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

Bottlenecks in Production Lines with Rework: A Systems Approach

Stephan Biller * **Jingshan Li** ** **Samuel P. Marin** *
Semyon M. Meerkov *** **Liang Zhang** ***

* *GM R&D and Planning, General Motors Corporation 30500 Mound Road, Warren, MI 48090-9055, USA (e-mail: stephan.biller@gm.com, samuel.p.marin@gm.com)*
** *Department of Electrical and Computer Engineering, Center for Manufacturing, University of Kentucky, Lexington, KY 40506, USA (e-mail: jingshan@engr.uky.edu)*
*** *Department of Electrical Engineering and Computer Science University of Michigan, Ann Arbor, MI 48109-2122, USA (e-mail: smm@eecs.umich.edu, liangzh@eecs.umich.edu)*

Abstract: The bottleneck (BN) of a production system is a machine with the strongest effect on the system's throughput. In this paper, a method for BN identification in serial lines with rework and Bernoulli machines is developed. In addition, the paper provides two system-theoretic results: First, it demonstrates that BNs may be shifting not only because of changes in machine and buffer parameters but also due to changes in quality of parts produced. Second, it shows that if the split and the merge machines are not the last and the first, respectively, Bernoulli lines with rework do not observe the property of reversibility, and downstream machines may have a larger effect on the throughput than those upstream.

Keywords: Production activity control; Manufacturing plant control; Bottleneck identification.

1. INTRODUCTION

Serial production lines with rework are used in large volume manufacturing when parts produced may have defects, which can be repaired and "re-worked". The block diagram of such a line is shown in Figure 1. Here the *main line* has M machines and $M-1$ buffers, while the *rework loop* contains M_r machines and M_r+1 buffers. It is assumed that each part (referred to interchangeably as a *job*) at the output of machine m_k is non-defective with probability q and defective with probability $1-q$. The non-defective jobs continue to be processed along the main line, while the defective ones are routed into the rework loop and re-enter the main line through machine m_j. To prevent deadlocks, m_j takes a job from buffer b_{j-1} only if b_{rM_r} is empty.

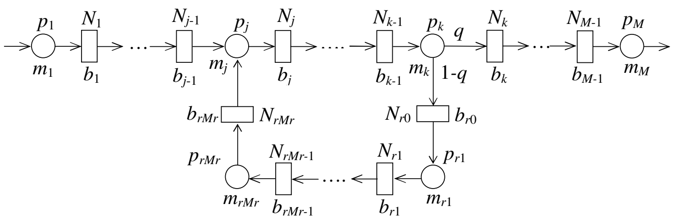


Fig. 1. Bernoulli serial production line with rework

Due to their practical importance, production lines with rework found considerable attention in recent literature (see Li (2004), Borgh et al. (2007) and the references therein). Mostly performance analysis problems have been addressed. Specifically, analytical techniques for evaluating their production rates have been derived under the assumption that the machines obey the *exponential* reliability model. In addition, since production lines with rework can be viewed as a part of the general field of re-entrant lines, issues of deadlock and stability have also been investigated in Kumar (1993). However, bottlenecks in such systems have not been analyzed. The current paper is intended to contribute to this end.

More precisely, the primary goal of this work is to develop a method for bottleneck (BN) identification in serial lines with rework where each machine m_i obeys the *Bernoulli* reliability model, i.e., produces a part in a cycle time with probability p_i and fails to do so with probability $1-p_i$. Such a reliability model is applicable to operations where the unscheduled downtime is primarily due to quality reasons, e.g., painting and assembly operations, see Li and Meerkov (2007). Since methods for performance analysis of Bernoulli lines with rework are not available in the literature and since the BN identification technique developed here requires data on machine blockages and starvations, the secondary goal of this paper is to present a method for performance analysis of Bernoulli serial lines with rework.

The outline of this paper is as follows: Section 2 formulates the model and the problems addressed; Sections 3 and 4 are devoted to performance analysis and BN identification; Section 5 presents the conclusions. Due to space limitations, many details are omitted and can be found in Biller et al. (2007).

2. SYSTEM MODEL AND PROBLEM FORMULATION

2.1 Model

Consider a production line shown in Figure 1. Assume that it operates according to the following assumptions: (i) The main line consists of M machines, $m_1, \ldots, m_M$, and $M-1$ buffers, $b_1, \ldots, b_{M-1}$; the repair part of the rework loop consists of M_r machines $m_{r1}, \ldots, m_{rM_r}$, and M_r+1 buffers, $b_{r0}, b_{r1}, \ldots, b_{rM_r}$. The set of all machines is denoted as m_i, $i \in I_m = \{1, \ldots, M, r1, \ldots, rM_r\}$; the set of all buffers is denoted as b_i, $i \in I_b = \{1, \ldots, M-1, r0, \ldots, rM_r\}$. Machines m_j and m_k are referred to as the *merge* and *split* machines, respectively. (ii) The machines have identical cycle time τ. The time axis is slotted with the slot duration τ. (iii) The machines obey the Bernoulli reliability model, i.e., m_i, $i \in I_m$, being neither blocked nor starved, produces a part during a time slot with probability p_i and fails to do so with probability $1-p_i$. Parameter p_i is referred to as the *efficiency* of m_i. (iv) Each buffer b_i, $i \in I_b$, is characterized by its capacity, N_i, where $1 \leq N_i < \infty$. (v) Machine m_i, $i \neq j$, is starved during a time slot if buffer b_{i-1} is empty at the beginning of the time slot. Machine m_j is starved if both b_{j-1} and b_{rM_r} are empty. Machine m_1 is never starved. (vi) Machine m_i, $i \neq k$, is blocked during a time slot if buffer b_i has N_i parts at the beginning of the time slot and machine m_{i+1} fails to take a part during this time slot. Machine m_k is blocked by the main line if b_k is full and m_{k+1} does not take a part during this time slot; it is blocked by the rework loop if b_{r0} is full and m_{r1} does not take a part during this time slot. Machine m_M is never blocked. (vii) Parts at the output of m_k are non-defective and defective with probability q and $1-q$, respectively. Parameter q is referred to as the *quality buy rate*. Defective and non-defective parts form a sequence of independent random variables. Non-defective and defective parts are routed to buffers b_k and b_{r0}, respectively. (viii) The repaired parts from the rework loop have a higher priority than those from the main line. In other words, m_j does not take a part from b_{j-1} unless b_{rM_r} is empty.

2.2 Problems addressed

2.2.1 Performance analysis problem: Given the machine and buffer parameters and the quality buy rate, evaluate the production rate of the line with rework (PR_{lwr}) and the probabilities of blockages (BL_i^{lwr}) and starvations (ST_i^{lwr}) of each machine in the system. Since m_j has two upstream buffers, b_{j-1} and b_{rM_r}, and m_k has two downstream buffers, b_{k+1} and b_{r0}, their starvations and blockages are denoted as follows:

$$
\begin{aligned}
ST_{j_1}^{lwr} &= P[m_j \text{ is starved by } b_{j-1}],\\
ST_{j_2}^{lwr} &= P[m_j \text{ is starved by } b_{rM_r}],\\
BL_{k_1}^{lwr} &= P[m_j \text{ is blocked by } b_{k+1}],\\
BL_{k_2}^{lwr} &= P[m_j \text{ is blocked by } b_{r0}].
\end{aligned}
$$

A solution to this problem is given in Section 3.

2.2.2 Bottleneck identification problem: The bottleneck machine (BN) of a production line has been defined in Kuo et al. (1996) as the machine that has the largest effect on the system production rate. In the framework of a production line with rework, this definition implies that m_i, $i \in I_m$, is the BN if

$$\frac{\partial PR_{lwr}}{\partial p_i} > \frac{\partial PR_{lwr}}{\partial p_l}, \quad \forall l \neq i. \tag{1}$$

Because this definition is difficult to apply in practice (since the partial derivatives involved cannot be evaluated analytically and are difficult to obtain from factory floor measurements), Kuo et al. (1996) offered an indirect method applicable to open serial lines (i.e., lines without rework). This method, illustrated in Figure 2, consists of calculating analytically or measuring on the factory floor the probabilities of blockages, BL_i, and starvations, ST_i, of all machines in the system and assigning an arrow directed from m_i to m_{i+1} if $BL_i > ST_{i+1}$ and from m_{i+1} to m_i if $BL_i < ST_{i+1}$ (see Figure 2). If there is a single machine with no emanating arrows, it is the BN in the sense of (1). If there are multiple machines with no emanating arrows (as in Figure 2), the one with the largest severity is the *primary* bottleneck (PBN), where the severity of the bottleneck is defined as

$$
\begin{aligned}
&S_i = |ST_{i+1} - BL_i| + |ST_i - BL_{i-1}|, \quad i = 2, \ldots, M-1,\\
&S_1 = |ST_2 - BL_1|, \quad S_M = |ST_M - BL_{M-1}|.
\end{aligned} \tag{2}
$$

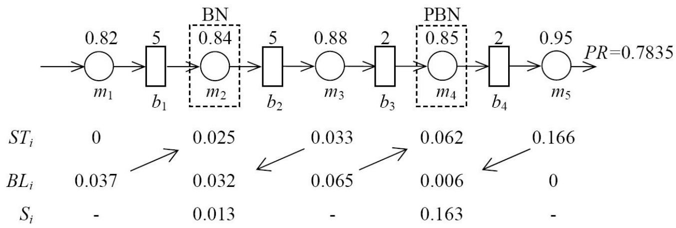


Fig. 2. Bottleneck identification in open serial lines

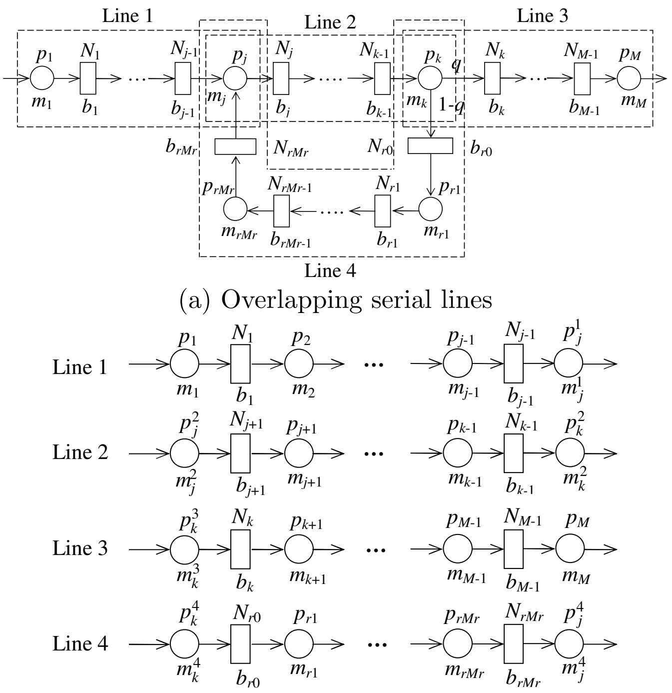
With the rework loop added, this method is not applicable due to the merge and split operations. The main goal of this paper is to extend this method to Bernoulli lines with rework. This is carried out in Section 4.

3. PERFORMANCE ANALYSIS

3.1 Approach

Due to complexity of the Markov chains involved in their description, direct analysis of Bernoulli lines with rework is all but impossible. Therefore, a simplification is necessary. In this paper we use two simplification techniques: overlapping decomposition and recursive aggregation. The method of overlapping decomposition, developed in Li (2004), represents a complex production system as several open serial lines. The recursive aggregation technique, proposed by Jacobs and Meerkov (1995), allows for analytical performance evaluation of open serial lines. Below, these two simplification techniques are briefly reviewed.

3.1.1 Overlapping decomposition: This method is based on representing the line with rework as four *overlapping* serial lines and four *virtual* serial lines shown in Figure 3(a) and (b), respectively. The overlapping lines include the overlapping machines m_j and m_k, each belonging to three lines. The virtual lines do not contain overlapping machines; instead, they include six virtual machines, m_j^1, m_j^2, m_j^4 and m_k^2, m_k^3, m_k^4, efficiencies of which are selected so as to represent the effect of the rest of the system on a particular virtual line. Calculating appropriately the efficiencies, p_j^1, p_j^2, p_j^4 and p_k^2, p_k^3, p_k^4, of the virtual machines, allows one to analyze the performance of the original line with rework. For the exponential model of machine reliability, this has been carried out in Li (2004). For Bernoulli machines, this approach is developed in Subsection 3.2.



(a) Overlapping serial lines

(b) Virtual serial lines

Fig. 3. Overlapping decomposition of serial production lines with rework

3.1.2 Aggregation of Bernoulli serial lines: It has been shown in Jacobs and Meerkov (1995) that the performance measures of a serial line consisting of M Bernoulli machines with efficiencies, $p_1, \dots, p_M$, and $M-1$ buffers with capacities, $N_1, \dots, N_{M-1}$, can be evaluated using the following:

Recursive Procedure 3.1:

$$p_i^b(s+1) = p_i[1 - Q(p_{i+1}^b(s+1), p_i^f(s), N_i)],$$
$$i = 1, \dots, M-1, s = 0, 1, 2, \dots,$$
$$p_i^f(s+1) = p_i[1 - Q(p_{i-1}^f(s+1), p_i^b(s+1), N_{i-1})],$$
$$i = 2, \dots, M, s = 0, 1, 2, \dots,$$

with initial conditions $p_i^f(0) = p_i$, $i = 1, \dots, M$, boundary conditions $p_1^f(s) = p_1$, $p_M^b(s) = p_M$, $s = 0, 1, \dots$, and

$$Q(p_1, p_2, N) = \begin{cases} \dfrac{(1-p_1)[1-\alpha(p_1,p_2,N)]}{1 - \frac{p_1}{p_2}\alpha^N(p_1,p_2)} & \text{if } p_1 \neq p_2, \\ \dfrac{1-p}{N+1-p} & \text{if } p_1 = p_2 = p, \end{cases}$$

$$\alpha(p_1, p_2) = \frac{p_1(1-p_2)}{p_2(1-p_1)}.$$

This procedure results in two sequences $p_2^f(s), \dots, p_M^f(s)$ and $p_1^b(s), \dots, p_{M-1}^b(s)$, $s = 1, 2, \dots$, which are proved to be convergent, and the following limits exist: $p_i^f = \lim_{s\to\infty} p_i^f(s)$, $p_i^b = \lim_{s\to\infty} p_i^b(s)$. In terms of these limits, the performance measures of a Bernoulli line are evaluated as follows:

$$\widehat{PR} = p_1^b = p_M^f = p_i^f[1 - Q(p_{i+1}^b, p_i^f, N_i)]$$
$$= p_{i+1}^b[1 - Q(p_i^f, p_{i+1}^b, N_i)], \quad i = 1, \dots, M-1, \quad (3)$$
$$\widehat{BL}_i = p_i Q(p_{i+1}^b, p_i^f, N_i), \quad i = 1, \dots, M-1, \quad (4)$$
$$\widehat{ST}_i = p_i Q(p_{i-1}^f, p_i^b, N_{i-1}), \quad i = 2, \dots, M. \quad (5)$$

In addition to these performance measures, the overlapping decomposition of Subsection 3.2 requires the probability that b_1 is full while m_2 is either blocked or down and the probability that b_{M-1} is empty; we denote these probabilities as $\widehat{bl}_1$ and $\widehat{st}_M$, respectively. As it follows from (4) and (5), they can be evaluated as

$$\widehat{bl}_1 = \widehat{BL}_1/p_1 = Q(p_2^b, p_1, N_1), \quad (6)$$
$$\widehat{st}_M = \widehat{ST}_M/p_M = Q(p_{M-1}^f, p_M, N_{M-1}). \quad (7)$$

Recursive Procedure 3.1 and the estimates of the performance measures (3)-(7) are used below for analysis of the virtual serial lines of Figure 3(b).

3.2 Recursive procedure for Bernoulli serial lines with rework

Consider the virtual lines of Figure 3(b) and introduce the following notations:

$$PR_l = \text{production rate of virtual line } l,\ l = 1, 2, 3, 4,$$
$$BL_i^l = P[m_i \text{ in virtual line } l \text{ is blocked}],\ l = 1, 2, 3, 4,$$
$$ST_i^l = P[m_i \text{ in virtual line } l \text{ is starved}],\ l = 1, 2, 3, 4,$$
$$bl_j = P[b_j \text{ is full and } m_{j+1} \text{ is either down or blocked}],$$
$$st_{j_1} = P[b_{j-1} \text{ is empty}],\ st_{j_2} = P[b_{rM_r} \text{ is empty}],$$
$$bl_{k_1} = P[b_k \text{ is full and } m_{k+1} \text{ is either down or blocked}],$$
$$bl_{k_2} = P[b_{r0} \text{ is full and } m_{r1} \text{ is either down or blocked}],$$
$$st_k = P[b_{k-1} \text{ is empty}].$$

The estimates of these probabilities, which allow us to evaluate the parameters p_j^1, p_j^2, p_j^4 and p_k^2, p_k^3, p_k^4, can be calculated using a recursive procedure described below.

Recursive Procedure 3.2:

Step 0: Select the initial conditions $\widehat{st}_k(0)$, $\widehat{bl}_j(0)$ and $\widehat{bl}_{k_1}(0)$ randomly and equiprobably from the interval (0,1).
Step 1: Consider virtual line 4 of Figure 3(b) and update the efficiencies of machines m_j^4 and m_k^4 as follows:

$$p_k^4(n+1) = p_k(1-q)[1 - \widehat{st}_k(n)][1 - \widehat{bl}_{k_1}(n)],$$
$$p_j^4(n+1) = p_j[1 - \widehat{bl}_j(n)], \quad n = 0, 1, \dots.$$

Perform Recursive Procedure 3.1 on virtual line 4 and, using (3)-(7), calculate $\widehat{PR}_4(n+1)$, $\widehat{BL}_i^4(n+1)$, $\widehat{ST}_i^4(n+1)$, $\widehat{bl}_{k_2}(n+1)$ and $\widehat{st}_{j_2}(n+1)$.

Step 2: Consider virtual line 3 of Figure 3(b) and update the efficiency of machine m_k^3 as follows:

$$p_k^3(n+1) = p_k q[1 - \widehat{st}_k(n)][1 - \widehat{bl}_{k_2}(n+1)],\ n = 0, 1, \ldots.$$

Perform Recursive Procedure 3.1 on virtual line 3 and, using (3)-(7), calculate $\widehat{PR}_3(n+1)$, $\widehat{BL}_i^3(n+1)$, $\widehat{ST}_i^3(n+1)$ and $\widehat{bl}_{k_1}(n+1)$.

Step 3: Consider virtual line 1 of Figure 3(b) and update the efficiency of machine m_j^1 as follows:

$$p_j^1(n+1) = p_j[1 - \widehat{bl}_j(n)]\widehat{st}_{j_2}(n+1),\ n = 0, 1, \ldots.$$

Perform Recursive Procedure 3.1 on virtual line 1 and, using (3)-(7), calculate $\widehat{PR}_1(n+1)$, $\widehat{BL}_i^1(n+1)$, $\widehat{ST}_i^1(n+1)$ and $\widehat{st}_{j_1}(n+1)$.

Step 4: Consider virtual line 2 of Figure 3(b) and update the efficiencies of machines m_j^2 and m_k^2 as follows:

$$p_j^2(n+1) = p_j[1 - \widehat{st}_{j_1}(n+1)\widehat{st}_{j_2}(n+1)],$$

$$p_k^2(n+1) = p_k[1 - \widehat{bl}_{k_1}(n+1)][1 - \widehat{bl}_{k_2}(n+1)],\ = 0, 1, \ldots.$$

Perform Recursive Procedure 3.1 on virtual line 2 and using (3)-(7), calculate $\widehat{PR}_2(n+1)$, $\widehat{BL}_i^2(n+1)$, $\widehat{ST}_i^2(n+1)$, $\widehat{bl}_j(n+1)$ and $\widehat{st}_k(n+1)$.

Step 5: If the stopping rule

$$\sum_{l=1}^{4} |\widehat{PR}_l(n+1) - \widehat{PR}_l(n)| < \varepsilon, \quad \varepsilon \ll 1, \tag{8}$$

is satisfied, the procedure is terminated; otherwise return to Step 1.

Denote the limits of this procedure as

$$\widehat{PR}_l = \lim_{n\to\infty} \widehat{PR}_l(n), \quad l = 1, 2, 3, 4, \tag{9}$$

$$\widehat{bl}_j = \lim_{n\to\infty} \widehat{bl}_j(n),\ \widehat{st}_{j_1} = \lim_{n\to\infty} \widehat{st}_{j_1}(n),\ \widehat{st}_{j_2} = \lim_{n\to\infty} \widehat{st}_{j_2}(n),$$

$$\widehat{st}_k = \lim_{n\to\infty} \widehat{st}_k(n),\ \widehat{bl}_{k_1} = \lim_{n\to\infty} \widehat{bl}_{k_1}(n),\ \widehat{bl}_{k_2} = \lim_{n\to\infty} \widehat{bl}_{k_2}(n).$$

Unfortunately, the existence of and the convergence to these limits cannot be proved analytically (due to the non-monotonic behavior of $\widehat{bl}_{k_2}(n)$, $n = 0, 1, \ldots$). Therefore, it has been investigated numerically. A total of 5,000,000 lines with rework have been analyzed, and in every case the convergence (with $\varepsilon = 10^{-10}$ in (8)) took place. The details of this investigation are as follows:

Justification: A total of 5,000,000 lines have been generated by selecting j, k, M, M_r, p_i's, N_i's and q randomly and equiprobably from the sets:

$$j \in \{2, 3, 4, 5\},\ k - j \in \{2, 3, 4, 5\}, \tag{10}$$

$$M - k \in \{1, 2, 3, 4\},\ M_r \in \{1, 2, 3, 4\}, \tag{11}$$

$$p_i \in [0.7, 0.95], \quad i = 1, \ldots, M, \tag{12}$$

$$p_i \in [0.1, 0.5], \quad i = r1, \ldots, rM_r, \tag{13}$$

$$N_i \in \{1, 2, 3, 4, 5\}, \quad i \in I_b, \tag{14}$$

$$q \in [0.7, 0.95]. \tag{15}$$

Note that the efficiencies of the machines in the rework loop are selected lower than those of the main line because in practice the capacity of the repair part of the system is typically smaller than that of the main line.

For each of the lines, we ran Recursive Procedure 3.2 and observed the convergence in all cases studied, with the convergence taking place within a second using a standard laptop with a Pentium M 1.60GHz processor and 1.23GB RAM. Thus, we conclude that this procedure can be used for analysis of production lines with rework defined by assumptions (i)-(viii).

Concluding this subsection, we point out the following relationships among the production rates of the virtual lines 1-4:

Theorem 1. The production rates of the virtual lines are related as follows:

$$\widehat{PR}_1 = \widehat{PR}_3, \quad \widehat{PR}_3 = q\widehat{PR}_2, \quad \widehat{PR}_4 = (1-q)\widehat{PR}_2.$$

Proof: See Biller et al. (2007).

3.3 Performance measure estimates and their accuracy

In Section 2, the production rate of the line with rework was denoted as PR_{lwr} and the probabilities of blockages and starvations as BL_i^{lwr} and ST_i^{lwr}. Based on the limits (9) of Recursive Procedure 3.2, their estimates are introduced as follows:

$$\widehat{PR}_{lwr} = \widehat{PR}_3, \tag{16}$$

$$\widehat{ST}_i^{lwr} = \widehat{ST}_i^l,\ \widehat{BL}_i^{lwr} = \widehat{BL}_i^l,\ i \neq j, k,\ l = 1, 2, 3, 4, \tag{17}$$

$$\widehat{ST}_{j_1}^{lwr} = \widehat{st}_{j_1} p_j,\ \widehat{ST}_{j_2}^{lwr} = \widehat{st}_{j_2} p_j,\ \widehat{BL}_j^{lwr} = \widehat{bl}_j p_j, \tag{18}$$

$$\widehat{ST}_k^{lwr} = \widehat{st}_k p_k,\ \widehat{BL}_{k_1}^{lwr} = \widehat{bl}_{k_1} p_k,\ \widehat{BL}_{k_2}^{lwr} = \widehat{bl}_{k_2} p_k. \tag{19}$$

The accuracy of these estimates has been investigated by simulations. For the system parameters $M = 10$, $j = 4$, $k = 7$ and $M_r = 2$, we constructed 100,000 lines with p_i's, N_i's and q's selected randomly and equiprobably from sets (12)-(15). The following metrics were used to evaluate the accuracy of the estimates:

$$\epsilon_{PR} = \frac{|PR_{lwr} - \widehat{PR}_{lwr}|}{PR_{lwr}} \cdot 100\%, \tag{20}$$

$$\epsilon_{ST} = \frac{1}{M + M_r} \sum_{i \in S_1} |ST_i^{lwr} - \widehat{ST}_i^{lwr}|, \tag{21}$$

$$S_1 = \{1, \ldots, j-1, j_1, j_2, j+1, \ldots, rM_r\},$$

$$\epsilon_{BL} = \frac{1}{M + M_r} \sum_{i \in S_2} |BL_i^{lwr} - \widehat{BL}_i^{lwr}|, \tag{22}$$

$$S_2 = \{1, \ldots, k-1, k_1, k_2, k+1, \ldots, rM_r\}.$$

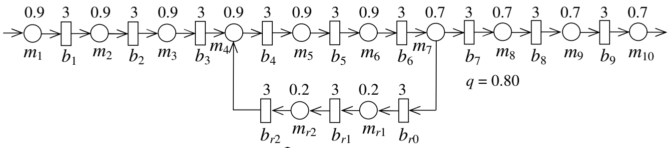
Among the 100,000 lines studied, the average of ϵ_{PR} was 3.97%, with very few extreme cases resulting in ϵ_{PR} up to 20.4%. This accuracy is comparable with that obtained in Li (2004) for the case of exponential machines with similar parameters. The average of ϵ_{ST} and ϵ_{BL} were both less than 0.01. Therefore, we conclude that Recursive Procedure 3.2 provides an effective tool for performance evaluation of serial lines with rework defined by assumptions (i)-(viii).

3.4 System-theoretic properties of lines with rework

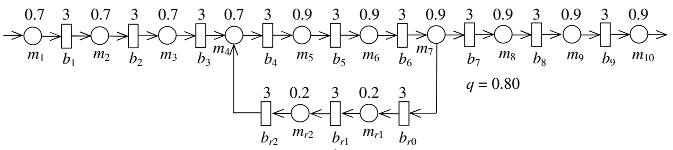
Using Recursive Procedure 3.2, we establish a system-theoretic property described next.

As it is well known, production lines observe the property of reversibility (see Yamazaki and Sakasegawa (1975)): the production rates of a serial line and its reverse (i.e., when the parts flow from the last machine to the first) are the same. This property certainly holds for open and closed Bernoulli lines (see Li and Meerkov (2007)). However, as we show below, reversibility does not hold for Bernoulli lines with rework, if the split and the merge machines are the last and the first, respectively.

Indeed, consider a serial line and its reverse shown in Figure 4(a) and (b). Using Recursive Procedure 3.2, we determine that the production rate of the reverse line is not the same as that of the original one; this conclusion is also supported by simulations (see the data of Figure 4). Thus, reversibility is violated. *The lack of reversibility constitutes a fundamental difference between the usual (i.e., open) Bernoulli lines and those with rework.*



(a) Production line: $\widehat{PR}_{lwr} = 0.4779$, $PR_{lwr} = 0.4657$



(b) Reversed production line: $\widehat{PR}_{lwr} = 0.5330$, $PR_{lwr} = 0.4903$

Fig. 4. Production line with rework and its reverse

In addition, comparing the data of Figure 4 we observe that *placing more efficient machines towards the end of the line results in higher production rate than placing them upstream.* This is also qualitatively different from serial lines with no rework where the position of a machine does not indicate its importance for performance of the system.

4. BOTTLENECK IDENTIFICATION

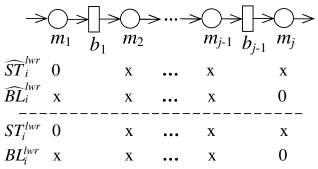
4.1 Approach

The approach to BN identification in serial lines with rework is based on a two-stage procedure.

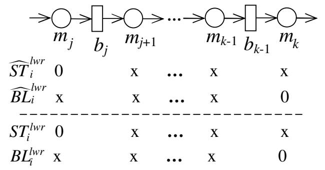
At the first stage, BNs of the four overlapping lines of Figure 3(a) are determined; we refer to them as *local bottlenecks* (LBNs). At the second stage, the overall bottleneck of the line with rework, referred to as the *global bottleneck* (GBN), is identified. We describe below how LBNs can be identified and show that one of them is practically always the GBN.

4.2 Local bottlenecks identification

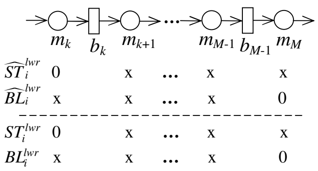
Consider the line with rework of Figure 3(a). Represent its overlapping lines as shown in Figure 5. Assume that these lines are in isolation, i.e., the first machines are not starved and the last ones are not blocked. The probabilities of blockages and starvations of all other machines can be either calculated using Recursive Procedure 3.2 or measured on the factory floor during normal system operation. Based on these data, the BN of each overlapping line can be identified using the arrow method of Subsection 2.2.2.



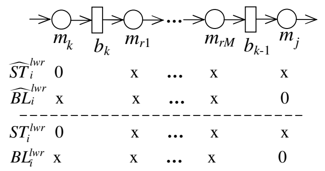
(a) LBN identification of overlapping line 1



(b) LBN identification of overlapping line 2



(c) LBN identification of overlapping line 3



(d) LBN identification of overlapping line 4

Fig. 5. LBNs identification

4.3 Global bottleneck identification

The foundation for GBN identification is based on the following:

Numerical Fact 1. In serial lines with rework defined by assumptions (i)-(viii), the GBN is practically always one of the four LBNs.

Although a justification of this fact (including the quantification of the term "practically always") is given later on (along with other numerical facts formulated below), its application is clear: To identify the GBN, one must test the effect of each LBN on the production rate of the system; the LBN with the largest effect is the GBN. While this process is somewhat involved (due to four, rather than one, BNs to be investigated), it can be facilitated by the following:

Numerical Fact 2. For serial lines with rework defined by assumptions (i)-(viii),

(a) if an overlapping machine is the LBN in three of the overlapping lines, then it is practically always the GBN;

(b) if an overlapping machine is the LBN in only one of the overlapping lines, then it is practically never the GBN.

Numerical Fact 3. In a serial line with rework defined by assumptions (i)-(viii) and with the quality buy rate q^*,

(α) if its GBN is a non-overlapping machine of line 1, then this machine is practically always the GBN for all $q > q^*$;

(β) if its GBN is a non-overlapping machine of line 3, then this machine is practically always the GBN for all $q > q^*$;

(γ) if its GBN is a non-overlapping machine of line 4, then this machine is practically always the GBN for all $q < q^*$;

(δ) if its GBN is a non-overlapping machine of line 2 or line 4, then the GBN is practically always in ether line 2 or line 4 for all $q < q^*$.

Justification: The justification of Numerical Facts 1-3 has been carried out as follows: A total of 100,000 lines have been generated with $M = 10$, $j = 4$, $k = 7$ and $M_r = 2$ and parameters of machines and buffers selected randomly and equiprobably from sets (12)-(15). The results are summarized in Table 1. Based on these data, we conclude that Numerical Facts 1-3 indeed take place.

	Calculation-based	Measurement-based
Numerical Fact 1	97.7%	93.3%
Numerical Fact 2(a)	88.1%	86.9%
Numerical Fact 2(b)	95.3%	94.9%
Numerical Fact 3(α)	91.3%	92.2%
Numerical Fact 3(β)	94.6%	94.0%
Numerical Fact 3(γ)	96.7%	95.4%
Numerical Fact 3(δ)	97.0%	97.2%

Table 1. Accuracy of Numerical Facts 1-3

4.4 Example

Consider the line with rework shown in Figure 6 along with the corresponding overlapping lines 1-4. Their local bottlenecks, identified both by calculations and the measurement-based approaches (using simulations), are also indicated. Since m_4 is the LBN in only one of the

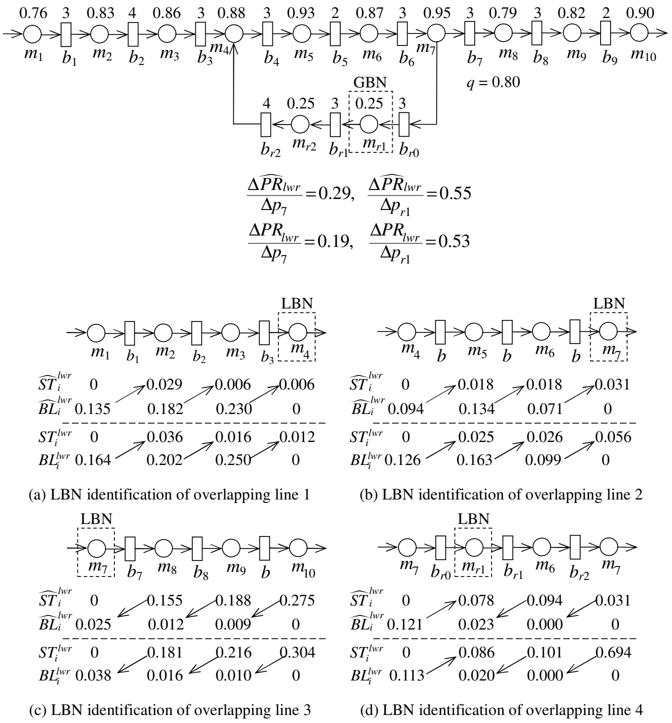


Fig. 6. GBN identification for $q = 0.8$

virtual lines, according to Numerical Fact 2(b) it is not the GBN. Thus, the candidates are m_7 and m_{r1}. Increasing their efficiencies by 0.01, we determine that m_{r1} is the GBN. According to Numerical Fact 3(γ), this machine remains the GBN for all quality buy rate less than 0.8.

Similar analysis for $q = 0.85$, $q = 0.90$ and $q = 0.95$ result in GBNs being machines m_6, m_8 and m_1, respectively. Thus, *the BNs of production lines are shifting not only due to changes in machine and buffer parameters but also due to changes in the quality buy rate.*

5. CONCLUSIONS

The method developed in this paper is an effective tool for BN identification in Bernoulli serial lines with rework. Using this method, we have shown that these lines possess specific system-theoretic properties, which differ from those observed in open lines.

REFERENCES

S. Biller, J. Li, S. P. Marin, S. M. Meerkov, and L. Zhang. Bottlenecks in serial production lines with rework. Control Group Report, CGR-07-06, University of Michigan, August 2007.

D. Borgh, M. Colledani, F. Simone, and T. Tolio. Integrated analysis of production logistics and quality performance in transfer lines with rework. In *Proceedings of 6th International Conference on the Analysis of Manufacturing Systems*, pages 15–20, 2007.

D. Jacobs and S. M. Meerkov. A system-theoretic property of serial production lines - improvability. *International Journal of Systems Science*, 26(5):755–785, 1995.

P. R. Kumar. Re-entrant lines. *Queueing Systems, Theory and Applications*, 13(1), 1993.

C.-T. Kuo, J.-T. Lim, and S. M. Meerkov. Bottlenecks in serial production lines: A system-theoretic approach. *Mathematical Problems in Engineering*, 2(3):233–276, 1996.

J. Li. Performance analysis of production systems with rework loops. *IIE Transactions*, 36(8):755–765, 2004.

J. Li and S. M. Meerkov. *Production Systems Engineering.* Preliminary Edition, WingSpan Press, 2007.

G. Yamazaki and H. Sakasegawa. Property of duality in tandem queueing systems. *Annals of the Institute of Statistical Mathematics*, 27:201–212, 1975.