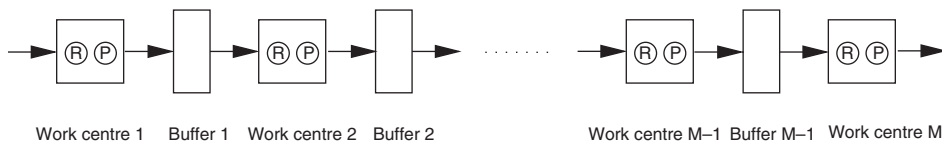


Figure 1. Serial production line with reactive and preventive maintenance operations.

patterns can be different. In addition, often, the scheduled maintenance is independent of machine breakdown history. Therefore, in order to describe and distinguish between them, we model the work centre as two individual machines, a regular machine and a preventive maintenance machine. Such modeling provides the possibility of introducing a control policy to integrate PM and RM by dynamically scheduling maintenance activities based on the status of the occurrence of different fault codes. To ensure that only one work item is placed in the work centre, the two machines are grouped with a limited (one) work count. Thus, from the point of view of the group, it functions as an individual machine with constant cycle time processing one part each cycle and having random breakdowns.

- (v) The buffers have finite capacity. Each buffer has a capacity randomly selected from [1, 8].
- (vi) Only one person is available for both RM and PM services. If both PM and RM are required on the same work centre, RM has higher priority.
- (vii) The repair staff will respond to any of the services immediately if one of the work centres (either PM or RM) is needed. When multiple work centres require servicing, the repair person will carry out the service according to the following policies: first-come-first-served, constant bottleneck and dynamic bottleneck policies.
- (viii) The maintenance worker will first complete the repair on the current machine if another service request is received during the repair, whether or not the new request has higher priority.
- (ix) Each work centre can be blocked if the downstream buffer is full and the work centre is ready to release a part. The last one is never blocked.
- (x) Each work centre can be starved if the upstream buffer is empty and the work centre is ready to load a part. The first one is never starved.
- (xi) It is assumed that the travel times are minimal so that they can be neglected.

Let PR be the throughput of the system, i.e. the average number of parts produced by the last machine per time unit. Also, introduce E as the utilisation of the repair person, i.e. the percentage of time the repair person is working. The problem addressed in this work is formulated as follows: given production system (i)–(xi), evaluate the system production rate and utilisation of the maintenance staff under different scheduling policies.

The simulation experiments for studying the problem are set up as follows. A warm-up period of 10,000 time units is taken to allow the system to reach steady state. The following 100,000 time units are used for statistical analysis with 20 replications. The 95% confidence interval is selected and is typically less than 0.1–0.5% of the expected values. In addition, the machine stoppage times (including repair and waiting for repair times), the blockage and starvation times are collected for priority assignment purposes. More than 100 lines with various configurations were generated and simulated for the analysis.

3. Dispatching policies

As mentioned above, three scheduling policies are considered in this work: first-come-first-served, constant bottleneck and dynamic bottleneck policies. The details of such policies are introduced as follows.

3.1 *First-come-first-served policy*

The first-come-first-served policy dispatches maintenance staff in the order that the machines break down. It is chosen as the baseline policy in this study since it is the default policy in most manufacturing industries. However, the policy fails to take into account the impact of the bottleneck machine on the system performance. Due to the nature of the bottleneck machine, the longer it waits for the maintenance staff to repair or perform maintenance on it, the greater its impact on the system production rate. Therefore, to counter this limitation, the bottleneck approach is introduced.

3.2 *Constant bottleneck policy*

The performance of any system is significantly affected by the bottleneck machine. If the bottleneck machine can be identified and prioritised for maintenance and repair work so as to reduce its waiting time for the maintenance staff, an appreciable increase in system performance can be predicted. The constant bottleneck policy refers to the identification of the bottleneck machine and assigning priority to it during the complete operation or simulation time period, T_{total} . Once a bottleneck machine is identified, the system is operated such that the maintenance staff will always repair the bottleneck machine first if multiple machines break down.

Clearly, since priority is assigned to the bottleneck machine only once during the complete operation time period, if the identified bottleneck machine is truly the bottleneck during the entire time horizon, this policy should lead to an improvement in system performance. However, if the bottleneck shifts to another machine during the operation, the 'real' bottleneck machine may need to wait for a longer time, which actually downgrades the system performance. This can happen since, when the bottleneck machine is assigned higher priority, other machines may need to wait to be repaired so that the total downtime will be longer, and they may finally become bottlenecks. In this case, wrong bottlenecks may be identified and inappropriate priorities assigned during the remaining time. Therefore, a dynamic bottleneck approach is presented to check bottlenecks more frequently.

3.3 *Dynamic bottleneck policy*

A dynamic bottleneck policy allows us to re-identify the bottleneck and re-assign higher priority to the bottleneck machine after a certain interval of time. Intuitively, the dynamic bottleneck policy should result in an improvement in performance if the bottleneck investigation is carried out at an appropriate time using a favorable time window for the analysis. However, to implement this policy, the following questions need to be answered. First, when bottleneck analysis is carried out, how long should the time period be that we trace back to collect data for the analysis? Let T_{data} denote the time period that data is collected and analysed for bottleneck identification. The DBN policy implies

$$T_{\text{data}} \ll T_{\text{total}}.$$

Then how large should T_{data} be? If T_{data} is too short, will the system status be stable or not? If it is too long, will the effect of the most recent changes in the system be represented or not?

The second question is related to how often bottleneck analysis should be conducted. Let T_f be the time period between two bottleneck analyses (here, subscript 'f' indicates the frequency of the analysis). However, selecting T_f is not trivial. If T_f is too long, the 'real' bottleneck may already have shifted to another one. If it is too short, will the new priority have an impact so that it still leads to consistent improvement or not? These are the questions to be answered in this study.

With these three policies, simulations are carried out to investigate the impact of each policy on system throughput and maintenance worker utilisation.

4. Bottleneck analysis

The priority assignment in both constant and dynamic bottleneck dispatching policies is based on the results of bottleneck identification. An identification method using the probabilities of blockage and starvation was introduced by Kuo *et al.* (1996), Chiang *et al.* (2001) and Li and Meerkov (2009) and is used throughout this work. Application of the analysis does not require any knowledge of the machine and buffer parameters. It is simply based on the data collected from factory-floor measurements. To make the paper self-contained, this method is briefly introduced below.

Consider a serial production line with M machines. The probability of a blockage, BL_i , and starvation, ST_i , of all machines in the system is measured, and then assigned by arrows directed from m_i to m_{i+1} , $i = 1, \dots, M-1$, if $BL_i > ST_{i+1}$, and from m_{i+1} to m_i when $BL_i < ST_{i+1}$. It was shown by Kuo *et al.* (1996), Chiang *et al.* (2001) and Li and Meerkov (2009) that if there is a single machine with both arrows pointing towards it, it is the bottleneck machine. In this case, higher priority is assigned to this machine. If there is more than one machine having both arrows pointing towards it, then multiple bottlenecks are identified. In the case of multiple bottlenecks, we assign equal higher priority to these machines and lower priority to others. (Kuo *et al.* (1996), Chiang *et al.* (2001) and Li and Meerkov (2009) show that a severity index can be calculated to identify the primary bottleneck machine. To simplify the control logic in our work, we ignore this part by assigning the same priority to all identified bottlenecks.) An illustration of the method is shown in Figure 2.

Remark 1: The rapid development in information and sensor technology has provided the possibility of collecting the necessary blockage and starvation data in real time. In many production lines, sensors have been installed to collect all the production data, such as working, stopping, blocking, starving, rejects, fault code, etc. Therefore, using blockage and starvation information to identify bottlenecks online becomes feasible.

5. Simulation results and analysis

In this work, the first-come-first-served policy is used as a baseline policy. All other policies, the CBN and DBN policies, are compared with the FCFS policy. The results of these comparisons are given below.

5.1 Constant bottleneck policy

To study the constant bottleneck policy, consider a 10-machine serial line constructed as follows. The uptime and downtime for all the machines are exponentially distributed.

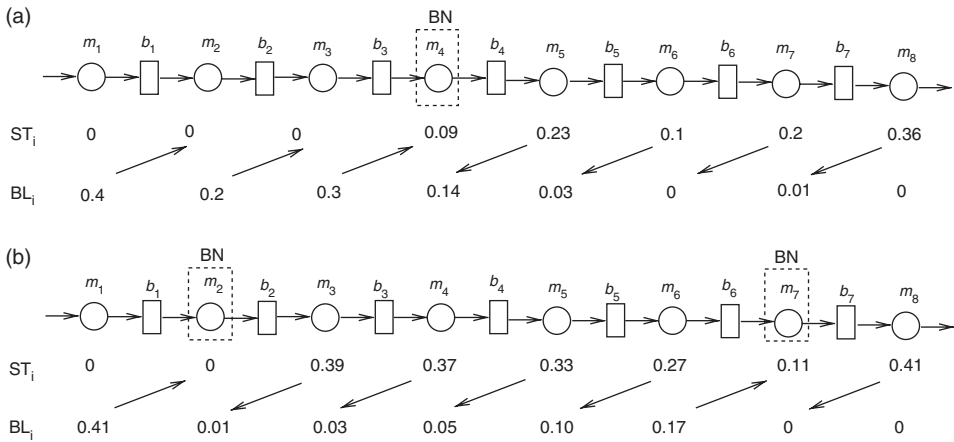


Figure 2. Bottleneck identification using probabilities of blockage and starvation. (a) Single bottleneck case. (b) Multiple bottlenecks case.

The machine and buffer parameters are selected according to assumptions (i)–(xi). The results of 10 representative examples of such lines are illustrated in Figure 3 for both production rate and maintenance worker utilisation. This shows that all lines exhibit similar performance for both the FCFS and CBN policies. Seven lines of the 10 have a slightly higher production rate for the constant bottleneck policy. A paired t -test suggests that $p=0.1172$, which implies that the difference between these two policies is not significant. Thus, we conclude that there may or may not be an increase in production rate for the CBN policy when compared with the FCFS policy. The rationale behind the result is that the bottleneck is shifting dynamically so that, during part of the remaining time, inappropriate priority is assigned, which results in no noteworthy improvement in the production rate. Similar results are obtained for maintenance staff utilisation for the two policies.

5.2 Dynamic bottleneck policy

To reduce the impact of a shifting bottleneck in the constant bottleneck policy, a dynamic bottleneck policy is introduced to carry out bottleneck analysis after a shorter interval of time, T_f (e.g., 5000 or 10,000 time units), rather than the complete time period, during each simulation. Identification of the bottleneck is based on the results of starvation and blockage percentages for the specified time period, T_{data} . In other words, blockage and starvation in the time interval $[kT_f - T_{data}, kT_f]$, $k=1, 2, \dots, \lfloor T_{total}/T_f \rfloor$, is used for the k th bottleneck analysis, and $\lfloor x \rfloor$ is the smallest integer greater than or equal to x .

5.2.1 Selecting T_{data}

The bottleneck identification method is based on average information for blockage and starvation in the steady state. Thus, a T_{data} that is too short may not be sufficient to establish steady-state estimates, and a T_{data} that is too long will not be able to identify the bottleneck dynamically. In order to determine an appropriate time period, T_{data} , to obtain reliable bottleneck information, we selected the time intervals for bottleneck identification

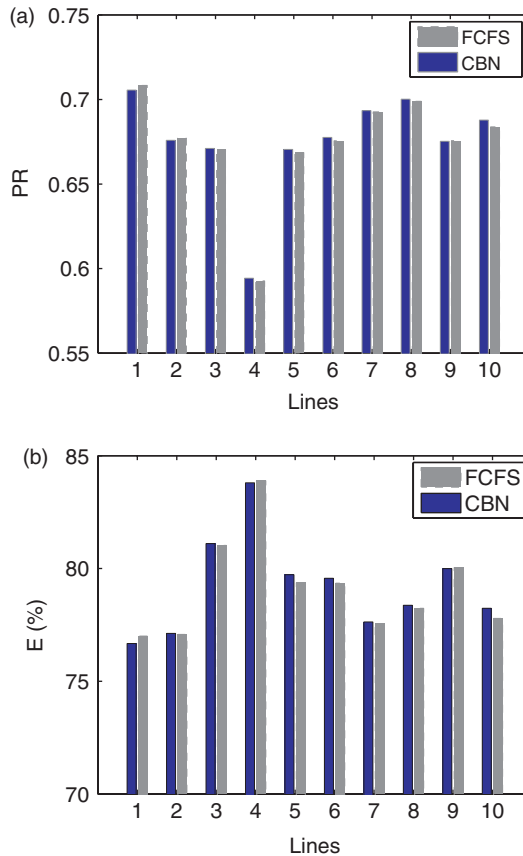


Figure 3. Comparison of the FCFS and CBN policies. (a) Production rate. (b) Utilisation.

as $T_{\text{data}} = 5000$ and 2500 time units. Results of a comparison of production rates and utilisation between the FCFS and DBN policies of 10 representative 20-machine lines are illustrated in Figure 4. Paired t -test results show that $p = 1.05E - 10$ between the FCFS and 2500 time unit DBN policies, and $p = 1.49E - 9$ between the FCFS and the 5000 time unit DBN policies. Clearly, the differences are significant and indicate that, with the dynamic bottleneck policy, the system production rate can be increased, with an improvement of about 5%. Moreover, the production rate for $T_{\text{data}} = 2500$ time units appears to be better than that for $T_{\text{data}} = 5000$ time units (where a paired t -test suggests $p = 9.8E - 9$). Also, utilisation of the maintenance staff is quite high, but there is almost no difference between the two policies.

To investigate how small T_{data} can be, additional simulations were carried out for $T_{\text{data}} = 2000, 1500, 1000$ and 500 time units. It turned out that this does not lead to a better production rate than for $T_{\text{data}} = 2500$ time units. In the case of $T_{\text{data}} = 1000$ and 500 time units, the results are similar to (and sometimes even worse than) that of the FCFS policy. An illustration of such experiments is shown in Figure 5. Again, this result justifies our intuition that too short a time period of data collection may be affected by random fluctuations and may not represent the nature of the system, and too long a period of data collection also leads to a slow response of bottleneck shifting.

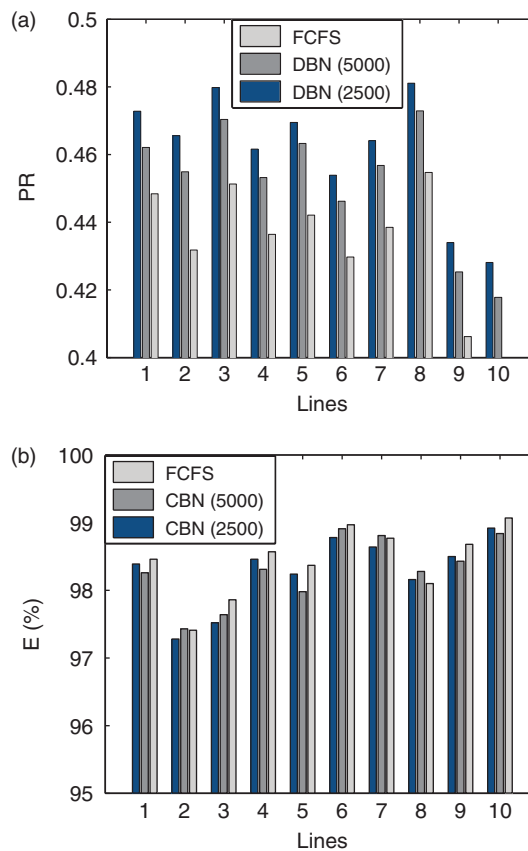


Figure 4. Comparison of the FCFS and DBN policies for 20-machine lines. (a) Production rate. (b) Utilisation.

Based on this result, we conclude that at least $T_{\text{data}} \in [1500, 2500]$ time units are needed in order to obtain steady-state data for bottleneck analysis. In the subsequent analysis, $T_{\text{data}} = 2500$ time units was used.

Remark 1: In practice, many automotive assembly lines operate at a cycle time close to 1 min. This implies that 1500–2500 time units represent about 25–41 h, i.e. roughly 1–3 days of operation (depending on the number of shifts per day). Thus, we need about half or a week of production data for bottleneck analysis.

5.2.2 Selecting T_f

Although an appropriate time period (e.g., 1500–2500 time units in this study) has been identified for bottleneck analysis, we still need to determine how frequently the analysis should be conducted. In other words, if we use $T_{\text{data}} = 2500$ time units, then how often should bottleneck analysis be carried out to re-assign the priorities? Therefore, the question is to determine T_f . To answer this question, experiments were carried out with $T_f = 500, 1000, 1500, 2000$ and 2500 time units. T_{data} is still selected as 2500 time units. Examples of ten 20-machine lines are shown in Figure 6. From the figure, one can see that,

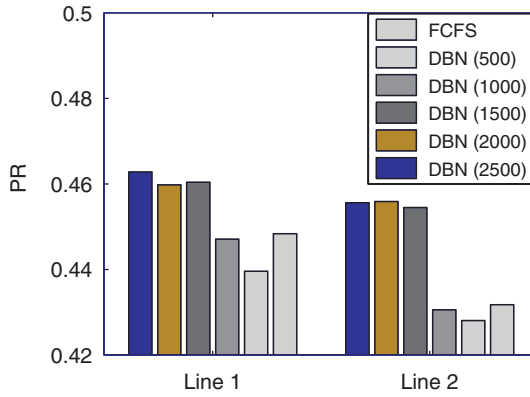


Figure 5. Comparison of the FCFS and DBN policies for 20-machine lines for smaller T_{data} .

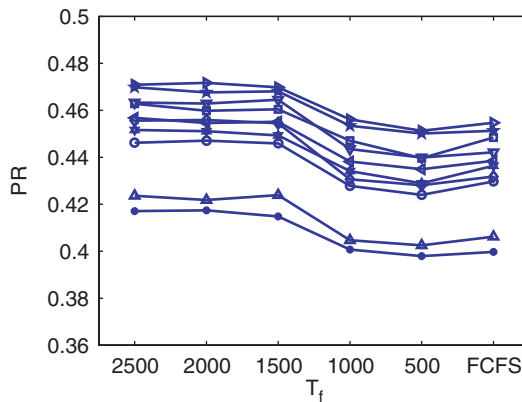


Figure 6. Comparison of the FCFS and DBN policies for 20-machine lines for selected T_f .

for all the lines considered, there is a significant decrease occurring after $T_f=1500$ time units. A paired t -test indicates that $p=3.16E-9$ between $T_f=1500$ and $T_f=1000$ time units, which suggests that the difference is significant. When $T_f \leq 1000$ time units, the DBN policy results in almost the same (or even worse) performance than the FCFP policy. When $T_f \in [1500, 2500]$, PR is relatively constant and the DBN policy consistently outperforms the FCFP policy. Additional experiments also show that if a larger T_f is used, e.g. 5000 time units, the production rate is not as good as in the case of $T_f=2500$ time units, although the difference is small. A possible explanation for this is that re-assigning priority needs time to have an impact. If we re-assign priority too often, the time between two re-assignments may be too short, and the re-assignment may not have any effect, with the system only chasing the variations. Hence, the results may be similar to those for the FCFS policy. Thus, a certain time period (e.g., 1500 time units) is needed to allow the new priority rule to have an impact on system performance.

Therefore, it is concluded that bottleneck analysis should be carried out every 1500–2500 time units (or one or two days) to re-assign priorities using the data for the last 1500–2500 time units (one or two days).

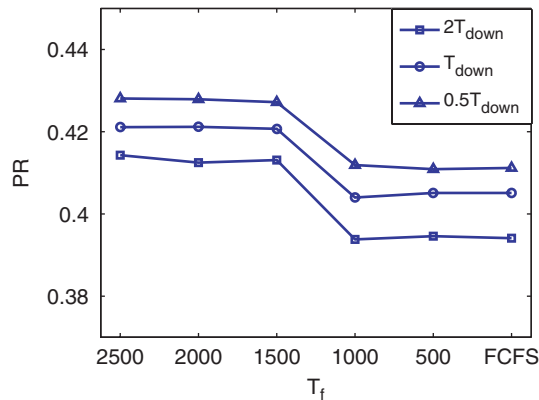


Figure 7. Sensitivity to the average machine downtime.

5.2.3 Sensitivity analysis

The above analysis suggests that using the dynamic bottleneck policy to dispatch maintenance staff could lead to an improvement in the system production rate. To further validate this result, an investigation of the sensitivity of such a result to variations in machines and buffers is needed. Therefore, we carried out a sensitivity analysis with respect to average machine downtime, efficiency, buffer capacity, and machine reliability models.

5.2.3.1 Sensitivity to machine downtime. The average downtime of a machine, T_{down} , is directly related to staff utilisation, line throughput and system bottlenecks. To study the impact of the average machine downtime on the DBN policy, we increased and decreased T_{down} and carried out the analysis with different T_f . One example of the results is shown in Figure 7. It can be seen that although the system throughput is reduced or increased when T_{down} is doubled or reduced by half, respectively, the qualitative behaviour does not change. This implies that the conclusion obtained above is robust to machine downtime.

5.2.3.2 Sensitivity to machine efficiency. When T_{down} is changed, the machine efficiency varies accordingly. This indirectly proves that the line behaviour should be insensitive to machine efficiency. To further validate this fact, we consider lines with identical machines, where each machine efficiency ranges from 0.6 to 0.95. Other parameters are selected according to assumptions (i)–(xi). The results of four such lines are illustrated in Figure 8. As can be seen, the results for the DBN policy are not sensitive to machine efficiency, which agrees with our expectation.

Remark 2: It has been reported in the literature that the buffer capacity is a function of the average machine downtime (Enginarlar *et al.* 2005a, b, Li and Meerkov 2009). Thus, a variation in buffer capacity is equivalent to a variation in machine downtime. Therefore, the impact of buffer capacity is also insignificant. In other words, the results that we have obtained are not sensitive to buffer capacity. Numerical experiments also justify this argument.

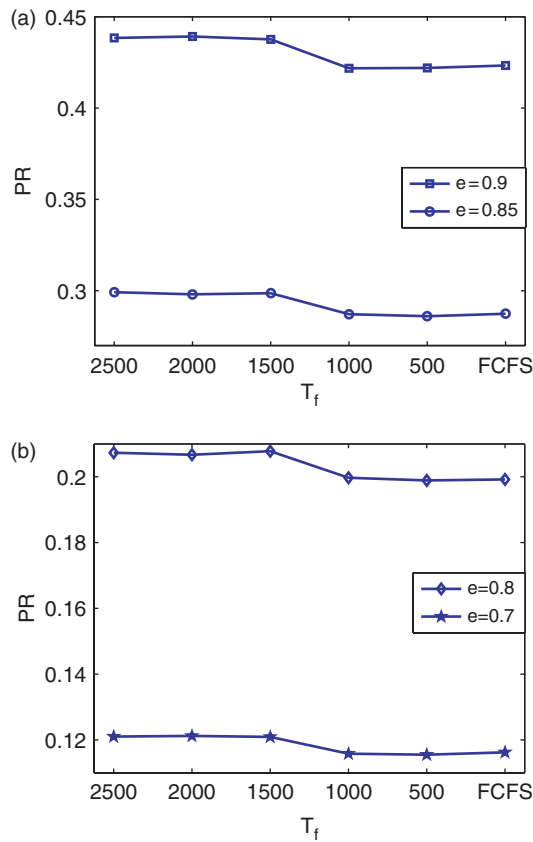


Figure 8. Sensitivity to machine efficiency. (a) $e=0.9$, $e=0.85$. (b) $e=0.8$, $e=0.7$.

5.2.3.3 *Sensitivity to machine reliability models.* It was shown by Li and Meerkov (2005a, 2009) that the throughput of a serial production line is not dependent on the type of distribution of the machine's up- and downtime when the repair resource is not an issue. Does this property still hold when only limited repair resource is available? To answer this question, a sensitivity study with respect to machine reliability models was conducted.

The results presented above are limited to an exponential reliability model of the machines. However, in practice, the distributions of the up- and downtimes of most equipment are not exponential. In other words, the coefficients of variation (CVs) are not equal to 1. In particular, CVs are most likely to be less than 1 (Inman 1999, Li and Meerkov 2005b). Therefore, we extended our study to non-exponential machines with CVs less than 1. Specifically, we first investigated the impact of different distributions of machine up- and downtimes, and then explored the sensitivity of the DBN policy with respect to the coefficient of variation. Weibull, Erlang and log-normal distributions were used for this investigation, since the two parameters involved enabled us to assign the desired coefficients of variation. The CVs were selected as 0.25, 0.5 and 0.75.

The results of this study are illustrated in Figures 9 and 10 for one 15-machine line. Figure 9 shows that the results of the DBN policy are not sensitive to the coefficient

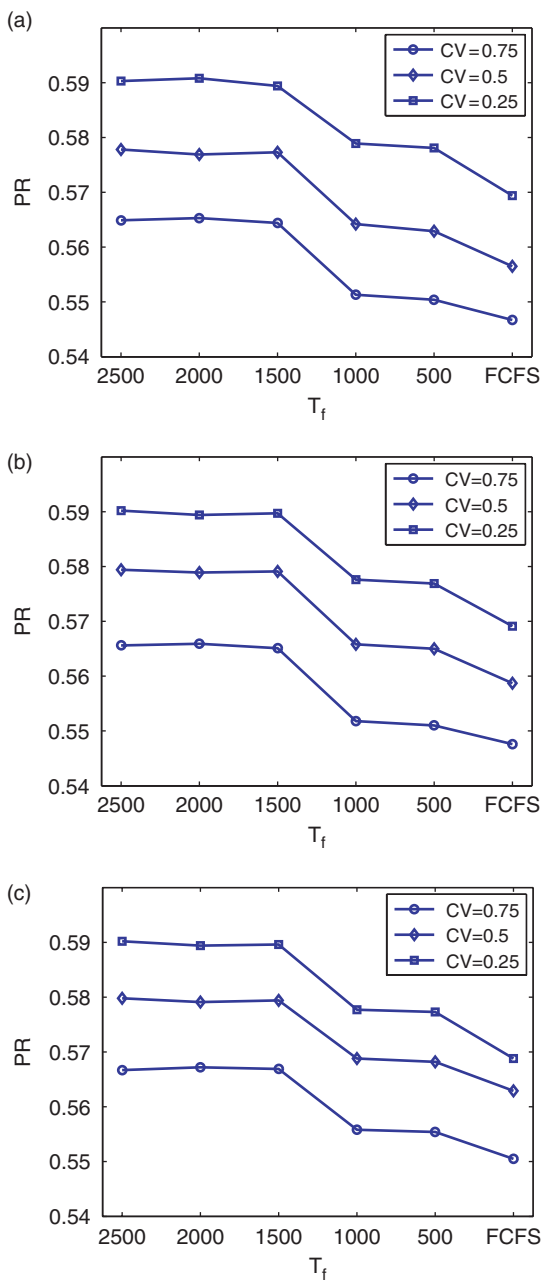


Figure 9. Sensitivity of the FCFS and DBN policies to machine reliability models. (a) Weibull machines. (b) Erlang machines. (c) Log-normal machines.

of variation. The qualitative behaviour does not change with the CV. A smaller CV leads to a higher production rate. $T_f \in [1500, 2500]$ remains a better choice for bottleneck analysis frequency. In addition, no matter what the distribution is, the qualitative behaviour remains the same. Figure 10 indicates that the impact of different distributions

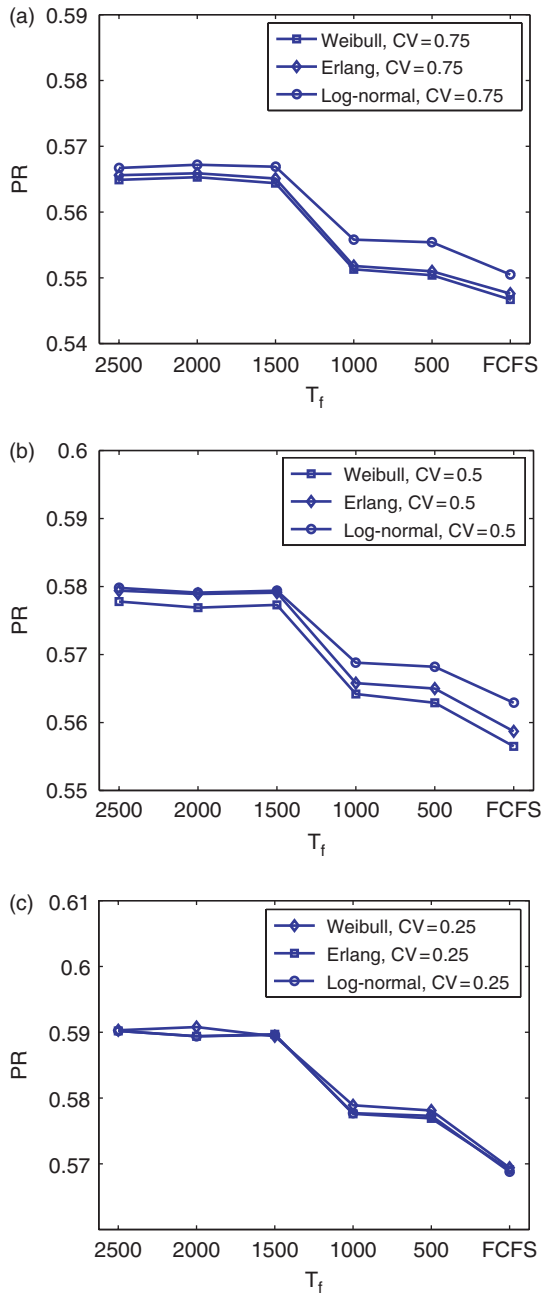


Figure 10. Sensitivity of the FCFS and DBN policies to machine reliability models. (a) $CV = 0.75$. (b) $CV = 0.5$. (c) $CV = 0.25$.

is not large. In most cases, as long as the CVs are the same, all three machine up- and downtime distributions result in similar production rates (except that when T_f is small and for the FCFS policy, the impact is relatively larger than for the other cases, but it remains smaller than for the impacts from CV and T_f).

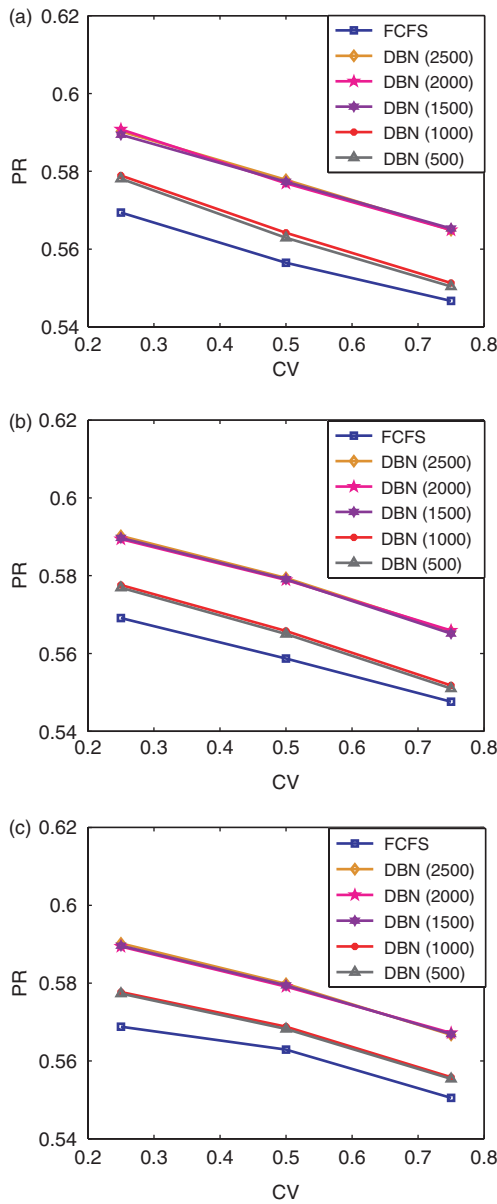


Figure 11. Production rate as a function of the CV. (a) Weibull machines. (b) Erlang machines. (c) Log-normal machines.

As shown by Li and Meerkov (2005a, 2009), the production rate of a serial line without repair resource constraints can be approximated by a linear function of the CV (when $CV \leq 1$). That is, if we know the production rates of the line, $PR_{CV_{\min}}$ and $PR_{CV_{\max}}$, for the smallest and largest CVs, respectively, we can approximate the line production rates with any $CV \in (CV_{\min}, CV_{\max})$ using

$$PR \approx PR_{CV_{\min}} - (PR_{CV_{\min}} - PR_{CV_{\max}})CV.$$

To verify such a property for lines with repair resource constraints, we plot the line production rate as a function of the CV in Figure 11 for Weibull, Erlang and log-normal machines. Linear behaviour is clearly indicated. This implies that, as long as the production rates for the minimum and maximum CVs are known, we can use linear behaviour to approximate the line production rate for any CVs in between without carrying out additional simulations.

In summary, the dynamic bottleneck policy can lead to an improvement in system throughput compared with other dispatching policies. It is important to select an appropriate time period for data analysis and the frequency for bottleneck identification in order to obtain sustainable improvement.

Remark 3: In addition to the three dispatching policies described in Section 2, other policies have also been examined. For example, the buffer-half-full policy was investigated, where priority is assigned to the machine whose surrounding buffer occupancies have the largest deviation from half of their capacity. It is shown that such a policy also leads to consistent improvement compared with the first-come-first-served and constant bottleneck policies, but not as much as for the dynamic bottleneck policy. In addition, the random serve policy implies that the maintenance worker will select the machine randomly when multiple service requests are available. It is shown that such a policy will generate the same results as the first-come-first-served policy. Another example is the priority policy based on machine downtime, i.e. the machine with the longest average downtime has higher priority. However, such a machine may not be the system bottleneck. Moreover, such a policy does not take into account the dynamic nature of the system. Therefore, priority may be assigned to the machine whose improvement would not lead to the largest improvement in system throughput. Numerical experiments also suggest that such a policy would not result in better performance.

6. Conclusions

Maintenance plays an important role in production operations and for continuous improvement. A significant amount of research has been devoted to maintenance scheduling and planning. The modern manufacturing environment requires the maintenance workforce to carry out reactive services quickly and preventive services on time, but with limited resources, to achieve high productivity and quality. Therefore, the dispatching of the maintenance workforce to quickly respond to equipment failures and to conduct preventive activities is very important. In this paper, we present bottleneck-based dispatching policies and investigate their impact on system performance. Specifically, three policies, the first-come-first-served, constant bottleneck and dynamic bottleneck policies, are discussed. By extensive numerical simulations we show that the dynamic bottleneck policy leads to most improvement in system throughput compared with the most commonly used policy, the first-come-first-served policy. We also investigate the appropriate time period for data collection and the frequency of carrying out bottleneck analysis. In addition, a sensitivity study is carried out and it is shown that the results obtained are not sensitive to machine downtime, efficiency, or reliability models.

In future work we will extend the model to integrate preventive maintenance planning so that PM is planned based on the condition of the equipment rather than on

a fixed schedule. The successful development of this work could provide production managers and engineers with quantitative guidance on maintenance scheduling and workforce dispatching on the factory floor.

References

- Andijani, A. and Duffuaa, S., 2002. Critical evaluation of simulation studies in maintenance. *Production Planning and Control*, 13, 336–341.
- Ashire, S., et al., 2000. Workforce-constrained preventive maintenance scheduling using evolution strategies. *Decision Sciences*, 31, 833–859.
- Azadivar, F. and Shu, J.V., 1998. Use of simulation in optimization of maintenance policies. In: *Proceedings of the 1998 winter simulation conference*. NJ: Piscataway.
- Boukas, E.K. and Yang, H., 1996. Optimal control of manufacturing flow and preventive maintenance. *IEEE Transactions on Automatic Control*, 41, 881–885.
- Cavory, G., Dupas, R., and Goncalves, G., 2001. A genetic approach to the scheduling of preventive maintenance tasks on a single product manufacturing production line. *International Journal of Production Economics*, 74, 135–146.
- Chiang, S.-Y., Kuo, C.-T., and Meerkov, S.M., 2001. c-Bottleneck in serial production lines. *Mathematical Problems in Engineering*, 7, 543–578.
- Das, T.K. and Sarkar, S., 1999. Optimal preventive maintenance in a production inventory system. *IIE Transactions*, 31, 537–551.
- Dekker, R. and Scarf, P.A., 1998. On the impact of optimization models in maintenance decision making: the state of the art. *Reliability Engineering Journal*, 30, 15–23.
- Enginarlar, E., Li, J., and Meerkov, S.M., 2005a. How lean can lean buffers be? *IIE Transactions*, 37, 333–342.
- Enginarlar, E., Li, J., and Meerkov, S.M., 2005b. Lean buffering in serial production lines with non-exponential machines. *OR Spectrum*, 27, 195–219.
- Garg, A. and Deshmukh, S.G., 2006. Maintenance management: literature review and directions. *Journal of Quality in Maintenance Engineering*, 12, 205–238.
- Gharbi, A. and Kenne, J.P., 2000. Production and preventive maintenance rates control for a manufacturing system: an experimental design approach. *International Journal of Production Economics*, 65, 275–287.
- Haige, J.W. and Paige, K.N., 2001. *Learning Simul8: the complete guide*. Bethlingham, WA: Plain Vu.
- Inman, R.R., 1999. Empirical evaluation of exponential and independence assumptions in queueing models of manufacturing systems. *Production and Operation Management*, 4, 409–432.
- Jacobs, D.A. and Meerkov, S.M., 1995. A system-theoretic property of serial production lines: improvability. *International Journal of System Science*, 26, 95–137.
- Kuo, C.T., Lim, J.-T., and Meerkov, S.M., 1996. Bottlenecks in serial production lines: a system-theoretic approach. *Mathematical Problems in Engineering*, 2, 233–276.
- Li, J. and Meerkov, S.M., 2005a. Evaluation of throughput in serial production lines with non-exponential machines, In: E.K. Boukas and R. Malhame, eds. *Analysis, control and optimization of complex dynamic systems*. New York: Springer, 55–82.
- Li, J. and Meerkov, S.M., 2005b. On the coefficients of variation of up- and downtime of manufacturing equipment. *Mathematical Problems in Engineering*, 2005, 1–6.
- Li, J. and Meerkov, S.M., 2009. *Production systems engineering*. New York: Springer.
- Marquez, A.C., Gupta, J.N.D., and Heguedas, A.S., 2003. Maintenance policies for a production system with constrained production rate and buffer capacity. *International Journal of Production Research*, 41, 1909–1926.

- Meller, R.D. and Kim, D.S., 1996. The impact of preventive maintenance on system cost and buffer size. *European Journal of Operational Research*, 95, 577–591.
- Mosley, A.A., Teyner, T., and Uzsoy, R.M., 1998. Maintenance scheduling and staffing policies in a wafer fabrication facility. *IEEE Transactions on Semiconductor Manufacturing*, 11, 389–395.
- Parida, A. and Kumar, U., 2006. Maintenance performance measurement (MPM): issues and challenges. *Journal of Quality and Maintenance in Engineering*, 12, 239–251.
- Savsar, M., 1997. Simulation analysis of maintenance policies in just-in-time production systems. *International Journal of Operations & Production Management*, 17, 256–266.
- Scarf, P.A., 1997. On the application of mathematical models in maintenance. *European Journal of Operational Research*, 99, 493–506.
- Schouten, F.C. der Duyn and Vanneste, S.G., 1995. Maintenance optimization of a production system with buffer capacity. *European Journal of Operational Research*, 82, 323–338.
- Wang, H., 2002. A survey of maintenance policies of deteriorating systems. *European Journal of Operational Research*, 139, 469–489.
- Yao, X.D., *et al.*, 2005. Optimal joint preventive maintenance and production policies. *Naval Research Logistics*, 52, 668–681.