REWORK LOOP AND QUALITY INFORMATION FEEDBACK

by
Ömer Ferhat Hançer
B.S., Industrial Engineering, Yildiz Technical University, 2004

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Industrial Engineering
Boğaziçi University
2008

Dedicated to
my parents
and
my primary school teacher
Meral Yılmaz

ACKNOWLEDGEMENTS

First of all, I would like to express my thankfulness to my thesis supervisor Assist. Prof. Aybek Korugan for his invaluable guidance and help through the thesis. This study would not be completed without his tolearance, patience as well as friendly behaviour. I am also grateful to Assoc. Prof. Ali Tamer Ünal for both his informative comments on the thesis and course IE 542 which changes my perspective. Next, I would like to sincerely thank Prof. Dr. Barış Tan whose comments and suggestions I had benefited very much.

I wish to thank my family for the support that they have given me. I am forever indebted to my parents, Sevgi Hançer and Bilal Hançer. I would like to thank my uncle and aunt, Kadir Hançer and Serpil Hançer, for their affection and unending love throughout my life. I also would like to thank all my cousins for their encouragement.

Furthermore, special thanks to my friends, Cihan Açıkkollu, Kemal Dingeç and Özge Çadırcı because of their support. None of this work would have been possible without them.

In the end, I will like to mention the dear beautiful Boğaziçi University. I am very happy to be a member of it. Special thanks to TUBİTAK which supported me financially during my master education.

ABSTRACT

REWORK LOOPS AND QUALITY INFORMATION FEEDBACK

High population and limited amount of resources increase competition, and in order to survive in this competition we must use these resources wisely. Today's manufacturers are conscious of this fact and they find efficient production methods which minimize the costs and reduce the scraps as well as provide the quality level that satisfy customer needs. In production plants, machines are not one-hundred percent reliable and they may sometimes produce defective parts. *Rework*, i.e. the transformation of products that do not meet the desired specifications into products that do, is one of the efficient methods that reduces the amount of scraps. In the production systems with rework loop, after the bad parts are detected and repaired by the rework line, they are sent back to the main transfer line. In order to reduce the number of defective parts, some inspection stations are located in the production systems. When a defective part is detected at inspection stations, the machine producing the bad parts is stopped so that it will not produce more defective parts. This is called *Quality Information Feedback*.

In this thesis, we present Markov models for the approximate solution of the production systems with both single rework line and multiple rework lines. We use overlapping decomposition approach that offered in Li (2004) to approximate the throughput rate of the production systems with rework loops. The idea of overlapping decomposition technique is to decompose the system into serial transfer lines. Our model is different from Li in that, we use decomposition technique instead of aggregation procedure while evaluating the serial transfer lines and we formulate the rework rate with respect to yields of the machines that have quality failures. The accuracy of the method is validated by simulation experiments. In this work, we also seek the answer to a fundamental question: Although the bad parts are repaired in a separate line (rework line), should we stop the machines producing bad parts? In other words, we analyze the effect of quality information feedback on the rework production systems in this thesis by using the models that are developed by Kim and Gershwin, (2005).

ÖZET

TEKRAR-ÜRETİM HATTI VE KALİTE KONTROL GERİBİLDİRİMİ

Nüfus yoğunluğunun yüksek olması ve kaynakların kısıtlı olması rekabeti arttırmaktadır ve bu rekabette hayatta kalabilmemiz için mevcut kaynaklarımızı akıllıca kullanmamız gerekmektedir. Bugünün üreticileri bu gerçeğin bilincindedirler ve maliyetleri aza indirgeyecek, bozuk parça sayısını azaltacak ve aynı zamanda müşterinin beklediği kalitede ürün üretecek etkili imalat metotları geliştirmektedirler. Üretim tesislerinde. makinalar yüzde yüz güvenilir değildir ve sık sık kalitesiz parça üretebilirler. Tekrar-üretim, gerekli nitelikleri karşılamayan ürünlerin tekrardan işlenip hatalı olan kısımlarının düzeltilmesidir. Tekrar-üretim tekniği, ıskarta sayısını azaltan etkili bir metotdur. Tekrar-üretim hatlı imalat sistemlerinde, kalitesiz parça fark edilip tekrar-üretim hattında tamir edildikten sonra, tekrar ana imalat sistemine gönderilir. Kalitesiz parça sayısını azaltmak için imalat sistemlerine ayrıca muayene istasyonları eklenmektedir. Bozuk parça muayene istasyonunda fark edildiğinde, bozuk parçayı üreten makina kalitesiz üretime daha fazla devam etmemesi için durdurulur. Bu Kalite Kontrol Geribildirimi olarak adlandırılır.

Bu çalışmada, bir veya birden fazla tekrar-üretim hattına sahip olan imalat sistemlerini analiz eden bir Markov modeli sunuyoruz. Bu sistemleri analiz etmek için Li tarafından sunulmuş küçük sistemlere bölme tekniğini kullanacağız. Yalnız bizim çözüm tekniğimizin Li’ nin sunduğu teknikten ayrıldığı bazı noktalar var: Biz kalitesiz parça bulma olasılığını makinaların güvenilirliklerine göre belirleyeceğiz. Bu çalışmada ayrıca şu sorunun cevabını arayacağız: Eğer kalitesiz parçalar ayrı bir hatta düzeltilip tekrar işleniyorsa, kalitesiz parçayı üreten makinayı genede durdurmalı mıyız? Bir başka deyişle, kalite kontrol geribildiriminin, tekrar-üretim hatları üzerindeki etksini inceleyeceğiz.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS /ABBREVIATIONS

B_i	Buffer after machine i
b_j	Probability that machine M_k is starved
b_{j1}	Probability that machine M_{j1} is blocked by B_{j1}
b_{j2}	Probability that machine M_{j2} is blocked by B_{j2}
b_{j3}	Probability that machine M_{j3} is blocked by B_{j3}
b_{k1}	Probability that machine M_k is blocked by B_k
b_{k2}:	Probability that machine M_k is blocked by B_{MT}
b_{k1_1}	Probability that machine M_{k1} is blocked by B_{k1}
b_{k1_2}	Probability that machine M_{k1} is blocked by B_{MT}
b_{k2_1}	Probability that machine M_{k2} is blocked by B_{k2}
b_{k3_1}	Probability that machine M_{k3} is blocked by B_{k3}
b_{k2_2}	Probability that machine M_{k2} is blocked by $B_{MT+MR1+1}$
b_{k3_2}	Probability that machine M_{k3} is blocked by $B_{MT+MR1+MR2+2}$
M_i	Machine i
M_j	Machine in front of which (input) main line and rework line join, viz. the merge machine
M_{j1}	The first merge machine in the production system
M_{j2}	The second merge machine in the production system
M_{j3}	The third merge machine in the production system
M_k	Machine after which the line splits into (output) main line and rework line, viz. the split machine
M_{k1}	The first split machine in the production system
M_{k2}	The second split machine in the production system
M_{k3}	The third split machine in the production system
MR	Number of machines in the rework loop
$MR1$	The number of machines in the first rework line

$MR2$	The number of machine in the second rework line
$MR3$	The number of machine in the third rework line
MT	Number of machines in the main line, i.e., the line without rework loop
$PT(i)$	Total throughput rate of *Line i*, $i = 1,...,10$
s_{j1}	Probability that machine M_j is starved by B_{j-1}
s_{j2}	Probability that machine M_j is starved by B_{MT+MR}
s_{j1_1}	Probability that machine M_{j1} is starved by B_{j1-1}
s_{j1_2}	Probability that machine M_{j1} is starved by B_{MT+MR1}
s_{j2_1}	Probability that machine M_{j2} is starved by B_{j2-1}
s_{j2_2}	Probability that machine M_{j2} is starved by $B_{MT+MR1+MR2+1}$
s_{j3_1}	Probability that machine M_{j3} is starved by B_{j3-1}
s_{j3_2}	Probability that machine M_{j3} is starved by $B_{MT+MR1+MR2+MR3+2}$
s_{k1}	Probability that machine M_{k1} is starved by B_{k1-1}
s_{k2}	Probability that machine M_{k2} is starved by B_{k2-1}
s_{k3}	Probability that machine M_{k3} is starved by B_{k3-1}.
Y_i	Yield of the Line i, $i = 1,...,6$
α	Rework rate, i.e. the probability that a part is defective and requires rework
α_1	The probability that a part will be sent to first rework line by M_{k1}
α_2	The probability that a part will be sent to second rework line by M_{k2}
α_3	The probability that a part will be sent to third rework line by M_{k3}
2M1B	Two-Machine-One-Buffer production system
QIF	Quality Information Feedback

1. INTRODUCTION

If there were no uncertainties in our lives, we would lead a perfectLY-planned life. In manufacturing systems, we all have to fight with deviations in our plans due to these uncertainties. Production lines are sets of machines arranged to produce a finished product or a component of a product. These machines are typically unreliable and randomly break down, which lead to unscheduled downtime and loss of production capacity. Failures of a machine affect all other machines in the system, causing blockage of upstream machines and starvation of downstream machines. To minimize these perturbations and to decrease the negative effects of blockage and starvation, buffers are used between the machines in the line. Here, the problem of modeling production systems with finite buffer arises.

Although we are not able to know the exact time when machines will breakdown or how much time it will take to repair a down machine, we can find some average performance parameters. These parameters are throughput rate, i.e. the number of parts produced by the production system per unit of time (production rate), average in-process inventory, and machines' blocking or starvation probabilities. By using these performance parameters, we can make a good production plan that aims to minimize variations of delivery dates promised to customers.

One of the objectives of manufacturers is to increase the throughput rate while maintaining the quality and minimizing the cost. So, customer satisfaction is improved by providing their orders in time and of the quality they expect. To increase the production rate while maintaining the same quality and keeping the cost at a low level is a challenging problem. That motivates the manufacturers to explore new methods in production systems such as *Quality Information Feedback* and *Rework Loops*.

A fraction of the parts processed at some station in the line may be scrapped or reworked to meet product quality requirements. In the *production systems with rework loops*, defective parts are repaired and sent back to the main line for re-processing. The use of rework loops can significantly increase the system yield and reduce scrap, cost, etc. In a rework loop system, there are two phenomena concerning the flow of material: Split and Merge operations. In the split operation, the split machine has multiple alternative immediate successors. After processing a part, the split machine sends this part to one of its immediate successors. In merge operations, both the raw parts and repaired parts are transferred into the merge machine.

Different inspection policies can be performed to improve the quality and the effective production rate. In a transfer line, inspection stations are sometimes designed to perform multiple inspections at the end of the line. When a bad part is detected, the machine that produced this part is informed of this condition and stopped for inspection and/or repair. This is called *quality information feedback.*

The purpose of this thesis is to investigate the production systems with rework loops and the effects of quality information feedback on these systems. We investigate the question of whether the quality information feedback gives us a better throughput rate in production systems with or without rework loops.

The rest of the thesis is organized as follows. In chapter 2, relevant works in manufacturing systems engineering and quality information is reviewed. In the following chapter, the objectives of the thesis are stated in detail. In chapter 4, the problem of interest is defined and solution techniques of production systems with single rework loop with and without *quality information feedback* are presented. Also in this chapter, an evaluation procedure for production systems with multiple loops is proposed. In chapter 5, numerical results are presented and the results are analyzed. Finally, in chapter 6, conclusions are drawn.

2. LITERATURE REVIEW

2.1. Manufacturing Systems Engineering Literature

A great deal of literature has been devoted to the modeling and analysis of transfer and production lines since the early 1950's as these types of production systems are widely encountered in industry. A comprehensive survey by Dallery and Gershwin (1992) provides extensive and elaborate reviews up to that time in this area. Current textbooks covering topics in this field include Viswanadham and Narahari (1992), Buzacott and Shanthikumar (1993), Askin and Standridge (1993), Papadopoulus *et.al.* (1993), Altiok (1996), Helber (1997), as well as Gershwin (2002) which gives a detailed introduction on how to model and analyze transfer lines.

Manufacturing systems engineering explores some important system problems in manufacturing. These are the problems that arise when several resources are used together to manufacture products. For instance, if a part must pass through two machines before it is completed, and one of those machines is out of order then the other machine cannot finish its operation. As a result, some capacity is lost because a perfectly good machine is forced to wait. This can be prevented (up to a point) if some parts have been stored for the operational machine to work on, and there is space to put the pieces it completes while the other is down. In designing such a system one must ask, how much space should be allocated for this purpose, and how much material storage (in-process inventory) should be allowed for this purpose.

2.1.1. Serial Transfer Lines

Transfer lines are the simplest of all alternative topologies for manufacturing systems engineering (Figure 2.1). In Figure 2.1, the squares represent the machines and the circles are the buffers. The usual assumption associated with capacity analysis of transfer lines is that there are always raw parts available at the input and that there are always empty spaces to accommodate the finished parts at the output (i.e., the first machine is never starved and the last machine is never blocked).

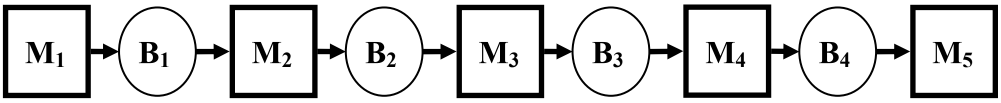

Figure 2.1. Five – Machine Transfer Line

Analysis of production lines is complex because unreliable machines and their effects on the whole production line. Although simulation is the most widely used tool in industry, analytical methods provide an alternative (and complementary) approach for performance evaluation of manufacturing flow systems as simulation is usually time-consuming.

Performance analysis of the transfer lines has begun with the simplest model which is the two – machine – one – buffer (2M1B) system, i.e. the line consisting of two machines separated by one finite buffer. Exact solutions of 2M1B were proposed by Gershwin and Schick (1980) in the case of the continuous flow model, and Gershwin and Berman (1981) for the discrete-state and discrete line Markovian model. The major advantage of the continuous flow model over the deterministic type model is that it applies to any production line, whereas the deterministic model is restricted to homogeneous lines, i.e., production lines in which all machines have the same processing time.

Several extensions of these basic works have then been developed. In the literature, two types of serial lines are considered; synchronous lines (homogeneous lines) where all machines have the same service rates and asynchronous lines (non- homogeneous lines) where machines in the line have different service rates. Two approximation techniques have been offered to evaluate these types of serial lines; decomposition and aggregation.

The *decomposition method* is originally proposed by Gershwin (1987) in the context of the synchronous model. Then, using an iterative algorithm proposed by Dallery *et al.* (1988) (*DDX algorithm*), the robustness of the method is improved. The decomposition method and the DDX algorithm were then adapted to the continuous flow model of homogenous lines by Dallery *et al.* (1989).

For the analysis of *asynchronous lines* (*non- homogeneous lines*), two approaches have been proposed: *A direct approach* and *a two-step approach*. The principle of the direct approach is to extend the decomposition method to non-homogenous models. Different variants of the decomposition method have been proposed for the continuous flow model of non-homogenous lines by Alvarez *et al.* (1991), Suri and Fu (1994), Burman (1995), and Burman and Gershwin (1997). An excellent illustration of the usefulness of analytical methods for the evaluation of transfer lines can be found in the case study reported in Burman *et al.* (1998). The method proposed by Burman (1995) has the advantage of always converging. The method provide a good approximation of the original asynchronous model as long as the average times to failure are significantly larger than the processing times, which is usually the case in production systems.

Le Bihan (1998) shows that Burman's (1995) method can significantly be simplified. This alternative approach to the analysis of non-homogenous lines consists of two steps. In the first step, the non-homogeneous line is transformed into an approximately equivalent homogeneous line. The resulting homogeneous line is then analyzed using the decomposition method for homogeneous lines. Another extension pertains to the improvement of the accuracy of decomposition methods in situations where the original decomposition method may not provide accurate results. Such a situation is encountered when the reliability parameters (mean times to failure and mean times to repair) of different machines have different orders of magnitude, which can be the case in real production lines. Such improvements of the original decomposition method are based on the replacement of the first-moment approximation of the repair time of the equivalent machines in the decomposition by a two-moment approximation (Dallery and Le Bihan, 1999) or a three-moment approximation (Le Bihan and Dallery, 1998).

The aggregation solution technique is offered by Meerkov and Top (1990) for the synchronous lines (homogeneous lines) with Bernoulli machine failures, and Jacobs and Meekov (1995) for the asynchronous lines, respectively.

2.1.2. Complex Systems

In practice, more complex production systems exist and need accurate analysis. Among the studies related to manufacturing systems engineering, exact analysis can be performed only for two-machine systems. Using two-machine line results, various aggregation and decomposition methods have been proposed to approximate the system performance measures for longer lines. By extending the results of serial lines, *assembly/disassembly lines, parallel lines, rework lines etc*., have been also studied.

Extensions of the decomposition methods to tree-structured assembly/disassembly flow systems have been proposed by Gershwin (1991) in the context of the synchronous model, by Di Mascolo et al. (1991) in the context of the continuous flow model of homogeneous lines, and by Gershwin and Burman (1998) in the context of the continuous flow model of non-homogeneous lines. An extension of the decomposition method has been proposed in the context of the continuous flow model of closed-loop production lines by Frein et al. (1996) for homogenous lines and by Patchong (1997) for non-homogenous lines.

Meerkov and Lim (1993), Dallery *et al.* (1996), and Gershwin *et al*. (2003) have studied a closed loop serial production line with constant number of carriers, where parts are loaded and attached on the pallets at the first machine to undergo all the operations. Meerkov and Lim (1993) analyze an asymptotically reliable two-machine two-buffer closed serial line. The closed loop line is reduced to an open production line where the effective buffer capacity depends on the relationship between the actual buffer capacity and the number of pallets. Dallery, *et al.* (1996) present a decomposition approach to approximate the homogeneous closed-loop system's production rate. They investigate the optimal number of carriers, which maximize system performance. Gershwin *et al.* (2003) offer an approximate analytical method for evaluating a three-machine three-buffer closed loop system where machines can fail in more than one mode. In their study, machines have deterministic processing times and geometrically distributed probabilities of failure and repair. The algorithm can also be applied to both small and large loops and takes into account the correlation between the numbers of pallets in the buffers.

Patchong and Willaeys (2001) focus on flow lines composed of multiple parallel-machine stages. The system is similar to a classical flow line, the only difference being that a given stage may consist of parallel machines. A method for modeling and analyzing this type of flow lines is presented. The algorithm replaces each parallel-machine stage by a single equivalent machine in order to obtain a classically-structured flow line with machines in series.

Helber (1997) presents a Markov process model and an approximate decomposition technique for a discrete material transfer line, including a rework loop, with limited buffer capacity by introducing two additional phenomena: Split and Merge. Dallery (1999) extends this work for the continuous flow systems. He solves a split and merge system by transforming it into a disassembly/assembly system under the assumption that failures and repairs of different machines have to occur exactly at the same instant in both (the original and the transformed lines) models.

Li (2004) introduces another approximation technique for analyzing production systems with rework loops. In that research, Li uses an aggregation technique instead of decomposition, and does not take into account operation dependent failures, instead he uses time dependent failures. The solution technique offered by Li is denominated as *Overlapping Decomposition Technique* and Li (2005) extends this methodology to solve various complex production systems such as split-merge systems and parallel lines.

2.2. Quality of Production Models

Modeling production quality has been studied in the literature for the last two decades since it has been recognized as a key factor affecting the competitiveness of companies. Many studies have emphasized the importance of quality.

In the quality of production models literature, two extreme kinds of quality failures based on the characteristics of variations that cause the failures are mentioned; *bernoulli* or *common* or *random quality failures* and *persistent quality failures* or *assignable* or *special*.

Bernoulli quality failures are due to common cause variations. Such failures occur often when an operation is sensitive to external perturbations like defects in raw material or when the operation uses a new technology that is difficult to control. Since no permanent changes have occurred in the machine, the occurrence of a bad part implies nothing about the quality of future parts. In this case, if bad parts are destined to be scrapped, it is useful to catch them as soon as possible because the longer it takes them to be scrapped; the more they consume the capacity of downstream machines. However, it is unnecessary to stop machine that has produced a bad part due to this kind of a failure.

Persistent quality failures are due to assignable cause variations. These kinds of quality failures only happen after a change occurs in the machine or raw material. In that case, once a bad part is produced, all subsequent parts will be bad until the machine is repaired. If the type of quality failure is of Bernoulli type, then the optimal policy is not to stop the machine. On the other hand, if the quality failure is persistent, then the machine should be stopped for repair. For this kind of quality failure, there is no inherent measure of yield, because the fractions of parts that are good and bad depend on how soon bad parts are detected and how quickly the machine is stopped for repair.

In the field of *Statistical Quality Control* (SQC) applied to production systems, Montgomery (1991) contributed in the diffusion of statistical process control theory and Raz (1986) dealt with the problem of the optimal allocation of inspection stations in multistage production lines. Colledani and Tolio (2005) proposed an approximate method for the analysis of production lines, in which SQC techniques are applied, which takes scrap and rework policies into account.

Only few papers consider the intersection between quality control and manufacturing systems modeling. The productivity and quality have been studied extensively, but there is a lack of research in their intersection. Kim and Gershwin (2004) look at the interrelation of quality and productivity. They develop a new Markov process model for machines with both quality and operational failures. They present analytical models, solution techniques, performance evaluations, and validation of two-machine systems as well as longer transfer lines. Here, they propose a three-state machine model where a machine produces good parts in “State 1” and produces bad parts due to a quality failure in “State −1”. When the machine is under repair, i.e “State 0”, an operator can not tell whether the machine is down due to a quality failure or an operational failure. Therefore, whenever a machine is under repair, the operator fixes the machine completely so that the machine goes back to "State 1".

Poffe and Gershwin (2005) develop a 2M1B model in which the first machine has both operational and quality failures and the second machine has only operational failures. Poffe and Gershwin (2005) set the number of states of the first machine to five, therefore, when the first machine is under repair, it can be distinguished whether it is due to a quality failure or operational failure.

Chiang (2005) develops a procedure for the analysis of the production systems with quality control devices. He analyzes the serial lines in which quality control devices are integrated after each machines and assumes that both the machines and the devices can fail following Bernoulli distributions. Chiang first solves the 2M1B system, and then he approximates longer lines by using aggregation technique.

In Toyota Production System, operators are equipped with means of stopping the production process whenever they encounter a quality problem. TPS advocates argue that this prevents the waste that would result from producing a series of defective items. So it is a means to improve quality and increase productivity at the same time. Li and Blumenfeld (2006) develop analytical models to calculate the performance of a serial transfer line featuring Andon. Andon, derived from the Japanese word for paper lantern, is a term for a visual control system using an electric light board (or other signal device) hung in a factory, so that a worker can call for help and stops the line when a defect is discovered. They also investigate conditions under which Andon should be introduced and implemented.

On the other hand, quality failures are often of the kind where the quality of each part is independent of the others. Thus, there is no reason to stop a machine that has produced a bad part because there is no reason to believe that stopping it will reduce the number of bad parts in the future. In this case, stopping the operation does not influence quality but it reduces productivity. Kim and Gershwin (2004) and Poffe and Gershwin (2005) investigate the effects of *quality information feedback* stopping policies into the transfer lines in which both operational and quality failure occur.

Solution techniques to analyze assembly/disassembly lines, parallel lines, closed-loop systems, and rework lines have been developed for production systems without quality failures. However, to the best of our knowledge, there exists no study that investigates complex production systems where machines have quality failures. In this thesis, we investigate this problem and offer a solution technique for the production systems with both single and multiple rework lines where machines have quality failures. We also explore the effect of quality information feedback on the production systems with rework loops.

3. OBJECTIVE OF THE THESIS

To gain competitive advantage in the business world, companies need to continually update their business operations and seek effective methodologies such as six sigma, lean manufacturing and TQM. Enterprises should reduce their costs and increase agility simultaneously by optimizing their resources. While reducing the costs, enterprises know that their products have to meet the desired quality level in order to provide customer satisfaction.

High quality level is an important requirement in business competitiveness. Kim and Gershwin (2005) are the first researchers that proposed an analytical method to evaluate the performance of a production line in which the machines have quality failures. They showed the relation between the capacity of the buffer and the effective throughput rate of the production system.

One of the objectives of the thesis is to observe the effect of quality information feedback on the production systems with rework loop. In the case where there is no quality information feedback in the rework loop, we only perform rework on products with bad quality. Here, machines with persistent quality failures are repaired after they have an operational failure or after the operator finds out the quality failure. In the case where there is quality information feedback in the rework loop system, when machines make quality failures and the downstream machines detect persistent quality failures, the machines producing defective parts are restored to their original state of good production. Another objective of the thesis is to find an efficient algorithm that analyze the production systems with multiple rework lines and to compare production systems with single rework line to production systems with multiple rework lines.

This thesis is an extended work of Kim and Gershwin (2005) and Li (2004). Kim and Gershwin (2005) evaluate the performance of *serial lines* where machines have quality failures. In this work, our aims are:

- To find an effective solution algorithm of the production system with quality failures where the bad parts are reworked by using the overlapping solution technique offered by Li (2004),

- To find the throughput rate of this rework loop system when we use *Quality Information Feedback* stopping policy
- To compare the serial transfer lines with QIF to rework loop systems. In order words, we aim to observe the effect of rework lines on throughput rate of the production systems with quality failures,
- To compare the production systems with single rework line to production systems with multiple rework lines (multiple loops).

4. PROBLEM DEFINITION AND MODEL

In manufacturing systems, parts or subassemblies may not always be of perfect quality. The production processes are unreliable and the cost of energy and materials are high as well as the input materials are limited. Therefore, in many production plants, rework loops are often included for the repair and multiple processing of jobs. In the rework loops, defective parts are repaired and sent back to the main production line for reprocessing.

We consider a saturated production system with rework loop in this work, viz. there are inexhaustible supplies of work pieces for the first machine and an unlimited storage area for the last machine. In this production system with rework loop, some machines may have operational and persistent type of quality failures (once a bad part is produced, all subsequent parts will be of bad quality until the machine is repaired) and these failures are operation dependent. That is, they occur only when the machine is processing a part. All the failures and repairs are independent which means that each machine works on a different feature. For example, two consecutive machines may be drilling two different holes. We do not consider cases where both machines work on the same hole, in which the first machine does a roughing operation and the second does a finishing operation. This allows us to assume that the quality failures of the machines are independent. Also there are unlimited repair personnel.

We consider a continuous model. Continuous models treat material traveling through the production system as if it were a continuous fluid. Continuous models assume constant service rates of the machines. These models are useful approximations to discrete material systems as long as service rates are relatively small in relation to failure and repair times and buffers are of a reasonable size.

4.1. The Production Systems with Rework Loop

The use of rework loops can significantly increase the system yield. In a rework loop system, there are two phenomena concerning the flow of material: Split and Merge operations. In split operation, the split machine has two alternative immediate successors. After processing a part, split machine sends this part either further for new production or to rework. In merge operation, the merge machine has two alternative upstream buffers and as long as there are parts in one of these upstream buffers, merge machine operates if it is not down.

The production system studied in this paper is shown in Figure 4.1. For convenience, the following notations are used throughout this work:

MT : Number of machines in the main line, i.e., the line without rework loop.
MR : Number of machines in the rework loop.
α : Rework rate, i.e. the probability that a part is defective and requires rework ($0 < \alpha < 1$).
M_k : Machine after which the line splits into (output) main line and rework line, viz. the split machine.
M_j : Machine in front of which (input) main line and rework line join, viz. the merge machine.
M_i : Machine i, $i = 1, 2, \ldots, MT + MR$.
B_i : Buffer after machine i, $i = 1, 2, \ldots, MT + MR$. (Li, 2004)

In Figure 4.1, the rectangles represent the machines and the circles are the buffers. The system consists of a main production line (machines $M_1, ..., M_{MT}$, buffers B_1, ..., B_{MT-1}) and a rework line (machines M_{MR+1}, ..., M_{MT+MR}, buffers $B_{MT}, ..., B_{MT+MR}$). In the main line, MT machines are arranged serially and $MT-1$ buffers separating each consecutive pair of machines. In the rework line, MR machines are also arranged serially, however, $MR + 1$ buffers separate each consecutive pair of machines, including buffers B_{MT} and B_{MT+MR} separating machine pairs (M_k, M_{MT+1}) and (M_{MT+MR}, M_j), respectively.

Machine M_k and M_j $(MT > k > j > 1)$ are the starting and ending points of the rework loop.

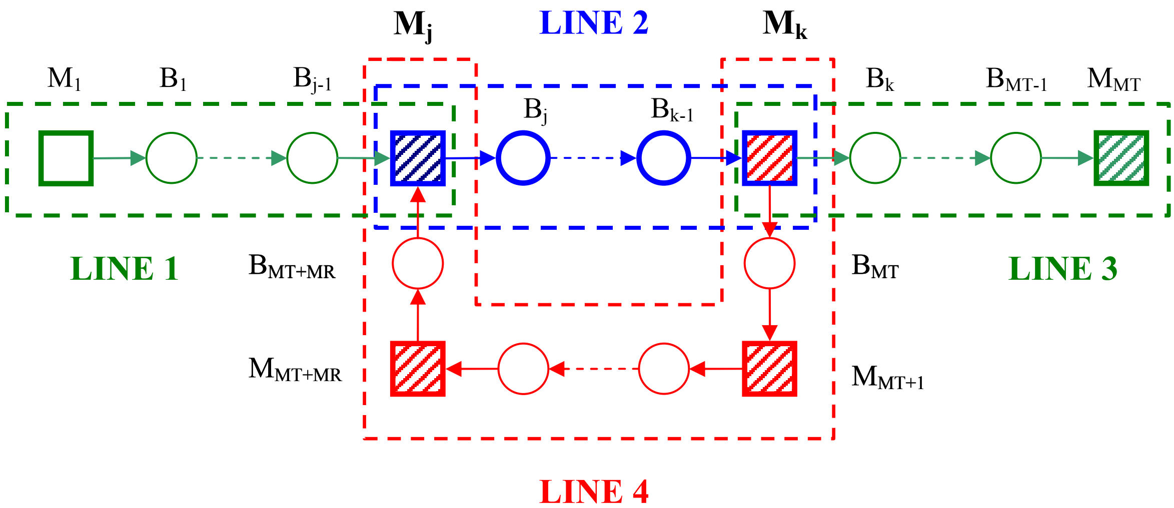

Figure 4.1. Production system with Rework Loop (Li, 2004)

In this production system, machines $M_1, \ldots, M_{j-1}, M_{j+1}, \ldots, M_{k-1}$ produce bad parts and these bad parts can be detected by M_k (split machine) and sent to the rework line, viz. **Line 4** at where they receive some treatments. These parts are fed back into the **Line 2** at machine M_j. The parts that meet the desired quality level are transferred into the **Line 3**. We assume that, the machines in **Line 3** and **Line 4** never make quality failure and always produce good parts.

In the case of when quality information feedback stopping policy is integrated into the system, we assume that machines M_j and M_k can detect the bad parts, and M_j may stop the **Line 1**'s machines while they are producing bad parts and M_k may stop the **Line 2**'s machines while they are producing bad parts.

The probability that a part is defective (bad) and sent to the rework line is α. We define α as rework rate and we will express this rate as a function of the yields of **Line 1** and **Line 2.**

4.1.1. The Solution Technique of Rework Loop Systems without Quality Information Feedback

The assumptions of our model are;

- Material flow is continuous, and μ_i is the service rate at which machine M_i ($i=1,\ldots,MT+MR$) processes material while it is operating and not constrained by the other machine or the buffer. It is a constant, in that μ_i does not depend on the repair state of the other machine or the buffer level.
- Each buffer B_i $i=1,\ldots,MT+MR$ is characterized by its capacity, $0<N_i<\infty$.
- Machine M_i is blocked at time t if B_i is full at time *t*. Machine M_{MT} is never blocked and machine M_1 is never starved. In particular, machine M_k is blocked by the main line if it processes a good part and B_k is full or it is blocked by the rework loop if it processes a defective part and B_{MT} is full.
- Machine M_j can process material either from B_{j-1} or B_{MT+MR}. To avoid deadlock, it is assumed that the merge machine gives the priority always to the repaired parts and takes parts from B_{MT+MR} first if it is not empty.
- M_k detects all bad parts and sends them to rework
- All the repair and failure rates of the machines are exponentially distributed.

In the model, there are two kinds of failure modes:

- ***Operational Failure:*** The machine stops producing parts due to failures like motor burnout.
- ***Quality Failure:*** The machine stops producing good parts and starts producing bad parts due to a failure like sudden tool damage.

All machines in the system make operational failures. The machines in **Line 1** and **Line 2** make persistent quality failures (machine $M_1,\ldots,M_{j-1},M_{j+1},\ldots,M_{k-1}$, in Figure 4.1) except split and merge machines (machines M_k, M_j). We model these machines as a discrete state, continuous time Markov process and the number of states of these machines is three: (Kim Joongyoon, 2005)

- **State 1:** The machine is operating and producing good parts.
- **State -1:** The machine is operating and producing bad parts, but the operator does not know this yet.
- **State 0:** The machine is not operating.

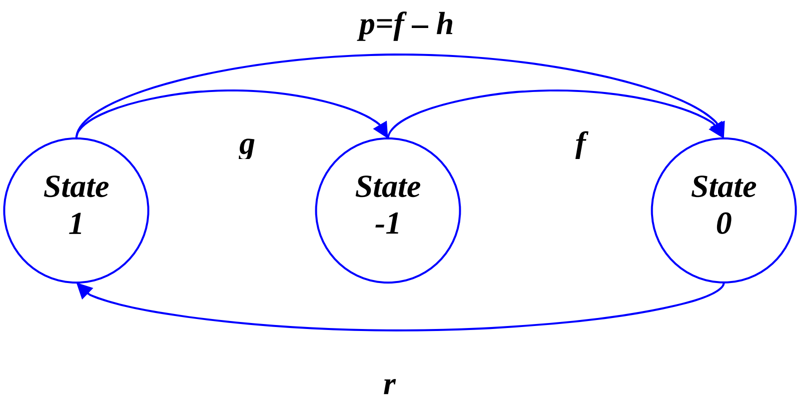

Figure 4.2. States of a machine that produces bad parts (Kim, 2005)

When a machine is in *State 1*, it can fail due to a non-quality-related event. It goes to *State 0* with rate p. After an operator fixes it, the machine goes back to *State 1* with rate r. The machine makes a transition from *State 1* to *State -1* with rate of g due to an assignable quality failure. Here g is the reciprocal of the *Mean Time To Quality Failure* (MTQF). A more stable operation leads to a larger MTQF and a smaller g.

When the machine is in *State -1*, it can be stopped for two reasons: It may experience the same kind of operational failure as it does when it is in *State 1*; or the operator may stop it for repair when he realizes that it is producing bad parts. The transition from *State -1* to *State 0* occurs at rate f which is equal to sum of p and h ($f = p + h$) where h is the reciprocal of the *Mean Time To Detect* bad parts *(MTTD)*. This implies that $f > p$. A more reliable inspection leads to a larger *f*. All these parameters are distributed exponentially.

The steady states probabilities of this model are (Kim Joongyoon, 2005)

$$P(1) = \frac{1}{1 + (p+g)/r + g/f} \tag{4.1}$$

$$P(0) = \frac{(p+g)/r}{1 + (p+g)/r + g/f} \tag{4.2}$$

$$P(-1) = \frac{g/f}{1 + (p+g)/r + g/f} \tag{4.3}$$

The *total production rate* of a machine, including good and bad parts, is

$$P_T = \mu\left(P(1) + P(-1)\right) = \mu \frac{1 + g/f}{1 + (p+g)/r + g/f} \tag{4.4}$$

The *effective production rate* of a machine, the production rate of good parts only, is

$$P_E = \mu\left(P(1)\right) = \mu \frac{1}{1 + (p+g)/r + g/f} \tag{4.5}$$

The fraction of input to a system that is transformed into output of acceptable quality is the yield of a system. The *yield* of a machine *is*

$$\frac{P_E}{P_T} = \frac{P(1)}{P(1) + P(-1)} = \frac{f}{f+g} \tag{4.6}$$

The *split machine*, the *merge machine* and the machines in **Line 3** and **Line 4** are modeled as a two-state continuous time Markov chain since they only make operational failures.

- **State 1:** The machine is operational
- **State 0:** The machine is under repair.

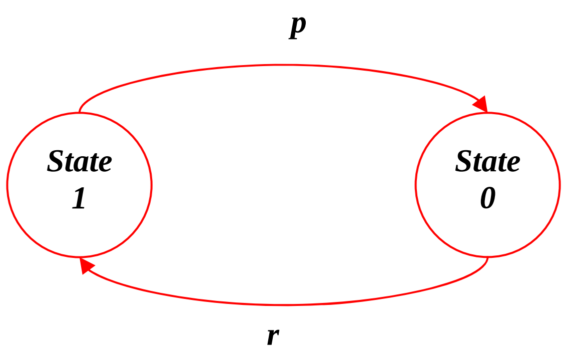

Figure 4.3. States of a machine that does not produce bad parts

When a machine is in *State 1*, it can make an operational failure with rate p. When the machine is in *State 0*, the probability rate that a repair is completed is r. Both the time between failures and the time until repairs are exponentially distributed.

To analyze the production systems with rework loop, we decompose the system into four serial transfer lines; **Line 1, Line 2, Line 3,** and **Line 4**. Therefore, we need an iterative algorithm to solve these serial lines. In the following sections solution techniques are given for analyzing serial transfer lines both with and without quality failures.

<u>4.1.1.1 Performance evaluation of serial lines without quality failures.</u> The **exact** analytical solutions (the production rate and average work in process) of production systems with finite buffer are only available in the case of two-machine-one-buffer transfer lines.

The most important performance measures of 2M1B system with two-state machines are the throughput rate, the average inventory, probability of the starvation of the second machine and the blocking probability of the first machine. In order to find these measures, we use the exact solution technique of 2M1B system that is proposed by Gershwin (1994) in the case of the continuous model. Here, the buffer level can change only a small amount during a short time interval. Therefore, Gershwin uses differential equations in order to evaluate performance of 2M1B system. The MATLAB code which finds the performance parameters of 2M1B system can be seen in Appendix C.

However, it is very difficult to obtain exact analytical solutions of transfer lines with more than three machines. The major reason is that the system states increase exponentially with the increase of machines. As a result, two main approximate techniques have been proposed: ***Decomposition methods*** and ***Aggregation methods***

The idea of decomposition technique is to decompose the analysis of a multi-stage line into the analysis of a set of two-machine lines (an upstream machine $M_u(i)$ and a downstream machine $M_d(i)$, separated by a buffer $B(i)$), i.e., $L(i)$ $i=1,\ldots,k-1$ where k is the number of machines, which are much easier to analyze (Figure 4.4).

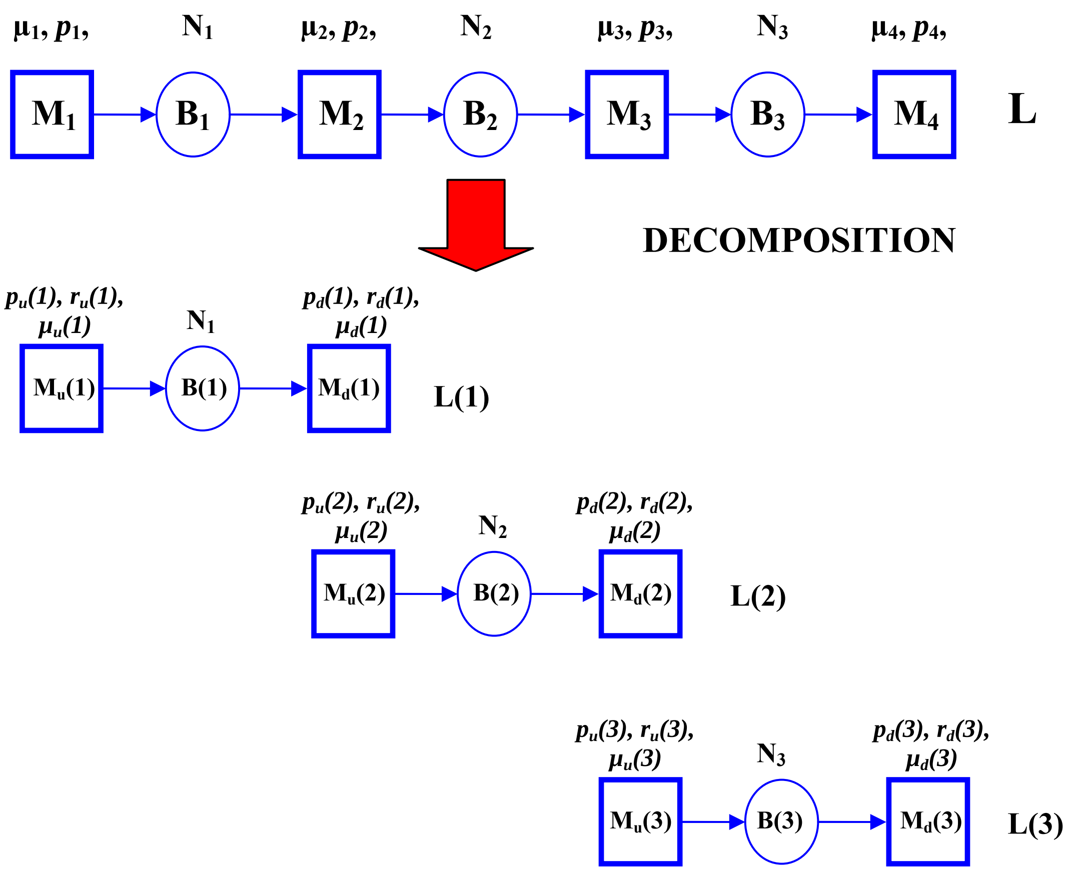

Figure 4.4 Decomposition of a four-machine line into three 2M1B system (Kim, 2005)

The behavior of the each of the decomposed 2M1B system is equivalent to the original system. There exist decomposition techniques for long non-homogenous lines such as *DDX algorithm* proposed by Dallery *et al.* (1988) and *accelerated DDX algorithm* formulated by Burman (1995). The principle of the decomposition is to determine the characteristics of the machines of each line *L(i)* such that the behavior of material flow through buffer *B(i)* closely matches that of the flow in buffer B_i of line *L*. (Figure 4.4)

We evaluate the long continuous transfer lines by using *the overlapping decomposition* algorithm. Overlapping Decomposition is a system-theoretic method which is presented by Li (2004) for the analysis of complex production system. This method requires less computation than the other solution methods. The principle is to choose the throughput rates of both machines of line *L(i)* $i=1,\ldots,k-1$ to be equal to those of the machines of line *L* and the capacity of buffer *B(i)* which is equal to N_i.

The solution method of the serial lines without quality failures is the following:

- INITIALIZATION

The boundary conditions are:

$$p_u(1) = p_1 \tag{4.7}$$

$$p_d(k-1) = p_k \tag{4.8}$$

$$r_u(1) = r_1 \tag{4.9}$$

$$r_d(k-1) = r_k \tag{4.10}$$

$$\mu_u(i) = \mu_i \quad i = 1,\ldots,k-1 \tag{4.11}$$

$$\mu_d(i) = \mu_{i+1} \quad i = 1,\ldots,k-1 \tag{4.12}$$

Provide the following initial guesses for the parameters of each 2M1B line:

$$b(i) = 0.1 \qquad i = 1,\ldots,k-2 \tag{4.13}$$

where *b(i)* is the probability that Machine $M_d(i)$ (or M_{i+1}) is blocked.

- ITERATIONS

Perform the steps until the convergence criterion is satisfied.

- *Step 1* Evaluate *L(1)* and find the throughput rate of this decomposed line, i.e. *P(1)*, and find $s(1)$ that the probability of machine $M_d(1)$ (or M_2) is starved by using the parameters $\mu_u(1)$, $\mu_d(1)$, N_1, $r_u(1) = r_1$, $p_u(1) = p_1$, and;

$$r_d(1) = r_2 \times (1 - b(1)) \tag{4.14}$$

$$p_d(1) = p_2 + r_2 \times b(1) \tag{4.15}$$

- *Step 2* Let *i* range over values from 2 to *k-2*. Compute the following parameters as the formulas 4.16 – 4.19 with the most recent values of $s(i-1)$ and $b(i)$.

$$r_u(i) = r_i \times (1 - s(i-1)) \tag{4.16}$$

$$p_u(i) = p_i + r_i \times s(i-1) \tag{4.17}$$

$$r_d(i) = r_{i+1} \times (1 - b(i)) \tag{4.18}$$

$$p_d(i) = p_{i+1} + r_{i+1} \times b(i) \tag{4.19}$$

Analyze *L(i)* and find $s(i)$ and $b(i-1)$ by using the continuous 2M1B system model with parameters $r_u(i),\ p_u(i),\ r_d(i),\ p_d(i),\ \mu_u(i),\ and\ \mu_d(i)$.

- *Step 3* Evaluate *L(k-1)* and find the throughput rate of this decomposed line, i.e. *P(k-1)*, and find $b(k-2)$ that the probability of machine $M_u(k-1)$ (or M_{k-1}) is blocked by using the parameters $\mu_u(k-1)$, $\mu_d(k-1)$, N_{k-1}, $p_d(k-1) = p_k$, $r_d(k-1) = r_k$, and;

$$r_u(k-1) = r_{k-1} \times (1 - s(k-2)) \tag{4.20}$$

$$p_u(k-1) = p_{k-1} + r_{k-1} \times s(k-2) \tag{4.21}$$

- THE CONVERGENCE CRITERIA : Stop the procedure when $\|P(k-1) - P(1)\| < \varepsilon$

In the analysis of transfer lines, we always assume that the first machine is never starved and last machine is never blocked. The parameters of machines in the decomposed 2M1B lines (*L(i) i=1,…,k-1*) are modified by taking into account for the existence of other machines and buffers, and we embed the starvation and blocking probabilities into the isolated efficiencies' of decomposed lines' machines. For instance, consider the decomposed line *L(1)*. Here, $M_d(1)$ is "not producing" if it is down or starved. Also, while we are estimating the performance parameters of *L(1)*, we should take into account the probability that machine $M_d(1)$ is blocked due to the other downstream machine's failures. Therefore, $r_d(1)$ and $p_d(1)$ are selected by following the conservation of flow such that:

$$\frac{r_d(1)}{r_d(1)+p_d(1)} = \frac{r_2}{r_2+p_2}\cdot\left(1-prob\{M_2\ is\quad blocked\}\right) \tag{4.22}$$

We set

$$r_d(1) = r_2.(1-prob(M_2\ \ is\ \ blocked)) \tag{4.23}$$

and from equation 4.22,

$$p_d(1) = p_2 + r_2.prob(M_2\ \ is\ \ blocked) \tag{4.24}$$

Similar principles are used for the upstream machines of decomposed 2M1B lines. For machine $M_u(i)$;

$$\frac{r_u(i)}{r_u(i)+p_u(i)} = \frac{r_i}{r_i+p_i}\cdot\left(1-prob\{M_i\ is\quad starved\}\right) \tag{4.25}$$

From equation 4.25, we set $r_u(i)$ equal to $r_i.(1-prob(M_i\ \ is\ \ starved))$ and $p_u(i) = p_i + r_i.prob(M_i\ \ is\ \ starved)$.

<u>4.1.1.2 Performance evaluation of serial lines with quality failures.</u> In this section, we are interested in serial production lines where some of the machines have random yield. The behavior of the model is the same as the one described in the previous section. Some of the machines in the line have both operational and quality failure, therefore a fraction of parts that produced in the line are not perfect quality.

Kim and Gershwin (2005) are the first to propose an approximate method for analyzing the serial transfer lines with quality and operational failures. We use their solution algorithm in order to analyze this kind of production systems. The assumptions of the case as follows:

- Each machine has both operational failures and quality failures.
- Each operation works on different features. Thus, quality failures at an operation do not influence the quality of other operations.
- Inspection at machine M_i can detect bad parts made by itself, not others.
- There is no scrap or rework in the line (Kim Jongyoon, 2005).

Every machine in the line has five parameters: $\mu_i,\ \ r_i,\ \ p_i,\ \ g_i,\ \ and\, f_i$. In the previous case where the machines do not have quality failures, we introduce $4(K-1)$ pseudo-machine parameters in order to analyze the system. In this case, we do not require $8(K-1)$ pseudo-machine parameters since *quality failures at an operation do not influence the quality of other operations* because each operations work on different features. Therefore,

$$g_u(i) = g_i \tag{4.26}$$

$$g_d(i) = g_{i+1} \tag{4.27}$$

Another fundamental assumption is that, machines can only identify bad features <u>made by its own operation</u>. Thus, f_i is also independent of other machines' parameters.

$$f_u(i) = f_i \tag{4.28}$$

$$f_d(i) = f_{i+1} \tag{4.29}$$

So, we have $4(K-1)$ remaining equations. To analyze this system, Kim and Gershwin derive a relationship between the three-state machine model and the two-state machine model. They transform the three-state machine into the two-state machine model by consolidating the two up states of the three-state machine into the up state of the two-state model. (Figure 4.5)

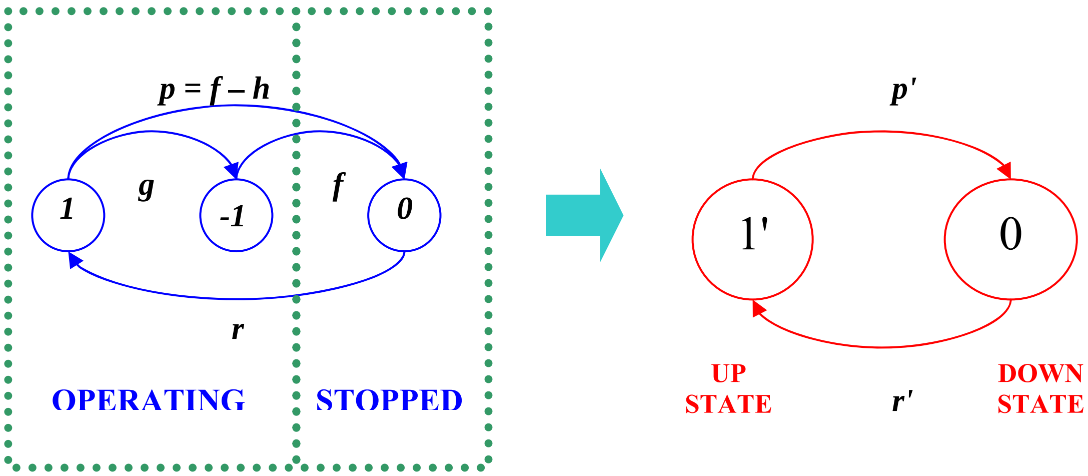

Figure 4.5 Transformation of the three-sate model into the two-state model (Kim, 2005)

By equating the sum of mean values of *State 1* and *State -1* of three-state machine model with the mean value of *State 1'* of two-state machine model, we express ***p'*** rate as a function of ***p, g*** and ***f*** (Kim, 2005);

$$\frac{1}{p'} = \frac{1}{p+g} + \frac{g}{(p+g)f} \tag{4.30}$$

$$p' = \frac{f(p+g)}{(f+g)} \tag{4.31}$$

As a result, the two-state machine model can approximate the three-state machine model with machine parameters $p' = \frac{f(p+g)}{(f+g)}, \quad \mu' = \mu, \quad r' = r$.

The equivalent two-state machine model gives us the total production rate of the three-state machine model. But the effective production rate should be estimated indirectly since the two-state machine model can not tell the difference between "good" state and "bad" state. We know that, the *yield* of a machine is $\frac{f}{f+g}$ and the effective production rate of a machine can be found by multiplying the yield of the machine by the total production rate of the machine. For multiple machine lines, the system yield becomes a product of the individual yields. Thus, the effective production rate can be calculated by multiplying the system yield by the total production rate.

The solution algorithm is the following:

- *Step 1* Transform all the three-state machines into the two-state machine model by setting $p'_i = \dfrac{f_i(p_i + g_i)}{(f_i + g_i)}$, $\mu_i' = \mu_i$, $r_i' = r_i$ $i = 1,...,k$ and $N_i' = N_i$ $i = 1,...,k$ where k is the number of the machines.

- *Step 2* Analyze the new transformed line by using overlapping decomposition techniquein section 4.1.1.1 with new parameters p'_i r_i' μ_i' N_i' and calculate the total production rate and average inventory levels.

- *Step 3* Calculate the system yield by multiplying all individual machines' yield.

$$Y_{sys} = \prod_{i=1}^{k} \frac{f_i}{f_i + g_i} \tag{4.32}$$

- *Step 4* Evaluate the effective production rate, i.e. PE, by multiplying the system yield (Y_{sys}) by the total production rate PT.

$$PE = PT \times Y_{sys} \tag{4.33}$$

4.1.2. The Iterative Solution Procedure for the Production Systems with Rework Loop

Due to the complexity in the production systems with rework loop, direct analysis is not possible. Therefore we use overlapping decomposition technique once more. The idea of the approach is to decompose the system into serial transfer lines, where the first and last machines in one serial line are overlapped with another serial line, and to modify the overlapped machines' parameters to accommodate the effects of other lines.

Our iterative procedure for evaluating the systems with rework loop is different from the solution technique that is offered by Li (2004) in that;

- Li (2004) assigns a *constant rework rate*, i.e. α , in his algorithm
- Li (2004) assumes that each machine has two states, makes time dependent failures and is capable of producing with the rate 1 part per unit of time
- Li (2004) uses aggregation technique in evaluation of the serial transfer lines

In our algorithm, the rework rate (the probability that a part will be reworked) of the system depends on all the individual machines' (the machines that have quality failures) yields. Thus, we have to express the rework rate as a function of machine yields (or yield of the decomposed serial lines). We use decomposition technique in the analysis of serial lines and assume operation dependent failures. In our system, the machines making quality failures have three states and may have different service rates (we are considering non-homogenous lines).

We decompose the production system in Figure 4.1 into four serial lines; **Line 1, Line 2, Line 3** and **Line 4** and introduce the fictitious machines; $\boldsymbol{M'_k}$, $\boldsymbol{M''_k}$, $\boldsymbol{M'''_k}$, $\boldsymbol{M'_j}$, $\boldsymbol{M''_j}$ and $\boldsymbol{M'''_j}$ (Figure 4.6).

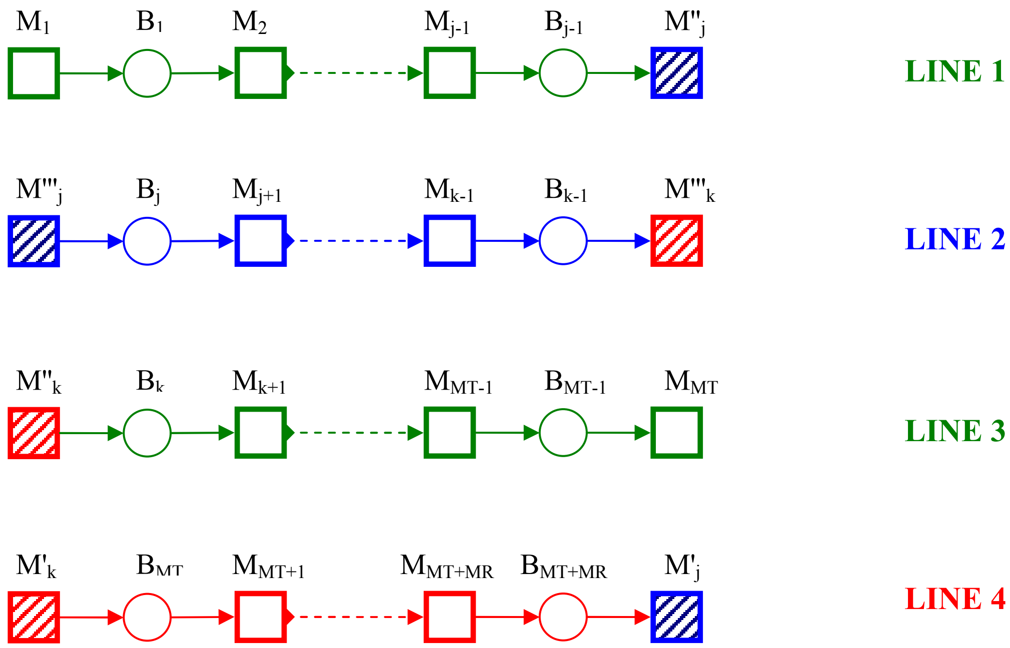

Figure 4.6. Decomposed serial lines of the production system with rework loop

The throughput rate of the system in Figure 4.1 is equal to the throughput rate of **Line 1** and **Line 3** in Figure 4.6. Also, the difference of the throughput rates of **Line 2** and **Line 4** gives us the throughput rate of the system. So, we can analyze the production system with rework loop by evaluating the serial **Lines 1, 3, 2 and 4** with the procedure in section 4.1.1.2. In order to use these procedures for the serial lines at hand, the parameters of machines M_k and M_j are modified so as to account for the existence of other machines and the rework rate should be expressed as a function of yields of the lines.

We assume that all machines in **Line 3** and **Line 4** always produce good parts. In contrast, all machines but machines M_k and M_j in **Line 1** and **Line 2** have quality failures and may produce bad parts. As a result, the rework rate, i.e. α, depends on the yields of **Line 1** and **Line 2**.

The computation of the "Rework Rate" (α) is the following: When a part is processed by M_k (split machine), the part will be transferred into the **Line 4** (rework line) with probability α. This probability depends on the probability that **Line 1** and **Line 2** produce bad parts. Therefore we have to express this rework rate as a function of yields of

Line 1 and **Line 2**. We introduce the following notation for simplification:

PT (i): Total throughput rate of serial ***Line i,*** $i = 1,2,3,4$

Y_1 : Yield of serial **Line 1**;

Y_2 : Yield of serial **Line 2**;

b_j : Probability {machine M_j is blocked};

s_k : Probability {machine M_k is starved };

b_{k1} : Probability {machine M_k is blocked by B_k};

b_{k2} : Probability {machine M_k is blocked by B_{MT}};

s_{j1} : Probability {machine M_j is starved by B_{j-1}};

s_{j2} : Probability {machine M_j is starved by B_{MT+MR}};

In the production system in Figure 4.1, all parts that are produced in **Line 1** are moved into **Line 3** through the **Line 2** either directly or after going through **Line 4**. The probability that a part is conveyed from **Line 1** to **Line 3** without being reworked is $Y_1 \times Y_2$. As a result, $PT(1).(1 - Y_1.Y_2)$ parts are reworked at least one time. A part that is reworked in **Line 4** is transferred into **Line 3** without being reworked again with probability Y_2. Therefore, $PT(1).(1 - Y_1.Y_2)(1 - Y_2)$ parts are reworked at least two times. Consequently,

$$PT(4) = PT(1).(1 - Y_1.Y_2).\sum_{i=1}^{\infty}(1 - Y_2)^{i-1} \tag{4.34}$$

$$PT(4) = PT(1).\frac{1 - Y_1.Y_2}{Y_2} \tag{4.35}$$

From the formulas in Li (2004),

$$\frac{PT(1)}{PT(4)} = \frac{(1-\alpha)}{\alpha}.\frac{(1-b_{k1})}{(1-b_{k2})} \tag{4.36}$$

So, by setting $\frac{Y_2}{1 - Y_1.Y_2} = \frac{(1-\alpha)}{\alpha}.\frac{(1-b_{k1})}{(1-b_{k2})}$ we have

$$\alpha = \frac{(1 - Y_1.Y_2).(1 - b_{k1})}{Y_2.(1 - b_{k2}) + (1 - Y_1.Y_2).(1 - b_{k1})} \tag{4.37}$$

Now, we are ready to use the overlapping decomposition technique and evaluate the serial lines in Figure 4.6 with modified parameters of machines M_k and M_j (i.e., split and merge). Assume that we know the probability that M_k is starved, i.e., s_k, the probability that M_j is blocked, i.e., b_j, and α . We first introduce the fictitious machines that are denoted as M'_k and M'_j with parameters p'_k, r'_k, p'_j and r'_j defined as

$$r'_k = r_k.\alpha.(1 - s_k) \tag{4.38}$$

$$p'_k = p_k + r_k.(1 - \alpha(1 - s_k)) \tag{4.39}$$

$$r'_j = r_j(1 - b_j) \tag{4.40}$$

$$p'_j = p_j + r_j.b_j \tag{4.41}$$

where these parameters are selected such that

$$\frac{r'_k}{r'_k + p'_k} = \frac{r_k}{r_k + p_k}.\alpha.(1 - s_k) \tag{4.42}$$

$$\frac{r'_j}{r'_j + p'_j} = \frac{r_j}{r_j + p_j}.(1 - b_j) \tag{4.43}$$

We can calculate the throughput rate of **Line 4**, *PT(4)* with Pr{ machine M_j is starved by B_{MT+MR} } (s_{j2}), and Pr{machine M_k is blocked by B_{MT}} (b_{k2}) by using the serial line evaluation procedure in section 4.1.1.1

Analogously, we can calculate the throughput rate of **Line 3** and the probability that machine M_k is blocked by B_k (b_{k1}) by modifying the parameters of M_k. We assumed that we know the probability that M_k is starved by B_{k-1} (s_k) which is the only needed parameter in order to analyze **Line 3**.

The production rate of **Line 1** can be calculated similarly because we found the probability that M_j is starved by B_{MT+MR} from the analysis of **Line 4** and we assumed that the probability that M_j is blocked by B_j is known.

Finally, the production rate of **Line 2** depends on the probabilities that machine M_k is blocked by B_k *and* B_{MT}, and the probabilities that machine M_j is starved by B_{k-1} and B_{j-1}.

These probabilities were found in evaluation of the **Line 4**, **Line 3** and **Line 1**. Then, the production rate of the system (which equals to the production rate of Line 3) can be obtained.

The following algorithm gives the iterative calculation procedure concisely

- INITIALIZATION

$b_j = 0.1$, $s_k = 0.1$, $\alpha = 0.1$

- ITERATIONS

Perform the steps until the convergence criterion is satisfied.

- *Step 1* Introduce fictitious machines M'_k (first machine of **Line 4**) and M'_j (last machine of **Line 4**) with parameters

$$r'_k = r_k.\alpha.(1 - s_k) \tag{4.44}$$

$$p'_k = p_k + r_k.(1 - \alpha(1 - s_k)) \tag{4.45}$$

$$r'_j = r_j(1 - b_j) \tag{4.46}$$

$$p'_j = p_j + r_j.b_j \tag{4.47}$$

Analyze **Line 4** by using evaluation procedure in section 4.1.1.1 and find *PT(4)*, b_{k2}, s_{j2} and average buffer levels.

- *Step 2* Introduce fictitious machines M''_k (first machine of **Line 3**) with parameters

$$r''_k = r_k(1 - \alpha)(1 - s_k) \tag{4.48}$$

$$p''_k = p_k + r_k.(1 - (1 - \alpha)(1 - s_k)) \tag{4.49}$$

Analyze **Line 3** by using evaluation procedure in section 4.1.1.1 and find *PT(3)*, b_{k1} and average buffer levels.

- *Step 3* Introduce fictitious machines M''_j (last machine of **Line 1**) with parameters

$$r''_j = r_j s_{j2}\left(1-b_j\right) \tag{4.50}$$

$$p''_j = p_j + r_j.\left(1-s_{j2}\left(1-b_j\right)\right) \tag{4.51}$$

Note that s_{j2} is found in step 1. Analyze **Line 1** by using evaluation procedure in section 4.1.1.2 and find *PT(1)* (total production rate of Line 1), s_{j1}, Y_1, and average buffer levels.

- *Step 4* Introduce fictitious machines M'''_j (first machine of Line 2) and M'''_k (last machine of **Line 2**) with parameters

$$r'''_j = r_j\left(1-s_{j1}s_{j2}\right) \tag{4.52}$$

$$p'''_j = p_j + r_j s_{j1} s_{j2} \tag{4.53}$$

$$r'''_k = r_k\left(1-\alpha.b_{k2}-\left(1-\alpha\right)b_{k1}\right) \tag{4.54}$$

$$p'''_k = p_k + r_k\left(\alpha.b_{k2}+\left(1-\alpha\right)b_{k1}\right) \tag{4.55}$$

Note that, b_{k2} is found in step 1, b_{k1} is found in step 2, and s_{j1} is found in step 3, Analyze **Line 2** by using evaluation procedure in section 4.1.1.2 and find *PT(2)*, b_j, s_k and average buffer levels.

- *Step 5* Estimate the new rework rate (α) by using the formula (4.37).

$$(\alpha = \frac{(1-Y_1.Y_2).(1-b_{k1})}{Y_2.(1-b_{k2})+(1-Y_1.Y_2).(1-b_{k1})})$$

- *Step 6* Go back to Step 1, use new α, b_j, s_k values for the next iteration, and perform the steps 1 to 5 until the convergence criteria is satisfied. The production rate of the system is equal to *PT(3)* (or *PT(1)*) .

- <u>THE CONVERGENCE CRITERIA</u>: Stop the procedure when $|PT(3) - PT(1)|$ and $|[PT(2) - PT(4)] - PT(1)|$ are smaller than a pre-defined small number ε. Li (2004) shows that the iterations are convergent and results in the estimate of system throughput rate.

4.1.3. Accuracy of the Model

The accuracy of the proposed solution technique for the production system with rework loop has been tested by comparing the results with a continuous-time, discrete part simulation. The simulation results have been obtained using Arena simulation software. The number of replications is selected as 30 and each replication consists of 120,000 time units. The warm-up period is selected as 20,000 time units for each replication.

In Table 4.1, *PT1*, *PT2*, *PT3* and *PT4* denote the estimates of the production rates of serial **Line 1**, **Line 2**, **Line 3**, and **Line 4** respectively. These parameters are calculated by the solution technique in 4.1.2. By changing machine and buffer parameters, 30 cases are generated which are given in Appendix A. In all cases, there are 8 machines in the main line and 3 machines in the rework line. Also, the fourth machine is the merge machine and the seventh machine is the split machine in all cases. The % errors in the production rates are calculated from

$$PT \quad \%error = \frac{PT(A) - PT(S)}{PT(S)} \times 100 \tag{4.56}$$

where *PT(A)s* are the total production rate calculated from the analytical model in 4.1.2, and *PT(S)* is the total production rate estimated from the simulation.

Table 4.1. Validation of *PT(i)* in rework loop system **without *qif***

Case Number	Throughput Rate	Simulation	Analytical	% Err
CASE 1	***PT1***	0.612	0.624	**1.83%**
	PT2	1.524	1.523	**-0.07%**
	PT3	0.612	0.624	**1.82%**
	PT4	0.911	0.899	**-1.35%**
CASE 2	***PT1***	0.441	0.440	**-0.12%**
	PT2	1.461	1.457	**-0.26%**
	PT3	0.441	0.440	**-0.13%**
	PT4	1.019	1.016	**-0.32%**
CASE 3	***PT1***	0.637	0.637	**0.08%**
	PT2	1.454	1.454	**0.03%**
	PT3	0.637	0.637	**0.08%**
	PT4	0.816	0.816	**-0.02%**
CASE 4	***PT1***	0.469	0.470	**0.15%**
	PT2	1.635	1.633	**-0.10%**
	PT3	0.469	0.470	**0.14%**
	PT4	1.165	1.163	**-0.20%**
CASE 5	***PT1***	1.146	1.153	**0.63%**
	PT2	1.777	1.777	**-0.01%**
	PT3	1.146	1.153	**0.63%**
	PT4	0.631	0.624	**-1.17%**
CASE 6	***PT1***	0.599	0.608	**1.47%**
	PT2	1.399	1.399	**0.02%**
	PT3	0.599	0.608	**1.47%**
	PT4	0.800	0.791	**-1.07%**
CASE 7	***PT1***	0.627	0.629	**0.43%**
	PT2	1.399	1.398	**-0.08%**
	PT3	0.627	0.629	**0.43%**
	PT4	0.772	0.768	**-0.51%**
CASE 8	***PT1***	0.700	0.703	**0.38%**
	PT2	2.063	2.061	**-0.13%**
	PT3	0.700	0.703	**0.38%**
	PT4	1.363	1.357	**-0.39%**
CASE 9	***PT1***	0.505	0.507	**0.35%**
	PT2	1.721	1.726	**0.26%**
	PT3	0.505	0.507	**0.33%**
	PT4	1.216	1.218	**0.23%**
CASE 10	***PT1***	0.651	0.651	**0.08%**
	PT2	1.396	1.395	**-0.09%**
	PT3	0.651	0.651	**0.08%**
	PT4	0.745	0.743	**-0.25%**

Table 4.2. Validation of *PT(i)* in rework loop system **without *qif*** – continued

Case Number	Throughput Rate	Simulation	Analytical	% Err
CASE 11	***PT1***	0.653	0.658	**0.65%**
	PT2	1.748	1.747	**-0.08%**
	PT3	0.653	0.658	**0.65%**
	PT4	1.094	1.089	**-0.51%**
CASE 12	***PT1***	0.745	0.743	**-0.19%**
	PT2	1.401	1.397	**-0.27%**
	PT3	0.745	0.743	**-0.19%**
	PT4	0.656	0.653	**-0.36%**
CASE 13	***PT1***	0.677	0.678	**0.11%**
	PT2	1.557	1.557	**-0.01%**
	PT3	0.677	0.678	**0.11%**
	PT4	0.879	0.879	**-0.11%**
CASE 14	***PT1***	0.619	0.621	**0.27%**
	PT2	1.522	1.522	**-0.01%**
	PT3	0.619	0.621	**0.25%**
	PT4	0.902	0.900	**-0.19%**
CASE 15	***PT1***	0.764	0.763	**-0.04%**
	PT2	1.544	1.542	**-0.09%**
	PT3	0.764	0.763	**-0.05%**
	PT4	0.780	0.779	**-0.15%**
CASE 16	***PT1***	0.221	0.221	**-0.21%**
	PT2	0.740	0.737	**-0.36%**
	PT3	0.221	0.221	**-0.20%**
	PT4	0.518	0.516	**-0.43%**
CASE 17	***PT1***	0.239	0.239	**0.17%**
	PT2	0.734	0.734	**0.07%**
	PT3	0.239	0.239	**0.17%**
	PT4	0.495	0.495	**0.03%**
CASE 18	***PT1***	0.347	0.348	**0.22%**
	PT2	0.806	0.806	**0.03%**
	PT3	0.347	0.348	**0.23%**
	PT4	0.458	0.458	**-0.12%**
CASE 19	***PT1***	1.377	1.380	**0.22%**
	PT2	2.481	2.486	**0.21%**
	PT3	1.377	1.380	**0.21%**
	PT4	1.104	1.106	**0.22%**
CASE 20	***PT1***	0.283	0.293	**3.59%**
	PT2	0.860	0.866	**0.67%**
	PT3	0.283	0.293	**3.59%**
	PT4	0.577	0.573	**-0.76%**

Table 4.3. Validation of *PT(i)* in rework loop system **without *qif*** – continued

Case Number	Throughput Rate	Simulation	Analytical	% Err
CASE 21	***PT1***	0.318	0.314	**-1.27%**
	PT2	0.985	0.974	**-1.15%**
	PT3	0.318	0.313	**-1.29%**
	PT4	0.667	0.660	**-1.09%**
CASE 22	***PT1***	0.281	0.283	**0.81%**
	PT2	1.912	1.909	**-0.17%**
	PT3	0.281	0.283	**0.75%**
	PT4	1.631	1.626	**-0.33%**
CASE 23	***PT1***	0.285	0.283	**-0.72%**
	PT2	3.256	3.240	**-0.50%**
	PT3	0.285	0.283	**-0.70%**
	PT4	2.971	2.956	**-0.49%**
CASE 24	***PT1***	0.172	0.171	**-0.60%**
	PT2	0.713	0.708	**-0.68%**
	PT3	0.172	0.171	**-0.59%**
	PT4	0.541	0.537	**-0.71%**
CASE 25	***PT1***	0.111	0.111	**0.12%**
	PT2	0.687	0.687	**0.04%**
	PT3	0.111	0.111	**0.10%**
	PT4	0.575	0.575	**0.03%**
CASE 26	***PT1***	1.936	1.937	**0.01%**
	PT2	3.909	3.909	**-0.02%**
	PT3	1.936	1.936	**0.01%**
	PT4	1.972	1.972	**-0.03%**
CASE 27	***PT1***	0.363	0.364	**0.30%**
	PT2	0.713	0.713	**0.03%**
	PT3	0.363	0.364	**0.30%**
	PT4	0.350	0.349	**-0.25%**
CASE 28	***PT1***	0.484	0.486	**0.35%**
	PT2	1.081	1.083	**0.23%**
	PT3	0.484	0.486	**0.36%**
	PT4	0.596	0.597	**0.12%**
CASE 29	***PT1***	0.099	0.099	**0.18%**
	PT2	0.867	0.867	**-0.03%**
	PT3	0.099	0.099	**0.10%**
	PT4	0.767	0.767	**-0.05%**
CASE 30	***PT1***	0.428	0.429	**0.11%**
	PT2	0.725	0.725	**-0.10%**
	PT3	0.428	0.429	**0.08%**
	PT4	0.297	0.296	**-0.37%**

The *%* error in average inventories are calculated from

$$Inv \quad \%error = \frac{Inv(A) - Inv(S)}{0.5 \times N} \times 100(\%) \tag{4.57}$$

where *Inv(A)* and *Inv(S)* are average inventory estimated from the analytical model and the simulation respectively and *N* is the capacity of the buffer. This equation is an unbiased way to calculate the error in average inventory (Kim Jongyoon, 2005).

In each case, the number of buffers in the system is 11 and we calculated the *Inv % error* for each 11 average inventory levels in all cases. The mean of the absolute percent differences of average inventories is the average of absolute *Inv % errors* of those 11 buffers. Table 4.4 shows the mean of the absolute percent differences of average inventories for 30 cases.

Table 4.4 The mean of the absolute percent differences of average inventories for rework loop systems without *qif*

CASE #	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
Mean Error	**1.10%**	**1.12%**	**0.57%**	**0.77%**	**0.75%**	**0.76%**	**0.43%**	**1.75%**	**1.12%**	**0.48%**
CASE **#**	*11*	*12*	*13*	*14*	*15*	*16*	*17*	*18*	*19*	*20*
Mean Error	**0.91%**	**0.70%**	**0.62%**	**0.71%**	**1.66%**	**1.45%**	**1.47%**	**0.58%**	**0.57%**	**5.46%**
CASE **#**	*21*	*22*	*23*	*24*	*25*	*26*	*27*	*28*	*29*	*30*
Mean Error	**1.43%**	**2.74%**	**0.87%**	**1.39%**	**0.37%**	**1.76%**	**0.85%**	**5.89%**	**3.68%**	**0.50%**

Table 4.5 shows the *Inv % errors* of 11 buffers for the Case 28. We choose Case 28 since it has the highest absolute mean error among the all cases.

Table 4.5. *Inv % errors* of average buffer levels for rework loop systems without *qif* – Case 28

	CAPACITY	SIM.	ANALYTICAL	Err %
B1	40	39.946	39.933	-0.03%
B2	35	34.495	34.293	-0.58%
B3	50	49.815	48.639	-2.35%
B4	40	39.984	39.970	-0.03%
B5	40	23.652	30.562	17.28%
B6	30	24.845	28.725	12.93%
B7	25	7.076	2.841	-16.94%
B8	55	48.151	54.479	11.51%
B9	35	28.121	28.169	0.14%
B10	25	0.257	0.353	0.38%
B11	45	2.549	3.737	2.64%

The average absolute value of the % errors in the throughput rate and the mean of the absolute percent differences of average inventories are %0.52 and %1.42 respectively. The observation that the throughput rates estimates are better than average buffer levels is consistent with the literature.

4.2. Quality Information Feedback

Machines are unreliable and can fail for different reasons. An operational failure stops the machine without involving quality issues. Quality failures lead machines to an out-of-control state; in this state machines are operational but produce defective parts (We make the assumption that once a defective part has been produced, all the subsequent parts will be bad until the machine is repaired).

It is very important to catch defective parts and stop the machine as soon as possible to minimize the production of bad parts and the waste of downstream capacity. The only way to stop the machine producing bad parts is due to its own inspection or its own operational failures when there is no quality information feedback. If the downstream machines can detect the bad parts produced by the upstream machines and inform the operator so that he stops the machine to fix the problems, we call this ***quality information feedback*** (inspection at downstream operations can detect bad features made by upstream machines).

As detailed in Kim and Gershwin (2005), the mean time to detect a bad part is a function of the size of the buffer. This is because when buffer gets larger, more material can accumulate between an operation and the inspection of that operation. To stop the upstream machine by quality information feedback, not only the part being processed by downstream machine should be defective, but also all the parts in the intermediate buffer must be defective (since a persistent quality failure takes place and the downstream machine detects that the upstream machine producing bad parts). Otherwise, if there is non-defective parts in the intermediate buffer, upstream machine will not be stopped. (Figure 4.7, Figure 4.8). If the buffer is larger, there tends to be more material in the buffer and consequently more material is produced before detection occurs.

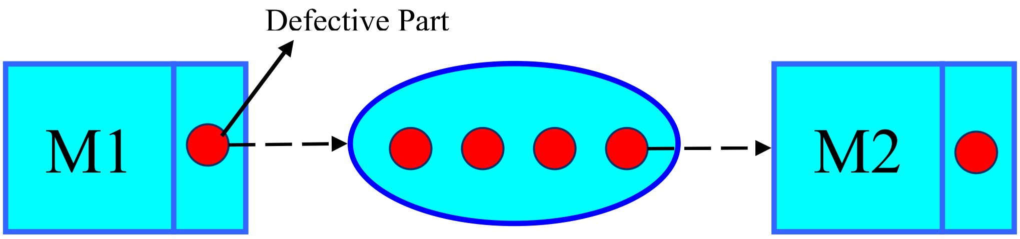

Figure 4.7 2M1B System where quality information feedback occurs

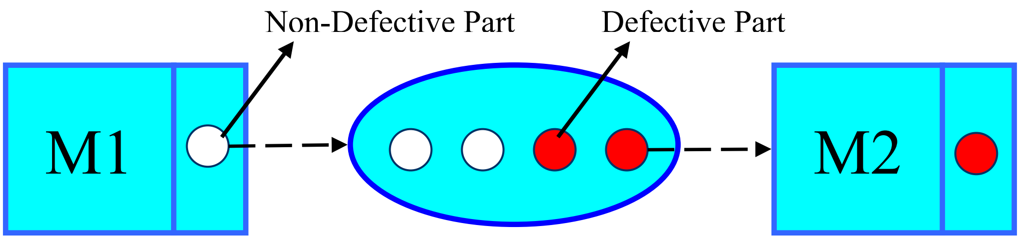

Figure 4.8 2M1B System where quality information feedback does not occur

In order to take into account the quality information feedback, we adjust the transition rate f from *State* -1 to *State 1*. We define the new rate as f^q. Quality information feedback decreases the time required to fix the machine when it produces bad parts. Therefore, f^q rate is higher than f rate.

For simplification, we first consider a 2M1B system and define f_1^q as the transition rate of M_1 from *State -1* to *State 1* when there is quality information feedback and f_1 as the transition rate without quality information feedback. Assumptions of the model are as follows:

- The first machine (M_1) and the last machine (M_2) have both operational failures and quality failures. B_1 is the intermediate buffer between M_1 and M_2 with capacity N_1. The machines are modeled as a three-state Markov Chain shown in Figure 4.2
- p_i's ($i = 1,2$) are the operational failure rates, r_i's ($i = 1,2$) are the repair rates, g_i's ($i = 1,2$) are quality failure rates and f_i's ($i = 1,2$) are the transition rates from *State -1* to *State 0*.
- Each machine works on different features. Quality failures at an operation do not influence the quality of other operations.
- Machine M_1 can only detect the abnormalities due to its quality failure. Machine M_2 can detect bad parts that are produced both by M_1 and itself.

We call K_1^b the expected number of bad parts generated by M_1 from the time it enters *State -1* until it leaves *State -1* and K_1^g the expected number of good parts produced by M_1 from the moment when M_1 leaves the *State -1* to the next time it arrives at *State -1*.

We defined yield as the fraction of parts – at either final or intermediate stages – that pass inspection, that is, that are measured to satisfy quality standards. We know that the yield of M_1 is $\dfrac{f_1}{f_1 + g_1}$ if there is no quality information feedback. The yield of M_1 is equal to $\dfrac{f_1^q}{f_1^q + g_1}$ when we integrate quality information feedback stopping policy into the 2M1B system. As a result,

$$\frac{f_1^q}{f_1^q + g_1} = \frac{K_1^g}{K_1^g + K_1^b} \tag{4.58}$$

When M_1 is in *State -1*, the probability of a transition to *State 0* before M_1 finishes a part is

$$x_1 = \frac{f_1}{\mu_1} \tag{4.59}$$

Eventually all the parts in the buffer are bad, so that defective parts reach M_2. Then, there is another way that M_1 can move to *State 0* from *State -1*: *Quality Information Feedback.* The probability that inspection at M_2 detects a nonconformity made by M_1 and stop M_1 is

$$x_2 = \frac{h_2}{\mu_2} \tag{4.60}$$

where $\frac{1}{h_2}$ is the mean time to stop M_1 if the all the parts in the buffer are defective when M_2 detects a bad part made by M_1

$$h_2 = f_2 - p_2 \tag{4.61}$$

The expected number of bad parts produced by M_1, i.e., K_1^b, before it makes a transition to *State 0* from *State -1* either by itself or by quality information feedback is

$$K_1^b = \left[x_1 \sum_{i=1}^{w} i \times (1 - x_1)^{i-1} \right] + \left[(1 - x_1)^w \sum_{i=1}^{\infty} (w + i) \times (x_1 + x_2) \times (1 - (x_1 + x_2))^{i-1} \right] \tag{4.62}$$

where w is the average inventory in the buffer B_1. This is an approximation since we simply use the average inventory rather than averaging the expected number of bad parts produced by M_1 depending on different inventory levels w_i. (Kim and Gershwin, 2005) After some mathematical manipulation,

$$K_1^b = \frac{1 - (1 - x_1)^w}{x_1} - w(1 - x_1)^w + \frac{(1 - x_1)^w \left[(w.(x_1 + x_2) + 1 \right]}{x_1 + x_2} \tag{4.63}$$

K_1^g is given in (Kim and Gershwin, 2005) as

$$K_1^g = \frac{\mu_1}{p_1+g_1} + \frac{p_1}{p_1+g_1}\frac{\mu_1}{p_1+g_1} + \left(\frac{p_1}{p_1+g_1}\right)^2 \frac{\mu_1}{p_1+g_1} + \ldots = \frac{\mu_1}{g_1} \tag{4.64}$$

From the equation $\frac{f_1^q}{f_1^q+g_1} = \frac{K_1^g}{K_1^g+K_1^b}$ we have,

$$f_1^q = \frac{\mu_1}{\frac{1-(1-x_1)^w}{x_1} - w(1-x_1)^w + \frac{(1-x_1)^w\left[(w.(x_1+x_2)+1\right]}{x_1+x_2}} \tag{4.65}$$

Since the average inventory is a function of f_1^q and f_1^q is dependent on the average inventory, an iterative method is used to get these values.

- <u>*Step 1*</u> Transform the M_1 and M_2 from the three-state-machines into the two-state machine model by setting $p'_i = \frac{f_i(p_i+g_i)}{(f_i+g_i)}$, $\mu_i' = \mu_i$, $r_i' = r_i$ $i = 1,2$.

- <u>*Step 2*</u> Calculate the system yield; $Y_{sys} = \frac{f_1}{f_1+g_1} \times \frac{f_2}{f_2+g_2}$

- <u>*Step 3*</u> Analyze the new transformed line and calculate the total production rate (PT) and the effective production rate (PE) by using the formula $PE = PT \times Y_{sys}$ and estimate average buffer level, i.e., w to get an initial estimate of f_1^q.

- <u>*Step 4*</u> Adjust f_1^q by using the formula

$$f_1^q = \frac{\mu_1}{\frac{1-(1-x_1)^w}{x_1} - w(1-x_1)^w + \frac{(1-x_1)^w\left[(w.(x_1+x_2)+1\right]}{x_1+x_2}}$$ where $x_1 = \frac{f_1}{\mu_1}$ and $x_2 = \frac{f_2 - p_2}{\mu_2}$.

- <u>*Step 5*</u> Calculate the new system yield $Y_{sys}^{new} = \frac{f_1^q}{f_1^q+g_1} \times \frac{f_2^q}{f_2^q+g_2}$

- *Step 6* Evaluate the system with f_1^q rate by adjusting the parameters $p'_1 = \frac{f_1^q(p_1+g_1)}{(f_1^q+g_1)}, p'_2 = \frac{f_2(p_2+g_2)}{(f_2+g_2)}$ *and* $\mu_i' = \mu_i,\ r_i' = r_i \quad i = 1,2$. Calculate the new total production rate (PT^{new}) and the new effective production rate (PE^{new}) by using the formula $PE^{new} = PT^{new} \times Y_{sys}^{new}$ and estimate new average buffer level, i.e. w^{new}.

- *Step 7* If PT^{new}, PE^{new} and w^{new} are close to PT, PE and w, then stop. Otherwise, set $PT = PT^{new}$, $PE = PE^{new}$ and $w = w^{new}$, go to *Step 4* and repeat the procedure.

  - ***Remark:*** In *Step 4*, we do not change the probabilities x_1 and x_2 at each iteration according to new f_1^q. We just adjust w in the formula.

The above solution technique can be generalized for the *k*-machine transfer line where only the last machine M_k can detect defective parts made by any of the machines M_i $i = 1,\dots,k-1$ (other machines can only detect the bad parts that they produced). The only difference is that there are more than one buffer between M_i $i = 1,\dots,k-2$ and M_k for k-machine transfer line case. Therefore, w in equation (4.65) is replaced by wip_i, $i = 1,\dots,k-1$ where wip_i is the sum of the average buffer levels between machine M_i $i = 1,\dots,k-1$ and machine M_k (last machine of the line). For instance, if there are 5 machines in the line,

$$wip_1 = w_1 + w_2 + w_3 + w_4$$
$$wip_2 = w_2 + w_3 + w_4$$
$$wip_3 = w_3 + w_4$$
$$wip_4 = w_4$$

where w_i is the average inventory level of B_i $i = 1,\dots,k-1$.

The evaluation technique of *k*-machine serial line with quality information feedback is as follow:

- *Step 1* Analyze the k-machine line by using the procedure 4.1.1.2 and calculate the total production rate (PT) and the effective production rate (PE). Find initial average buffer levels, i.e., w_i $i = 1,\dots,k-1$ and estimate wip_i $i = 1,\dots,k-1$ by using the formula,

$$wip_i = \sum_{t=i}^{k-1} w_t \quad (i = 1,\dots,k-1) \tag{4.66}$$

- *Step 2* Adjust f_i^q $(i=1,...,k-1)$ $i=1,...,k-1$ by using the formula

$$f_i^q = \frac{\mu_i}{\frac{1-(1-x_i)^{wip_i}}{x_i} - wip_i(1-x_i)^{wip_i} + \frac{(1-x_i)^{wip_i}\left[(wip_i.(x_i+x_k)+1\right]}{x_i+x_k}} \quad (i=1,...,k-1) \quad (4.67)$$

where $x_i = \dfrac{f_i}{\mu_i}$ $i=1,...,k-1$ and $x_k = \dfrac{f_k - p_k}{\mu_k}$

- *Step 3* Calculate the new system yield $Y_{sys}^{new} = \prod_{i=1}^{k} \dfrac{f_i^q}{f_i^q + g_i}$

- *Step 4* Evaluate the system with f_i^q $i=1,...,k-1$ rates by adjusting the parameters $p'_i = \dfrac{f_i^q(p_i+g_i)}{(f_i^q+g_i)}$, $N'_i = N_i$ $i=1,...,k-1$, $p'_k = \dfrac{f_k(p_k+g_k)}{(f_k+g_k)}$ *and* $\mu_i' = \mu_i$, $r_i' = r_i$ $i=1,...,k$. Calculate the new total production rate (PT^{new}) and the new effective production rate (PE^{new}) by using the formula $PE^{new} = PT^{new} \times Y_{sys}^{new}$ and estimate new wip_i^{new}.

- *Step 5* If PT^{new}, PE^{new} and wip_i^{new} are sufficiently close to PT, PE and wip_i, then stop. Otherwise, set $PT = PT^{new}$, $PE = PE^{new}$ and $wip_i = wip_i^{new}$, go to *Step 2* and repeat the procedure.

4.2.1. The Production Systems with Rework Loop and Quality Information Feedback

In this section, we consider the production systems with rework loop (Figure 4.1) where both machine M_j (merge machine) and machine M_k (split machine) can detect bad quality features made by their upstream machines. We assume that machine M_j detect the bad parts made by machine M_i $i=1,...,j-1$ and may stop them with rate h_j, and machine M_k detect the bad parts made by machine M_i $i = j+1,...,k-1$ and may stop them with rate h_k.

The solution algorithm of this system is similar to the solution algorithm given in section 4.1.2 for the rework loop systems without quality information feedback. In Step 3 and in Step 4, we analyzed serial **Line 1** and serial **Line 2** by using the procedure given in section 4.1.1.2 If there is quality information feedback in the rework loop system, we analyze serial **Line 1** and serial **Line 2** (Step 3 and Step 4 in section 4.1.2) by using the evaluation technique of *k*-machines serial line with quality information feedback given in section 4.2 instead of the procedure given in section 4.1.1.2.

4.2.2. Accuracy of the Method

In the tables below, we report the results of the same 30 cases (Appendix A) showing the accuracy of the proposed solution technique by giving the expected values of the performance measures of the rework loop production systems with quality information feedback. These expected performance measures are the throughput rates of decomposed lines **Line 1**, **Line 2**, **Line 3**, and **Line 4**, and average work in processes. In Table 4.6, *PT1*, *PT2*, *PT3* and *PT4* denote the estimates of the production rates of serial **Line 1**, **Line 2**, **Line 3**, and **Line 4** respectively. The % errors in the production rates are calculated as in (4.56).

Table 4.6. Validation of *PT(i)* in rework loop system **with *qif***

Case Number	Throughput Rate	Simulation	Analytical	% Err
CASE 1	***PT1***	0.695	0.696	**0.15%**
	PT2	1.523	1.523	**-0.01%**
	PT3	0.695	0.696	**0.14%**
	PT4	0.828	0.827	**-0.14%**
CASE 2	***PT1***	0.462	0.456	**-1.17%**
	PT2	1.451	1.449	**-0.08%**
	PT3	0.462	0.456	**-1.16%**
	PT4	0.988	0.992	**0.42%**
CASE 3	***PT1***	0.652	0.651	**-0.18%**
	PT2	1.448	1.449	**0.08%**
	PT3	0.652	0.651	**-0.17%**
	PT4	0.795	0.797	**0.29%**
CASE 4	***PT1***	0.498	0.494	**-0.79%**
	PT2	1.633	1.633	**-0.04%**
	PT3	0.498	0.494	**-0.78%**
	PT4	1.135	1.138	**0.29%**
CASE 5	***PT1***	1.182	1.182	**-0.02%**
	PT2	1.778	1.777	**-0.02%**
	PT3	1.182	1.182	**-0.02%**
	PT4	0.595	0.595	**-0.02%**

Table 4.7. Validation of *PT(i)* in rework loop system **with *qif*** – continued

Case Number	Throughput Rate	Simulation	Analytical	% Err
CASE 6	***PT1***	0.615	0.622	**1.18%**
	PT2	1.400	1.399	**-0.04%**
	PT3	0.615	0.622	**1.17%**
	PT4	0.785	0.777	**-1.01%**
CASE 7	***PT1***	0.683	0.673	**-1.45%**
	PT2	1.399	1.398	**-0.07%**
	PT3	0.683	0.673	**-1.45%**
	PT4	0.715	0.724	**1.25%**
CASE 8	***PT1***	0.761	0.721	**-5.23%**
	PT2	2.123	2.079	**-2.05%**
	PT3	0.761	0.721	**-5.24%**
	PT4	1.362	1.358	**-0.28%**
CASE 9	***PT1***	0.510	0.509	**-0.07%**
	PT2	1.721	1.726	**0.25%**
	PT3	0.510	0.509	**-0.09%**
	PT4	1.211	1.216	**0.38%**
CASE 10	***PT1***	0.683	0.680	**-0.50%**
	PT2	1.388	1.387	**-0.05%**
	PT3	0.683	0.680	**-0.51%**
	PT4	0.704	0.707	**0.39%**
CASE 11	***PT1***	0.766	0.772	**0.70%**
	PT2	1.746	1.742	**-0.21%**
	PT3	0.766	0.772	**0.70%**
	PT4	0.979	0.970	**-0.92%**
CASE 12	***PT1***	0.750	0.745	**-0.66%**
	PT2	1.397	1.396	**-0.06%**
	PT3	0.750	0.745	**-0.66%**
	PT4	0.646	0.650	**0.64%**
CASE 13	***PT1***	0.757	0.745	**-1.58%**
	PT2	1.553	1.556	**0.15%**
	PT3	0.757	0.745	**-1.58%**
	PT4	0.795	0.810	**1.78%**
CASE 14	***PT1***	0.668	0.668	**-0.04%**
	PT2	1.520	1.520	**0.00%**
	PT3	0.668	0.667	**-0.06%**
	PT4	0.852	0.852	**0.04%**
CASE 15	***PT1***	0.781	0.777	**-0.40%**
	PT2	1.523	1.530	**0.44%**
	PT3	0.781	0.777	**-0.40%**
	PT4	0.742	0.752	**1.32%**

Table 4.8. Validation of *PT(i)* in rework loop system **with *qif*** – continued

Case Number	Throughput Rate	Simulation	Analytical	% Err
CASE 16	***PT1***	0.225	0.224	**-0.17%**
	PT2	0.739	0.735	**-0.45%**
	PT3	0.225	0.224	**-0.18%**
	PT4	0.514	0.511	**-0.56%**
CASE 17	***PT1***	0.251	0.253	**0.73%**
	PT2	0.728	0.728	**-0.07%**
	PT3	0.251	0.253	**0.72%**
	PT4	0.476	0.474	**-0.49%**
CASE 18	***PT1***	0.363	0.364	**0.30%**
	PT2	0.806	0.806	**-0.02%**
	PT3	0.363	0.364	**0.30%**
	PT4	0.443	0.442	**-0.28%**
CASE 19	***PT1***	1.492	1.495	**0.21%**
	PT2	2.432	2.439	**0.31%**
	PT3	1.492	1.495	**0.21%**
	PT4	0.939	0.943	**0.47%**
CASE 20	***PT1***	0.330	0.340	**2.92%**
	PT2	0.855	0.855	**0.00%**
	PT3	0.330	0.340	**2.91%**
	PT4	0.524	0.515	**-1.83%**
CASE 21	***PT1***	0.322	0.316	**-1.89%**
	PT2	0.988	0.976	**-1.22%**
	PT3	0.322	0.315	**-1.89%**
	PT4	0.666	0.660	**-0.89%**
CASE 22	***PT1***	0.545	0.592	**8.51%**
	PT2	1.453	1.557	**7.12%**
	PT3	0.545	0.592	**8.50%**
	PT4	0.907	0.964	**6.29%**
CASE 23	***PT1***	0.337	0.331	**-1.66%**
	PT2	3.193	3.186	**-0.21%**
	PT3	0.337	0.331	**-1.68%**
	PT4	2.856	2.855	**-0.05%**
CASE 24	***PT1***	0.190	0.190	**0.10%**
	PT2	0.688	0.692	**0.53%**
	PT3	0.190	0.190	**0.10%**
	PT4	0.497	0.501	**0.70%**
CASE 25	***PT1***	0.112	0.112	**-0.45%**
	PT2	0.687	0.687	**0.02%**
	PT3	0.112	0.112	**-0.49%**
	PT4	0.574	0.575	**0.12%**

Table 4.9. Validation of *PT(i)* in rework loop system **with *qif*** – continued

Case Number	Throughput Rate	Simulation	Analytical	% Err
CASE 26	***PT1***	1.755	1.778	**1.34%**
	PT2	3.531	3.576	**1.27%**
	PT3	1.755	1.778	**1.34%**
	PT4	1.775	1.797	**1.21%**
CASE 27	***PT1***	0.370	0.370	**0.07%**
	PT2	0.712	0.712	**-0.01%**
	PT3	0.370	0.370	**0.06%**
	PT4	0.342	0.341	**-0.09%**
CASE 28	***PT1***	0.489	0.488	**-0.17%**
	PT2	1.085	1.085	**0.01%**
	PT3	0.488	0.488	**-0.16%**
	PT4	0.596	0.597	**0.15%**
CASE 29	***PT1***	0.101	0.101	**-0.32%**
	PT2	0.868	0.868	**-0.02%**
	PT3	0.101	0.101	**-0.40%**
	PT4	0.767	0.767	**0.03%**
CASE 30	***PT1***	0.428	0.429	**0.09%**
	PT2	0.725	0.725	**0.00%**
	PT3	0.428	0.429	**0.09%**
	PT4	0.296	0.296	**-0.13%**

Table 4.10 shows the mean of the absolute percent differences of average inventories for 30 cases and Table 4.11 shows the *Inv % errors* of 11 buffers for the Case 28. We choose Case 28 since it has the highest absolute mean error among the all cases.

Table 4.10. The mean of the absolute percent differences of average inventories for rework loop systems with *qif*

CASE #	1	2	3	4	5	6	7	8	9	10
Mean Error	**1.00%**	**1.07%**	**0.58%**	**0.75%**	**0.81%**	**0.76%**	**0.45%**	**3.06%**	**1.18%**	**0.68%**
CASE #	*11*	*12*	*13*	*14*	*15*	*16*	*17*	*18*	*19*	*20*
Mean Error	**1.12%**	**0.83%**	**0.77%**	**0.74%**	**1.76%**	**1.38%**	**1.39%**	**0.50%**	**0.47%**	**3.53%**
CASE #	*21*	*22*	*23*	*24*	*25*	*26*	*27*	*28*	*29*	*30*
Mean Error	**1.50%**	**3.39%**	**1.24%**	**1.31%**	**2.01%**	**1.61%**	**0.84%**	**6.10%**	**5.82%**	**0.51%**

Table 4.11. *Inv % errors* of average buffer levels for rework loop systems with *qif* – Case 28

	CAPACITY	SIM.	ANALYTICAL	Err %
B1	40	39.945	39.933	**-0.03%**
B2	35	34.486	34.289	**-0.56%**
B3	50	49.812	48.631	**-2.36%**
B4	40	39.983	39.970	**-0.03%**
B5	40	23.329	30.472	**17.86%**
B6	30	24.528	28.712	**13.95%**
B7	25	7.139	2.869	**-17.08%**
B8	55	47.803	54.476	**12.13%**
B9	35	28.180	28.169	**-0.03%**
B10	25	0.261	0.353	**0.37%**
B11	45	2.506	3.715	**2.69%**

The average absolute value of the % errors in the throughput rate and the mean of the absolute percent differences of average inventories are %1.10 and %1.57 respectively.

4.3. The Production Systems with Multiple Loops

Rework systems can be generalized into multiple loops. In this section, we consider a production system with three rework loops as an illustration (Figure 4.9).

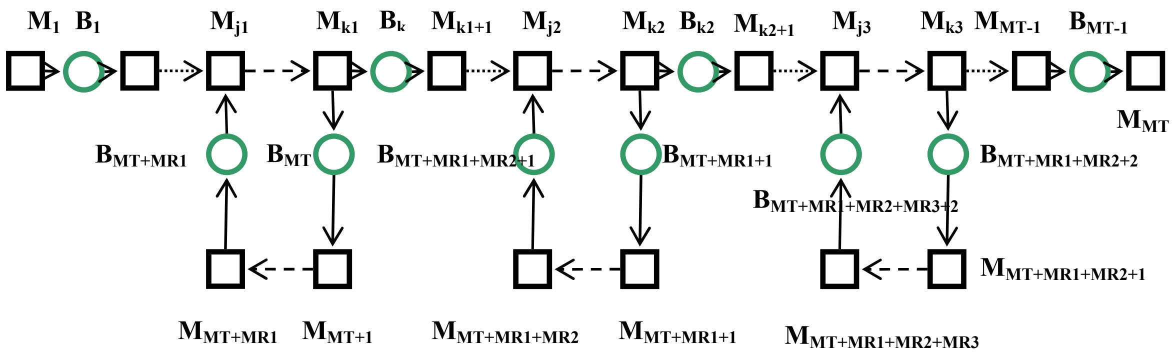

Figure 4.9. The production system with three rework loops

The overlapping decomposition method (Li, 2005) is used to estimate the throughput of this complex production in Figure 4.9. The idea of the method is to decompose the system into a couple of serial lines and modify the parameters of overlapping machines to accommodate the effects of other lines and to introduce a recursive procedure to approximate the system performance. Finally, the system performance can be calculated when the procedure converges.

The notation we use is given below:

MT : The number of machine in the main line.

$MR1$: The number of machine in the first rework line.

$MR2$: The number of machine in the second rework line.

$MR3$: The number of machine in the third rework line.

M_{j1} : The first merge machine in the production system.

M_{j2} : The second merge machine in the production system.

M_{j3} : The third merge machine in the production system.

M_{k1} : The first split machine in the production system.

M_{k2} : The second split machine in the production system.

M_{k3} : The third split machine in the production system.

$PT(i)$: Total production rate of Line i, $i = 1,...,10$.

Y_i : Yield of the Line i, $i = 1,...,6$

s_{k1} : Probability that machine M_{k1} is starved by B_{k1-1}.

s_{k2} : Probability that machine M_{k2} is starved by B_{k2-1}.

s_{k3} : Probability that machine M_{k3} is starved by B_{k3-1}.

b_{j1} : Probability that machine M_{j1} is blocked by B_{j1}.

b_{j2} : Probability that machine M_{j2} is blocked by B_{j2}.

b_{j3} : Probability that machine M_{j3} is blocked by B_{j3}.

s_{j1_1} : Probability that machine M_{j1} is starved by B_{j1-1}.

s_{j1_2} : Probability that machine M_{j1} is starved by B_{MT+MR1}.

s_{j2_1} : Probability that machine M_{j2} is starved by B_{j2-1}.

s_{j2_2} : Probability that machine M_{j2} is starved by $B_{MT+MR1+MR2+1}$.

s_{j3_1} : Probability that machine M_{j3} is starved by B_{j3-1}.

s_{j3_2} : Probability that machine M_{j3} is starved by $B_{MT+MR1+MR2+MR3+2}$.

b_{k1_1} : Probability that machine M_{k1} is blocked by B_{k1}.

b_{k1_2} : Probability that machine M_{k1} is blocked by B_{MT}.

b_{k2_1} : Probability that machine M_{k2} is blocked by B_{k2}.

b_{k2_2} : Probability that machine M_{k2} is blocked by $B_{MT+MR1+1}$.

b_{k3_1} : Probability that machine M_{k3} is blocked by B_{k3}.

b_{k3_2} : Probability that machine M_{k3} is blocked by $B_{MT+MR1+MR2+2}$.

α_1 : The probability that a part will be sent to first rework line by M_{k1}.

α_2 : The probability that a part will be sent to second rework line by M_{k2}.

α_3 : The probability that a part will be sent to third rework line by M_{k3}.

The assumptions of the model are as follows:

- Machine M_i is blocked at time t if the downstream buffer is full at time t (Machine M_{MT} is never blocked). In particular, machine M_{k1} is blocked by the B_{k1} if the part is non-defective and B_{k1} is full, while machine M_{k1} is blocked by B_{MT} if the part is defective and B_{MT} is full. Machine M_{k2} is blocked by the B_{k2} if the part is non-defective and B_{k2} is full, while machine M_{k2} is blocked by $B_{MT+MR1+1}$ if the part is defective and $B_{MT+MR1+1}$ is full. Machine M_{k3} is blocked by the B_{k3} if the part is non-defective and B_{k3} is full, while machine M_{k3} is blocked by $B_{MT+MR1+MR2+2}$ if the part is defective and $B_{MT+MR1+1}$ is full.
- Machine M_i is starved at time t if the upstream buffer is empty at time t (M_1 is never starved). M_{j1} is starved if both B_{j1-1} and B_{MT+MR1} are empty. M_{j2} is starved if both B_{j2-1} and $B_{MT+MR1+MR2+1}$ are empty. M_{j3} is starved if both B_{j3-1} and $B_{MT+MR1+MR2+MR3+2}$ are empty.
- After processing by machine M_{k1}, a part is defective with probability α_1, $0 < \alpha_1 < 1$, and needs to be repaired. This defective part is sent to B_{MT} if it is not full. The good part will be sent to B_{k1} with probability $1 - \alpha_1$ if B_{k1} is not full. After processing by machine M_{k2}, a part is defective with probability α_2, $0 < \alpha_2 < 1$, and this defective part is sent to $B_{MT+MR1+1}$ if it is not full. The good part will be sent to B_{k2} with probability $1 - \alpha_2$ if B_{k2} is not full. After processing by machine M_{k3}, a part is defective with probability α_3, $0 < \alpha_3 < 1$, and this defective part is sent to $B_{MT+MR1+MR2+2}$ if it is not full. The good part will be sent to B_{k3} with probability $1 - \alpha_3$ if B_{k2} is not full.
- Machine M_{j1} can take one part each cycle either from B_{j1-1} or B_{MT+MR1}. To avoid deadlock, M_{j1} always takes part from B_{MT+MR1} first if it is not empty. We consider the same assumption for other merge machines M_{j2} and M_{j3}.
- The rework lines' machines, merge machines, split machines and machines after M_{k3} have two states: Up and down, since they do not make any quality failures. Other machines have three states.

4.3.1. The Evaluation Procedure for the Production Systems with Three Rework Loop

We decompose the production system in Figure 4.9 by starting with the first machine M_1 where raw materials are supplied and going along with all the machines and buffers until a merge machine (M_{j1}, M_{j2} or M_{j3}) is met, which is the last machine of the rework lines. We continue to decompose the system beginning with this machine, visiting other machines and buffers until a split (M_{k1}, M_{k2} or M_{k3}) machine is met to construct another serial line. This process is repeated until all machines and buffers have been selected and finally a set of overlapped serial lines is obtained (Li, 2005). The analysis of these serial lines gives us the performance of the whole system.

While evaluating a serial transfer line, we always assume that the first and last machines are not starved and blocked, respectively. In the decomposed serial lines, if we know the probabilities that the first and last machines are starved and blocked respectively, we can introduce fictitious machines to accommodate these probabilities and use the solution algorithms in section 4.1.1.1 and 4.1.1.2 to calculate the throughput rates of the decomposed serial lines which gives us the throughput of the complete system.

The decomposed lines are as follows:

Line 1 : $M_1,...,M_{j1}$

Line 2 : $M_{j1},...,M_{k1}$

Line 3 : $M_{k1},...,M_{j2}$

Line 4 : $M_{j2},M_{j2+1},...,M_{k2}$

Line 5 : $M_{k2},M_{k2+1},...,M_{j3}$

Line 6 : $M_{j3},M_{j3+1},...,M_{k3}$

Line 7 : $M_{k3},M_{k3+1},...,M_{MT}$

Line 8 : $M_{k1},M_{MT+1},...,M_{MT+MR1},M_{j1}$

Line 9 : $M_{k2},M_{MT+MR1+1},...,M_{MT+MR1+Mr2},M_{j2}$

Line 10 : $M_{k3},.M_{MT+MR1+MR2+1},...,M_{MT+MR1+MR2+MR3},M_{j3}$

The following is the evaluation procedure of the system:

- <u>INITIALIZATION</u>

  $b_{j1} = 0.1,\ b_{j2} = 0.1,\ b_{j3} = 0.1;$

  $s_{k1} = 0.1,\ s_{k2} = 0.1,\ s_{k3} = 0.1;$

  $\alpha_1 = 0.1,\ \alpha_2 = 0.1,\ \alpha_3 = 0.1,$

- <u>ITERATIONS</u>

  Perform the steps until the convergence criterion is satisfied.

  - *Step 1* Introduce fictitious machines *M'k3* (first machine of Line 10) and *M'j3* (last machine of Line 10) with parameters

$$r'_{k3} = r_{k3}.\alpha_3.(1 - s_{k3}) \tag{4.68}$$

$$p'_{k3} = p_{k3} + r_{k3}.[1 - \alpha_3(1 - s_{k3})] \tag{4.69}$$

$$r'_{j3} = r_{j3}(1 - b_{j3}) \tag{4.70}$$

$$p'_{j3} = p_{j3} + r_{j3}.b_{j3} \tag{4.71}$$

Analyze Line 10 by using evaluation procedure in section 4.1.1.1 and find $PT(10)$, b_{k3_2}, s_{j3_2} and average buffer levels.

- *Step 2* Introduce fictitious machines *M'k2* (first machine of Line 9) and *M'j2* (last machine of Line 9) with parameters

$$r'_{k2} = r_{k2}.\alpha_2.(1 - s_{k2}) \tag{4.72}$$

$$p'_{k2} = p_{k2} + r_{k2}.[1 - \alpha_2(1 - s_{k2})] \tag{4.73}$$

$$r'_{j2} = r_{j2}(1 - b_{j2}) \tag{4.74}$$

$$p'_{j2} = p_{j2} + r_{j2}.b_{j2} \tag{4.75}$$

Analyze Line 9 by using evaluation procedure in section 4.1.1.1 and find $PT(9)$, b_{k2_2}, s_{j2_2} and average buffer levels.

- *Step 3* Introduce fictitious machines *M'k1* (first machine of Line 8) and *M'j1* (last machine of Line 8) with parameters

$$r'_{k1} = r_{k1}.\alpha_1.(1 - s_{k1}) \tag{4.76}$$

$$p'_{k1} = p_{k1} + r_{k1}.[1 - \alpha_1(1 - s_{k1})] \tag{4.77}$$

$$r'_{j1} = r_{j1}(1 - b_{j1}) \tag{4.78}$$

$$p'_{j1} = p_{j2} + r_{j1}.b_{j1} \tag{4.79}$$

Analyze Line 8 by using evaluation procedure in section 4.1.1.1, and find $PT(8)$, b_{k1_2}, s_{j1_2} and average buffer levels.

- *Step 4* Introduce fictitious machine *M''j1* (last machine of Line 1) with parameters

$$r''_{j1} = r_{j1} s_{j1\_2}(1 - b_{j1}) \tag{4.80}$$

$$p''_{j1} = p_{j1} + r_{j1}.(1 - s_{j1\_2}(1 - b_{j1})) \tag{4.81}$$

Note that we found s_{j1_2} found in Step 3. Analyze Line 1 by using evaluation procedure in section 4.1.1.2 and find $PT(1)$ (total production rate of Line 1), s_{j1_1}, Y_1, and average buffer levels.

- *Step 5* Introduce fictitious machines *M''k1* (first machine of Line 3) and *M''j2* (last machine of Line 3) with parameters

$$r''_{k1} = r_{k1}(1 - \alpha_1)(1 - s_{k1}) \tag{4.82}$$

$$p''_{k1} = p_{k1} + r_{k1}[1 - (1 - \alpha_1)(1 - s_{k1})] \tag{4.83}$$

$$r''_{j2} = r_{j2} s_{j2\_2}(1 - b_{j2}) \tag{4.84}$$

$$p''_{j2} = p_{j2} + r_{j2}(1 - s_{j2\_2}(1 - b_{j2})) \tag{4.85}$$

Note that we found s_{j2_2} in Step 2. Analyze Line 3 by using evaluation procedure in section 4.1.1.2 and find $PT(3)$, s_{j2_1}, b_{k1_1}, Y_3, and average buffer levels.

- *Step 6* Introduce fictitious machines M'''_{j1} (first machine of Line 2) and M'''_{k1} (last machine of Line 2) with parameters

$$r'''_{j1} = r_{j1}\left(1 - s_{j1\_1}s_{j1\_2}\right) \tag{4.86}$$

$$p'''_{j1} = p_{j1} + r_{j1}s_{j1\_1}s_{j1\_2} \tag{4.87}$$

$$r'''_{k1} = r_{k1}\left(1 - \alpha_1 b_{k1\_2} - (1-\alpha_1)b_{k1\_1}\right) \tag{4.88}$$

$$p'''_{k1} = p_{k1} + r_{k1}\left(\alpha_1 b_{k1\_2} + (1-\alpha_1)b_{k1\_1}\right) \tag{4.89}$$

Note that we found b_{k1_1} in Step 5, b_{k1_2} and s_{j1_2} in Step 3, and s_{j1_1} in Step 4. Analyze Line 2 by using evaluation procedure in section 4.1.1.2 and find $PT(2)$, s_{k1}, b_{j1}, Y_2, and average buffer levels.

- *Step 7* Introduce fictitious machines M''_{k2} (first machine of Line 5) and M''_{j3} (last machine of Line 5) with parameters

$$r''_{k2} = r_{k2}(1-\alpha_2)(1-s_{k2}) \tag{4.90}$$

$$p''_{k2} = p_{k2} + r_{k2}\left[1 - (1-\alpha_2)(1-s_{k2})\right] \tag{4.91}$$

$$r''_{j3} = r_{j3}s_{j3\_2}\left(1 - b_{j3}\right) \tag{4.92}$$

$$p''_{j3} = p_{j3} + r_{j3}\left(1 - s_{j3\_2}\left(1 - b_{j3}\right)\right) \tag{4.93}$$

Note that we found s_{j3_2} in Step 1. Analyze Line 5 by using evaluation procedure in section 4.1.1.2 and find $PT(5)$, b_{k2_1}, s_{j3_1}, Y_5, and average buffer levels.

- *Step 8* Introduce fictitious machines M'''_{j2} (first machine of Line 4) and M'''_{k2} (last machine of Line 4) with parameters

$$r'''_{j2} = r_{j2}\left(1 - s_{j2\_1}s_{j2\_2}\right) \tag{4.94}$$

$$p'''_{j2} = p_{j2} + r_{j2}s_{j2\_1}s_{j2\_2} \tag{4.95}$$

$$r'''_{k2} = r_{k2}\left(1 - \alpha_2 b_{k2\_2} - (1-\alpha_2) b_{k2\_1}\right) \tag{4.96}$$

$$p'''_{k2} = p_{k2} + r_{k2}\left(\alpha_2 b_{k2\_2} + (1-\alpha_2) b_{k2\_1}\right) \tag{4.97}$$

Note that we found b_{k2_1} in Step 7, b_{k2_2} and s_{j2_2} in Step 2, and s_{j2_1} in Step 5. Analyze Line 4 by using evaluation procedure in section 4.1.1.2 and find $PT(4)$, s_{k2}, b_{j2}, Y_4, and average buffer levels.

- *Step 9* Introduce fictitious machine M''_{k3} (first machine of Line 7) with parameters

$$r''_{k3} = r_{k3}(1-\alpha_3)(1-s_{k3}) \tag{4.98}$$

$$p''_{k3} = p_{k3} + r_{k3}.[1-(1-\alpha_3)(1-s_{k3})] \tag{4.99}$$

Analyze Line 7 by using evaluation procedure in section 4.1.1.1 and find $PT(7)$ (total production rate of Line 1), b_{k3_1}, and average buffer levels.

- *Step 10* Introduce fictitious machines M'''_{j3} (first machine of Line 6) and M'''_{k3} (last machine of Line 6) with parameters

$$r'''_{j3} = r_{j3}\left(1 - s_{j3\_1} s_{j3\_2}\right) \tag{4.100}$$

$$p'''_{j3} = p_{j3} + r_{j3} s_{j3\_1} s_{j3\_2} \tag{4.101}$$

$$r'''_{k3} = r_{k3}\left(1 - \alpha_3 b_{k3\_2} - (1-\alpha_3) b_{k3\_1}\right) \tag{4.102}$$

$$p'''_{k3} = p_{k3} + r_{k3}\left(\alpha_3 b_{k3\_2} + (1-\alpha_3) b_{k3\_1}\right) \tag{4.103}$$

Note that we found b_{k3_1} in Step 9, b_{k3_2} and s_{j3_2} in Step 1, and s_{j3_1} in Step 7. Analyze Line 6 by using evaluation procedure in section 4.1.1.2 and find $PT(6)$, s_{k3}, b_{j3}, Y_6, and average buffer levels.

- *Step 11* Estimate the new rework rates α_1, α_2, *and* α_3, by using the formulas

$$\alpha_1 = \frac{(1-Y_1.Y_2).(1-b_{k1\_1})}{Y_2.(1-b_{k1\_2})+(1-Y_1.Y_2).(1-b_{k1\_1})} \tag{4.104}$$

$$\alpha_2 = \frac{(1-Y_3.Y_4).(1-b_{k2\_1})}{Y_4.(1-b_{k2\_2})+(1-Y_3.Y_4).(1-b_{k2\_1})} \tag{4.105}$$

$$\alpha_3 = \frac{(1-Y_5.Y_6).(1-b_{k3\_1})}{Y_6.(1-b_{k3\_2})+(1-Y_5.Y_6).(1-b_{k3\_1})} \tag{4.106}$$

- *Step12* Go back to Step 1, use new α_1, α_2, α_3, b_{j1}, b_{j2}, b_{j3}, s_{k1}, s_{k2} *,and* s_{k3} values for the next iteration, and perform the steps 1 to 11 until the convergence criteria is satisfied. The production rate of the system is equal to *PT(1)* (or *PT(7)*) .

- THE CONVERGENCE CRITERIA: Stop the procedure when $|PT(7)-PT(1)|$, $|PT(3)-PT(1)|$, $|(PT(2)-PT(8))-PT(1)|$, and $|(PT(4)-PT(9))-PT(5)|$ are smaller than a pre-defined small number ε .

4.3.2. Accuracy of the Method

We generated new 30 cases for the validation of multiple loop solution technique. The 30 cases are in Appendix B. In all cases, there are three rework lines. The main line consists of 14 machines and each rework line consists of one machine. There are 19 buffers in the system. In all cases the third machine in the main line is the first merge machine, the seventh machine in the main line is the second merge machine, the eleventh machine in the main line is the third merge machine, the fifth machine in the main line is the first merge machine, the ninth machine in the main line is the second split machine and the thirteenth machine in the main line is the third split machine. In the tables below, the accuracy of the proposed solution technique of the multiple loops production system is reported. In Table 4.11, *PT1*, *PT2*, *PT3*, *PT4*, *PT5*, *PT6*, *PT7*, *PT8*, *PT9*, and *PT10* denote the estimates of the production rates of the decomposed serial lines 1 to 10 respectively. The % errors in the production rates are calculated from (4.56).

Table 4.12. Validation of throughput rates of multiple loops

Case	Pro. Rate	Simulation	Analytical	% Err	Case	Pro. Rate	Simulation	Analytical	% Err
1	PT1	0.7398	0.7399	0.01%	5	PT1	0.6536	0.6515	-0.33%
	PT2	1.3482	1.3488	0.05%		PT2	1.7991	1.7994	0.01%
	PT3	0.7398	0.7399	0.01%		PT3	0.6537	0.6515	-0.33%
	PT4	1.5316	1.5327	0.07%		PT4	1.1152	1.0936	-1.94%
	PT5	0.7398	0.7399	0.01%		PT5	0.6537	0.6515	-0.33%
	PT6	1.5049	1.4587	-3.07%		PT6	1.0907	1.0880	-0.24%
	PT7	0.7398	0.7400	0.02%		PT7	0.6537	0.6516	-0.32%
	PT8	0.6084	0.6089	0.09%		PT8	1.1454	1.1479	0.22%
	PT9	0.7918	0.7928	0.12%		PT9	0.4616	0.4421	-4.23%
	PT10	0.7651	0.7188	-6.05%		PT10	0.4371	0.4366	-0.13%
2	PT1	0.6883	0.6993	1.59%	6	PT1	1.1567	1.1636	0.60%
	PT2	1.1952	1.1955	0.02%		PT2	1.7462	1.7591	0.74%
	PT3	0.6883	0.6993	1.59%		PT3	1.1568	1.1636	0.59%
	PT4	0.9128	0.9246	1.29%		PT4	1.5472	1.5561	0.58%
	PT5	0.6883	0.6993	1.59%		PT5	1.1568	1.1636	0.59%
	PT6	1.4673	1.4601	-0.49%		PT6	1.8568	1.8617	0.27%
	PT7	0.6883	0.6994	1.61%		PT7	1.1569	1.1636	0.58%
	PT8	0.5069	0.4962	-2.11%		PT8	0.5894	0.5955	1.03%
	PT9	0.2245	0.2253	0.38%		PT9	0.3904	0.3925	0.53%
	PT10	0.7790	0.7608	-2.33%		PT10	0.7000	0.6982	-0.26%
3	PT1	0.9678	0.9690	0.12%	7	PT1	0.6783	0.6747	-0.54%
	PT2	1.6384	1.6392	0.05%		PT2	1.3457	1.3405	-0.38%
	PT3	0.9678	0.9690	0.13%		PT3	0.6783	0.6747	-0.53%
	PT4	1.1318	1.1329	0.10%		PT4	0.7148	0.7103	-0.64%
	PT5	0.9677	0.9690	0.14%		PT5	0.6783	0.6747	-0.54%
	PT6	2.6227	2.6244	0.06%		PT6	1.1466	1.1380	-0.75%
	PT7	0.9677	0.9690	0.14%		PT7	0.6783	0.6747	-0.52%
	PT8	0.6706	0.6702	-0.06%		PT8	0.6674	0.6659	-0.23%
	PT9	0.1642	0.1639	-0.17%		PT9	0.0366	0.0356	-2.61%
	PT10	1.6550	1.6554	0.02%		PT10	0.4684	0.4633	-1.09%
4	PT1	0.7009	0.7014	0.07%	8	PT1	0.6884	0.6831	-0.77%
	PT2	1.3466	1.3469	0.03%		PT2	1.5604	1.5474	-0.83%
	PT3	0.7009	0.7014	0.07%		PT3	0.6884	0.6831	-0.77%
	PT4	1.4638	1.4095	-3.71%		PT4	0.9884	0.9715	-1.72%
	PT5	0.7009	0.7014	0.07%		PT5	0.6884	0.6831	-0.78%
	PT6	1.3426	1.3327	-0.74%		PT6	1.1516	1.1418	-0.85%
	PT7	0.7009	0.7014	0.08%		PT7	0.6885	0.6831	-0.77%
	PT8	0.6457	0.6456	-0.02%		PT8	0.8721	0.8644	-0.89%
	PT9	0.7630	0.7081	-7.19%		PT9	0.3000	0.2884	-3.88%
	PT10	0.6418	0.6313	-1.63%		PT10	0.4631	0.4587	-0.96%

Table 4.13. Validation of throughput rates of multiple loops – continued

Case	Pro. Rate	Simulation	Analytical	% Err	Case	Pro. Rate	Simulation	Analytical	% Err
9	PT1	0.9646	0.9657	0.11%	13	PT1	0.6198	0.6139	-0.96%
	PT2	2.4276	2.4155	-0.50%		PT2	1.4442	1.4286	-1.08%
	PT3	0.9647	0.9656	0.10%		PT3	0.6197	0.6138	-0.95%
	PT4	1.1046	1.1060	0.13%		PT4	0.9604	0.9502	-1.07%
	PT5	0.9647	0.9656	0.10%		PT5	0.6196	0.6138	-0.94%
	PT6	2.8968	2.8993	0.09%		PT6	1.1773	1.1663	-0.93%
	PT7	0.9647	0.9656	0.10%		PT7	0.6196	0.6138	-0.93%
	PT8	1.4629	1.4499	-0.89%		PT8	0.8244	0.8148	-1.17%
	PT9	0.1400	0.1404	0.24%		PT9	0.3408	0.3364	-1.29%
	PT10	1.9321	1.9337	0.08%		PT10	0.5577	0.5525	-0.95%
10	PT1	0.6324	0.6368	0.70%	14	PT1	0.6807	0.6814	0.11%
	PT2	1.3983	1.4100	0.84%		PT2	1.5620	1.5622	0.01%
	PT3	0.6324	0.6368	0.69%		PT3	0.6807	0.6814	0.10%
	PT4	0.6761	0.6808	0.70%		PT4	0.8929	0.8859	-0.79%
	PT5	0.6324	0.6368	0.69%		PT5	0.6807	0.6814	0.10%
	PT6	1.0773	1.0768	-0.04%		PT6	1.4030	1.3629	-2.86%
	PT7	0.6324	0.6368	0.69%		PT7	0.6807	0.6815	0.11%
	PT8	0.7660	0.7733	0.95%		PT8	0.8814	0.8807	-0.07%
	PT9	0.0437	0.0440	0.71%		PT9	0.2122	0.2044	-3.66%
	PT10	0.4449	0.4401	-1.09%		PT10	0.7223	0.6815	-5.65%
11	PT1	0.6159	0.6164	0.08%	15	PT1	0.4974	0.4972	-0.03%
	PT2	1.2297	1.2293	-0.03%		PT2	1.0915	1.0907	-0.07%
	PT3	0.6159	0.6164	0.08%		PT3	0.4974	0.4972	-0.04%
	PT4	0.8251	0.8237	-0.17%		PT4	0.7261	0.7182	-1.08%
	PT5	0.6159	0.6164	0.08%		PT5	0.4974	0.4972	-0.03%
	PT6	1.0428	1.0274	-1.48%		PT6	0.5635	0.5623	-0.20%
	PT7	0.6159	0.6165	0.09%		PT7	0.4974	0.4973	-0.03%
	PT8	0.6138	0.6129	-0.15%		PT8	0.5941	0.5935	-0.10%
	PT9	0.2092	0.2073	-0.91%		PT9	0.2287	0.2210	-3.36%
	PT10	0.4269	0.4110	-3.73%		PT10	0.0661	0.0651	-1.44%
12	PT1	0.7784	0.7880	1.24%	16	PT1	0.8558	0.8626	0.79%
	PT2	1.3286	1.3468	1.37%		PT2	1.7802	1.7861	0.33%
	PT3	0.7785	0.7881	1.23%		PT3	0.8558	0.8626	0.79%
	PT4	1.1957	1.2000	0.36%		PT4	1.1692	1.1764	0.61%
	PT5	0.7785	0.7881	1.23%		PT5	0.8559	0.8626	0.78%
	PT6	1.4756	1.4890	0.91%		PT6	1.7219	1.6841	-2.19%
	PT7	0.7784	0.7881	1.24%		PT7	0.8558	0.8626	0.80%
	PT8	0.5503	0.5588	1.54%		PT8	0.9245	0.9236	-0.10%
	PT9	0.4173	0.4119	-1.29%		PT9	0.3134	0.3138	0.13%
	PT10	0.6973	0.7009	0.52%		PT10	0.8661	0.8216	-5.14%

Table 4.14. Validation of throughput rates of multiple loops – continued

Case	Pro. Rate	Simulation	Analytical	% Err	Case	Pro. Rate	Simulation	Analytical	% Err
17	PT1	0.8964	0.9152	2.09%	21	PT1	0.8342	0.8430	1.05%
	PT2	1.4797	1.5100	2.05%		PT2	1.6323	1.6449	0.77%
	PT3	0.8964	0.9151	2.09%		PT3	0.8342	0.8430	1.06%
	PT4	1.3417	1.3727	2.31%		PT4	1.6096	1.6157	0.38%
	PT5	0.8964	0.9151	2.09%		PT5	0.8342	0.8430	1.06%
	PT6	1.8169	1.8194	0.14%		PT6	1.3146	1.3275	0.98%
	PT7	0.8964	0.9151	2.09%		PT7	0.8342	0.8430	1.06%
	PT8	0.5833	0.5949	1.97%		PT8	0.7981	0.8019	0.48%
	PT9	0.4453	0.4576	2.76%		PT9	0.7755	0.7727	-0.35%
	PT10	0.9205	0.9042	-1.77%		PT10	0.4805	0.4845	0.83%
18	PT1	0.5201	0.6075	16.81%	22	PT1	1.8475	1.8491	0.09%
	PT2	1.6151	1.6152	0.00%		PT2	3.2319	3.2212	-0.33%
	PT3	0.5200	0.6075	16.81%		PT3	1.8475	1.8491	0.09%
	PT4	0.5823	0.6794	16.68%		PT4	1.8970	1.8988	0.09%
	PT5	0.5201	0.6075	16.80%		PT5	1.8475	1.8491	0.08%
	PT6	0.7292	0.8488	16.40%		PT6	2.3597	2.3580	-0.07%
	PT7	0.5201	0.6075	16.81%		PT7	1.8476	1.8491	0.08%
	PT8	1.0951	1.0077	-7.98%		PT8	1.3844	1.3721	-0.89%
	PT9	0.0623	0.0720	15.60%		PT9	0.0494	0.0497	0.57%
	PT10	0.2091	0.2413	15.39%		PT10	0.5121	0.5089	-0.62%
19	PT1	0.8639	0.8623	-0.18%	23	PT1	0.5924	0.5935	0.19%
	PT2	1.6806	1.6760	-0.27%		PT2	1.4588	1.4625	0.25%
	PT3	0.8639	0.8623	-0.18%		PT3	0.5923	0.5935	0.19%
	PT4	1.1073	1.1025	-0.44%		PT4	1.2998	1.3000	0.01%
	PT5	0.8639	0.8623	-0.18%		PT5	0.5923	0.5935	0.20%
	PT6	1.1519	1.1456	-0.55%		PT6	1.0351	1.0138	-2.05%
	PT7	0.8638	0.8623	-0.17%		PT7	0.5923	0.5935	0.20%
	PT8	0.8168	0.8137	-0.38%		PT8	0.8665	0.8690	0.29%
	PT9	0.2435	0.2402	-1.35%		PT9	0.7075	0.7065	-0.14%
	PT10	0.2881	0.2833	-1.65%		PT10	0.4429	0.4204	-5.08%
20	PT1	0.7284	0.7387	1.41%	24	PT1	0.5913	0.6110	3.33%
	PT2	0.8485	0.8621	1.61%		PT2	1.4088	1.4526	3.11%
	PT3	0.7284	0.7387	1.41%		PT3	0.5913	0.6110	3.34%
	PT4	1.4826	1.4773	-0.36%		PT4	1.2984	1.3238	1.96%
	PT5	0.7284	0.7386	1.41%		PT5	0.5913	0.6110	3.33%
	PT6	1.2797	1.2956	1.24%		PT6	0.9410	0.9590	1.91%
	PT7	0.7284	0.7387	1.42%		PT7	0.5913	0.6111	3.35%
	PT8	0.1201	0.1235	2.79%		PT8	0.8176	0.8416	2.95%
	PT9	0.7543	0.7386	-2.07%		PT9	0.7071	0.7128	0.81%
	PT10	0.5514	0.5570	1.01%		PT10	0.3497	0.3480	-0.50%

Table 4.15. Validation of throughput rates of multiple loops – continued

Case	Pro. Rate	Simulation	Analytical	% Err	Case	Pro. Rate	Simulation	Analytical	% Err
25	***PT1***	0.5091	0.5159	**1.34%**	28	***PT1***	0.7182	0.7621	**6.12%**
	PT2	0.9542	0.9657	**1.21%**		***PT2***	1.1214	1.1903	**6.15%**
	PT3	0.5091	0.5159	**1.34%**		***PT3***	0.7181	0.7621	**6.12%**
	PT4	0.7259	0.7352	**1.28%**		***PT4***	0.8424	0.8942	**6.14%**
	PT5	0.5091	0.5159	**1.34%**		***PT5***	0.7181	0.7621	**6.12%**
	PT6	1.3733	1.3731	**-0.01%**		***PT6***	1.4597	1.4697	**0.69%**
	PT7	0.5091	0.5159	**1.33%**		***PT7***	0.7181	0.7621	**6.12%**
	PT8	0.4451	0.4498	**1.06%**		***PT8***	0.4033	0.4282	**6.19%**
	PT9	0.2168	0.2193	**1.13%**		***PT9***	0.1243	0.1321	**6.24%**
	PT10	0.8642	0.8572	**-0.80%**		***PT10***	0.7416	0.7076	**-4.58%**
26	***PT1***	0.7545	0.7545	**0.00%**	29	***PT1***	0.2756	0.2762	**0.24%**
	PT2	1.2545	1.2557	**0.10%**		***PT2***	0.6901	0.6909	**0.12%**
	PT3	0.7545	0.7545	**0.00%**		***PT3***	0.2756	0.2762	**0.21%**
	PT4	0.9623	0.9620	**-0.04%**		***PT4***	0.3155	0.3164	**0.28%**
	PT5	0.7544	0.7545	**0.01%**		***PT5***	0.2757	0.2762	**0.20%**
	PT6	1.4008	1.4000	**-0.06%**		***PT6***	0.8293	0.8293	**0.00%**
	PT7	0.7544	0.7545	**0.02%**		***PT7***	0.2757	0.2762	**0.20%**
	PT8	0.5000	0.5012	**0.23%**		***PT8***	0.4146	0.4147	**0.04%**
	PT9	0.2079	0.2075	**-0.19%**		***PT9***	0.0398	0.0402	**0.83%**
	PT10	0.6464	0.6455	**-0.14%**		***PT10***	0.5537	0.5531	**-0.09%**
27	***PT1***	0.4714	0.5233	**11.00%**	30	***PT1***	0.2370	0.2371	**0.02%**
	PT2	0.8502	0.9384	**10.37%**		***PT2***	0.5773	0.5769	**-0.08%**
	PT3	0.4714	0.5233	**10.99%**		***PT3***	0.2370	0.2371	**0.03%**
	PT4	0.5519	0.6116	**10.82%**		***PT4***	0.3392	0.3393	**0.04%**
	PT5	0.4714	0.5233	**10.99%**		***PT5***	0.2370	0.2371	**0.02%**
	PT6	1.5006	1.5000	**-0.04%**		***PT6***	0.5053	0.5043	**-0.18%**
	PT7	0.4714	0.5233	**10.99%**		***PT7***	0.2370	0.2371	**0.08%**
	PT8	0.3788	0.4151	**9.58%**		***PT8***	0.3403	0.3398	**-0.15%**
	PT9	0.0805	0.0884	**9.80%**		***PT9***	0.1021	0.1022	**0.06%**
	PT10	1.0291	0.9767	**-5.09%**		***PT10***	0.2683	0.2673	**-0.39%**

Table 4.16 shows the mean of the absolute percent differences of average inventories for 30 cases and Table 4.17 shows the *Inv % errors* of 11 buffers for the Case 27. We choose Case 27 since it has the highest absolute mean error among the all cases.

Table 4.16 The mean of the absolute percent differences of average inventories for multiple loop

CASE #	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*9*	*10*
Mean Error	**2.19%**	**2.09%**	**2.01%**	**2.95%**	**1.32%**	**4.58%**	**1.55%**	**2.66%**	**5.32%**	**3.23%**
CASE #	*11*	*12*	*13*	*14*	*15*	*16*	*17*	*18*	*19*	*20*
Mean Error	**1.79%**	**3.17%**	**5.50%**	**1.72%**	**1.14%**	**2.25%**	**4.84%**	**0.90%**	**3.21%**	**1.66%**
CASE #	*21*	*22*	*23*	*24*	*25*	*26*	*27*	*28*	*29*	*30*
Mean Error	**4.61%**	**1.60%**	**2.97%**	**5.00%**	**2.28%**	**3.44%**	**5.97%**	**1.68%**	**1.95%**	**1.19%**

Table 4.17. *Inv % errors* of average buffer levels for multiple loop systems – Case 27

	CAPACITY	*SIM.*	*ANALYTICAL*	*Err %*
1	50	49.589	49.583	-0.01%
2	25	24.933	24.923	-0.04%
3	10	7.577	8.420	8.43%
4	20	14.939	15.393	2.27%
5	10	8.616	9.495	8.79%
6	20	19.315	18.624	-3.45%
7	40	25.846	39.799	34.88%
8	20	19.956	19.944	-0.06%
9	25	20.482	24.565	16.33%
10	45	40.957	44.931	8.83%
11	30	0.585	0.701	0.39%
12	10	0.652	0.032	-6.20%
13	35	0.108	1.920	5.18%
14	55	0.386	0.140	-0.45%
15	50	0.027	2.679	5.30%
16	30	0.001	0.029	0.10%
17	10	0.070	0.459	3.88%
18	55	0.06	0.863	1.46%
19	30	1.157	5.136	13.26%

5. NUMERICAL RESULTS

In this section, in order to gain some insights into effects of *rework line*, *Quality Information Feedback* (QIF) as well as the effect of *the number of rework lines* on the production systems, we perform a numerical analysis. Firstly, we investigate the effect of rework line on the performance of the transfer line. For this purpose, we compare the performance of three production systems which are serial transfer line with QIF, rework loop system without QIF and rework loop system with QIF. Secondly, we investigate the effect of QIF on the performance of rework loop system. Finally, we compare the rework loop systems with single loop to multiple loops.

5.1. Effect of the Rework Line on the Production Systems Having Random Yield

In this section, we observe the effect of the rework line on the performance of a serial transfer line. We take into account two cases and carry out 10 experiments for each case. We consider a serial transfer line where machines produce less defective-part in the first case. The machines in this production system have identical quality failure rates as i.e. $g_i = 0.01, i = 1,2,3,5,6$. In the second case, we consider a serial transfer line in which machines produce defective-parts frequently and machines have identical quality failure rates which are high, i.e. $g_i = 0.45$, $i = 1,2,3,5,6$.

In each experiment, we take three different models into account. The first model considered is a serial transfer line with QIF, the second one is a rework loop system without QIF, and the last one is a rework loop system with QIF. In all models, the serial (transfer) line has 8 machines and the rework loop has 3 machines. For the rework loop system, the fourth machine of the serial is the merge machine and the seventh machine is the split machine. In the experiments, the service rates of all machines in the rework line, i.e. μ_i, $i = 9,10,11$, are increased from 1 to 10 for the two cases while other parameters are held constant. The service rates of rework machines are identical since we want to prevent a bottleneck effect of a machine in the rework line. We change the service rates of the machines in rework line since we can see the effect of the rework line clearly by changing the rework machines' service rates.

- ***Case 1:*** The parameters used in the experiments are in Table 5.1.

Table 5.1. The machine parameters used in the first case of section 5.1

	Machine	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	3.000	0.900	0.150	0.100	0.010	40
	2	3.000	0.900	0.150	0.100	0.010	40
	3	3.000	0.900	0.150	0.100	0.010	40
	4	**3.000**	**0.900**	**0.450**	**0.010**	**0.000**	**40**
	5	3.000	0.900	0.150	0.100	0.010	40
	6	3.000	0.900	0.150	0.100	0.010	40
	7	**3.000**	**0.900**	**0.450**	**0.010**	**0.000**	**40**
	8	3.000	0.900		0.100		40
REWORK LINE	*9*		0.950		0.100		40
	10		0.950		0.100		40
	11		0.950		0.100		40

Figures 5.1 shows the effect of rework machines' service rates for the three models considered when the yield of the main line is high. Here, the total and the effective throughput rates of the serial line with QIF are given as "Main Line's Total. Pro. Rate" and "Main Line's Eff. Pro. Rate" while the throughput rate of the rework loop system without QIF and with QIF are represented as "Without QIF" and "With QIF" respectively.

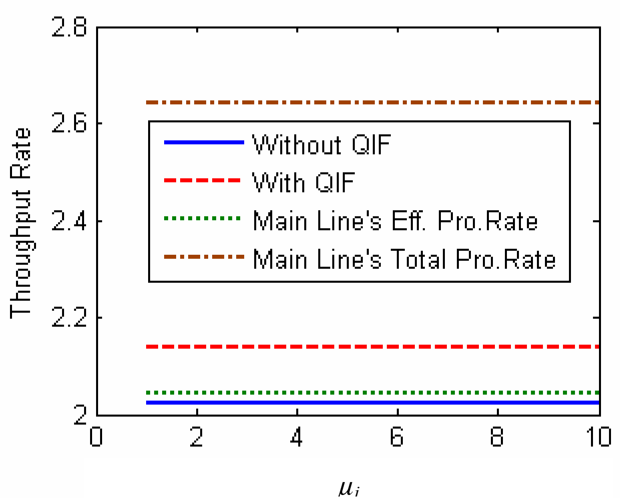

Figure 5.1. Effect of the rework line when the rework machines service rates are increased in the case of high yield

We also compare the throughput rates of decomposed serial Line 2 of the rework loop systems ("With QIF" and "Without QIF") to the total throughput rate of the serial line ("Main Line") with QIF which is shown in Figure 5.2.

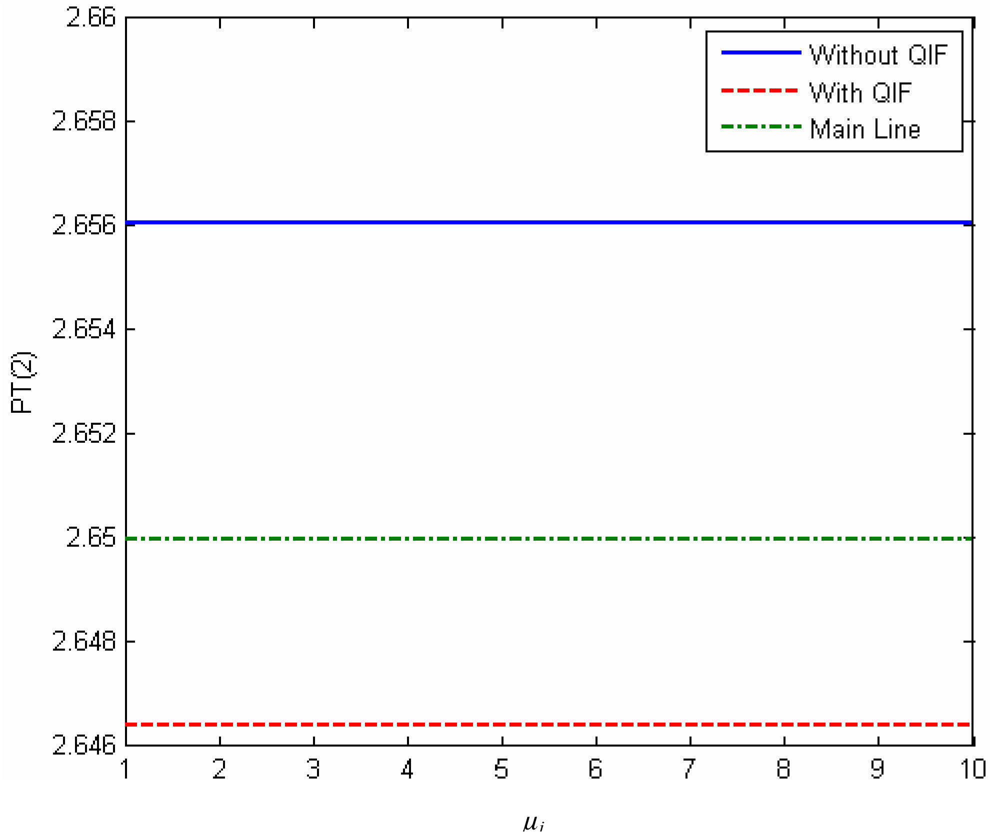

Figure 5.2. Comparing *PT(2)* of the rework systems to total throughput of the serial transfer line with QIF in the case of high yield

- ***<u>Case 2:</u>*** The parameters used in the experiments are in Table 5.2.

In the second case, we consider a serial line in which machines produce more defective-parts than the first case. Machines in the serial line have identical quality failure rates, i.e. $g_i = 0.45$, $i = 1,2,3,5,6$. Figures 5.3 shows the effect of rework machines' service rates for the three models considered when the yield of the main line is low. Here, the total and the effective throughput rates of the serial line with QIF are given as "Main Line's Total. Pro. Rate" and "Main Line's Eff. Pro. Rate" while the throughput rates of the rework loop system without QIF and with QIF are represented as "Without QIF" and "With QIF" respectively.

Table 5.2. The machine parameters used in the second case of section 5.1

	Machine	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	3.000	0.900	0.150	0.100	0.450	40
	2	3.000	0.900	0.150	0.100	0.450	40
	3	3.000	0.900	0.150	0.100	0.450	40
	4	**3.000**	**0.900**	**0.450**	**0.010**	**0.000**	**40**
	5	3.000	0.900	0.150	0.100	0.450	40
	6	3.000	0.900	0.150	0.100	0.450	40
	7	**3.000**	**0.900**	**0.450**	**0.010**	**0.000**	**40**
	8	3.000	0.900		0.100		40
REWORK LINE	*9*		0.950		0.100		40
	10		0.950		0.100		40
	11		0.950		0.100		40

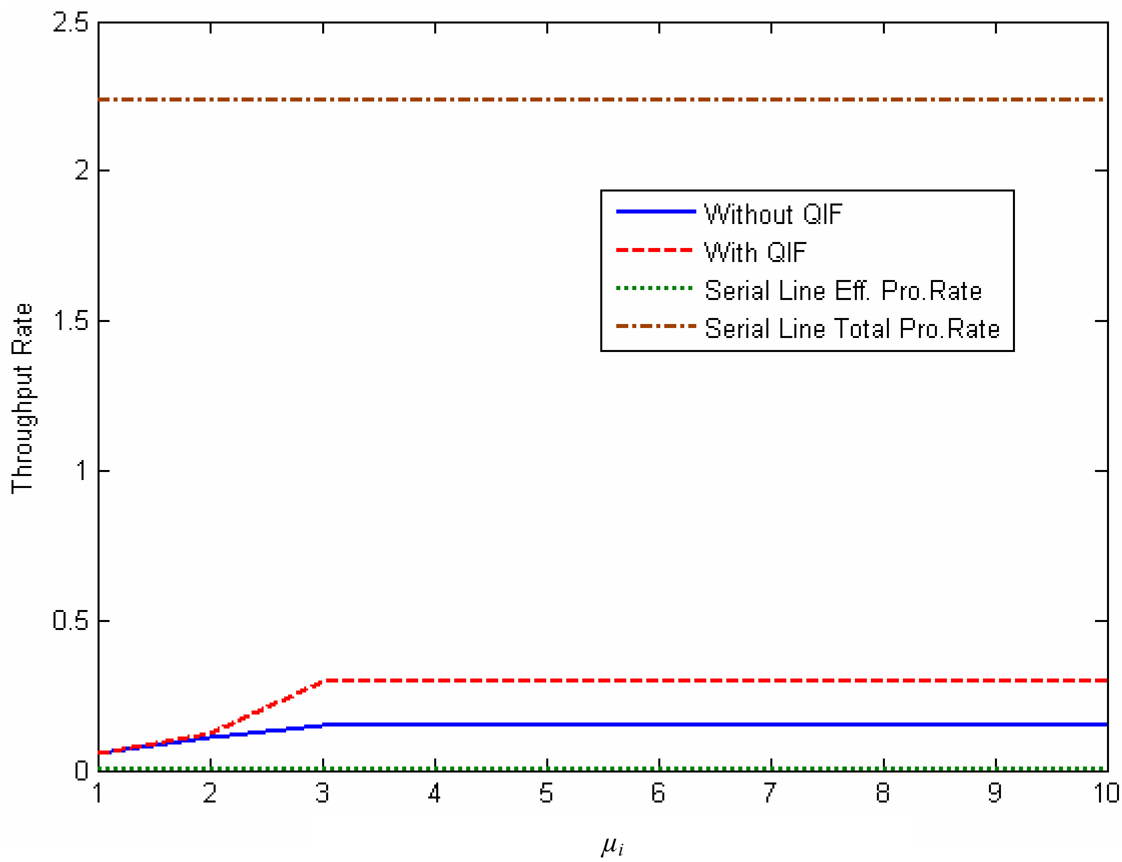

Figure 5.3. Effect of the rework line when the rework machines service rates are increased in the case of low yield

We also compare the throughput rates of decomposed serial Line 2 of the rework loop systems ("With QIF" and "Without QIF") to the total throughout rate of the serial line ("Main Line") with QIF which is shown in Figure 5.4.

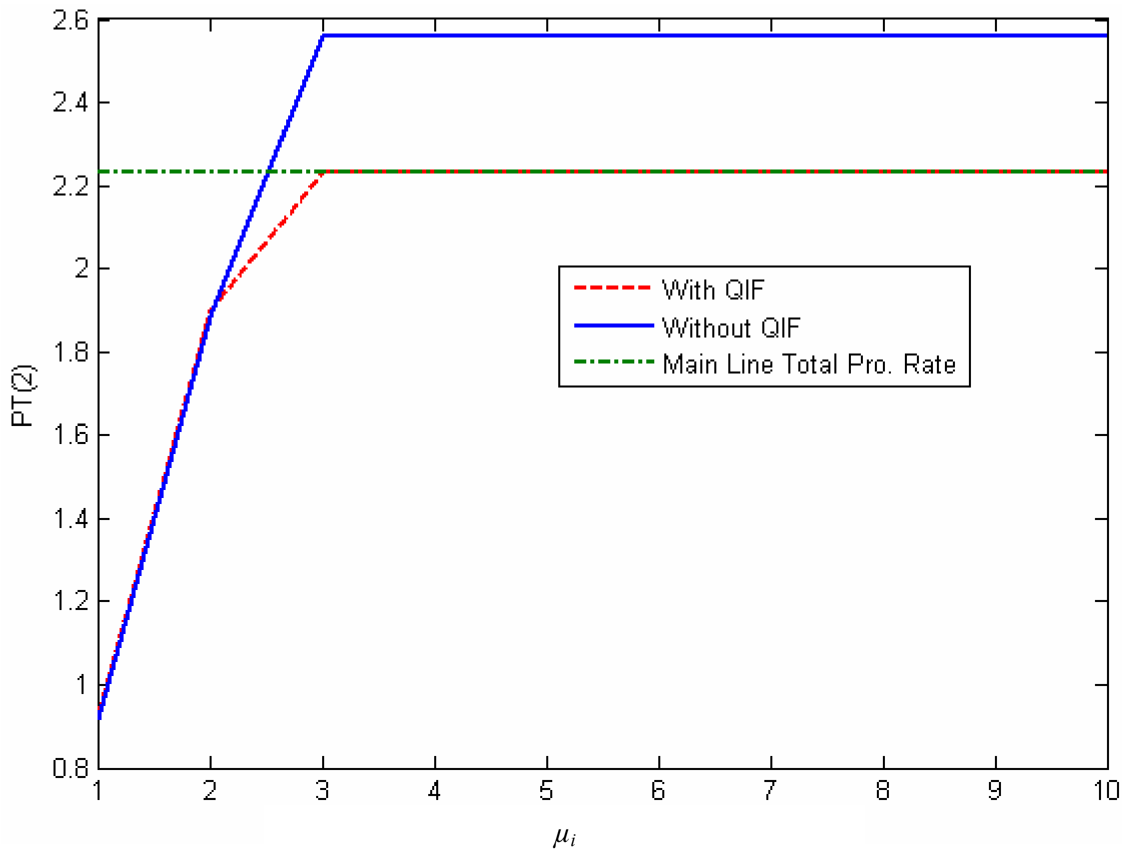

Figure 5.4. Comparing *PT(2)* of the rework systems to total throughput of the serial transfer line with QIF in the case of low yield

When we add a rework line into a serial transfer line, the total throughput rate of the production system decreases in all cases. However, this decrease in the throughput rate represents the defective parts that will be scrapped later, so this reduction of total throughput rate is not a harmful throughput effect. Therefore, it is convenient to compare the throughput rate of rework loop systems with the **effective** throughput rate of the main line instead of **total** throughput rate. It is observed from the graphs that effect of rework line differs with respect to both existence of QIF in the rework loop system and the yield of the main line. For instance, in Figure 5.1, it can be seen that the effective throughput rate of the main line is higher than the throughput rate of the rework loop system without QIF. So, when the yield of a production system is high, removing QIF from the main line and adding a rework line instead of QIF is not beneficial in terms of throughput rates of non-defective parts.

In the first case, there is no change in the throughput rate while service rates of the rework line's machines are increasing. This is because the main line produces good parts frequently in the first case, and accordingly the rework rate is small. In the second case, the throughput rate is increasing until the service rates are equal to 3 since the main line produces bad parts frequently and the rework line is the bottleneck of the system when the service rates of its machines are below three parts per time unit. Note that the production rates of the serial Line 2 and Line 4 are lower in the case with QIF rather than the case of without QIF. It is due to the fact that, if there is QIF in the rework system, less amount of defective parts will sent to Line 4 and this leads a decrease in the Line 2. However, the throughput rate of the rework loop system is higher in the case of with QIF.

5.2. Effect of the Quality Information Feedback on Rework Loop Production Systems

In this section, we present a set of numerical experiments that provide insight for the behavior of QIF in rework loop production with productivity issues.

We decompose the rework production system into four serial lines as in Figure 4.1. We state that the throughput rate of the rework production system is equal to the total (not effective) throughput rate of decomposed serial Line 1. When we analyze the serial Line 1, we take into account the other machines' blocking and starvation effects, and we embed these effects into the merge machine's parameters. In other words, we change the repair and failure rates of merge machine by using the formulas $r^{new}{}_j = r_j s_{j2}\left(1-b_j\right)$ and $p^{new}{}_j = p_j + r_j.\left(1-s_{j2}\left(1-b_j\right)\right)$. We assume that merge machine always gives the priority to the rework line when it takes a part from its upstream buffers. Therefore, when there is QIF in the rework production system, less parts are transferred to the rework line and there is more starvation in B_{MT+MR} . Consequently, s_{j2} increases, and this leads to an increase in the isolated efficiency of merge machine in decomposed Line 1. On the other hand, having QIF means having more inspections than otherwise, and therefore other machines in Line 1 tend to stop more often. As a result, the isolated efficiencies of other machines in decomposed Line 1 decrease.

To sum up, when we integrate QIF policy into rework loop system and stop the machines in Line 1 while they are producing bad parts, the isolated production rate of the merge machine increases but the isolated efficiencies of other machines in Line 1 decrease. Consequently we can not decide whether QIF increases the production rate of Line 1 or not without any computation. For that reason, we conduct experiments by changing one parameter while others are held constant and observe their effects on the difference between the production rate of the rework loop system with QIF and without QIF in order to find out the cases where QIF is beneficial for the rework loop system.

5.2.1. Cases Where QIF Increases the Throughput Rate of the Rework Loop System

In this section, we analyze the cases where QIF increases the production rate of the rework loop system. Here, we perform 10 experiments for each case and increase only one parameter such as service rate, buffer capacity, etc. while the other parameters are fixed. In other words, a sensitivity study is done with respect to buffer capacities, the machines' service rates, operational failure rates, quality failure rates and mean times to detect bad parts. For all cases, we investigate the improvement of the production rate when we use QIF in the rework loop system. The improvement is calculated according to (5.1).

$$Improvement\% = 100 \times \frac{PT(1)_{with\ QIF} - PT(1)_{without\ QIF}}{PT(1)_{without\ QIF}} \tag{5.1}$$

In all experiments, there are 8 machines in the main line and 3 machines in the rework line. The fourth machine is the merge machine and the seventh machine is the split machine for all experiments.

<u>5.2.1.1 Changes in quality failure rates in the main line.</u> In this section we observe the effect of quality failure rates on the throughput rate improvement when QIF is integrated into rework loop system. We consider three cases and carry out 10 experiments for each case. In the first case, the detecting quality failure rates, viz. h $(h = f - p)$, of split machine, merge machine, machines in serial Lines 1 and Line 2 are high in the experiments, i.e. $h_i = 0.44,\ \ i = 1,...,7$. In the second case, the detecting quality failure rates of split machine and merge machine are high but detecting quality failure rates of machines

in serial Line 1 and Line2 are low, i.e. $h_i = 0.44 \;\; for \;\; i = 4,7$ and $h_k = 0.005 \;\; for \;\; k = 1,2,3,5,6$. In the final case, the detecting quality failure rates of split machine and merge machine are low but detecting quality failure rates of machines in serial Line 1 and Line2 are high, i.e. $h_i = 0.005 \;\; for \;\; i = 4,7$ and $h_k = 0.44 \;\; for \;\; k = 1,2,3,5,6$. In all experiments, we alter the quality failure rates of the machines in the main line, i.e. g_j, $j = 1,2,3,5,6$, from 0.001 to 0.5 simultaneously for the three cases while other parameters are held constant.

- ***<u>Case 1:</u>*** The parameters used in the experiments are in Table 5.3.

Table 5.3. The machine parameters used in the first case of section 5.2.1.1

	Machine	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	3.000	0.900	0.450	0.010		55
	2	3.000	0.900	0.450	0.010		55
	3	3.000	0.900	0.450	0.010		55
	4	**3.000**	**0.900**	**0.450**	**0.010**		**55**
	5	3.000	0.900	0.450	0.010		55
	6	3.000	0.900	0.450	0.010		55
	7	**3.000**	**0.900**	**0.450**	**0.010**		**55**
	8	3.000	0.900		0.010		55
REWORK LINE	*9*	3.000	0.950		0.010		55
	10	3.000	0.950		0.010		55
	11	3.000	0.950		0.010		55

Figures 5.5-5.9 are graphs of the performance of rework loop production system with both QIF and without QIF for $h_i = 0.44, \; i = 1,2,3,4,5,6,7$. Figure 5.5 shows the comparison of throughput rate. Figure 5.6, 5.7 and 5.8 demonstrate the effect of quality failure rates of the machines on throughput rate of serial Line 2, serial Line 4 and the rework rate. Figure 5.9 shows the effect of quality failure rates of on the improvement of the production rate of the system when we use QIF.

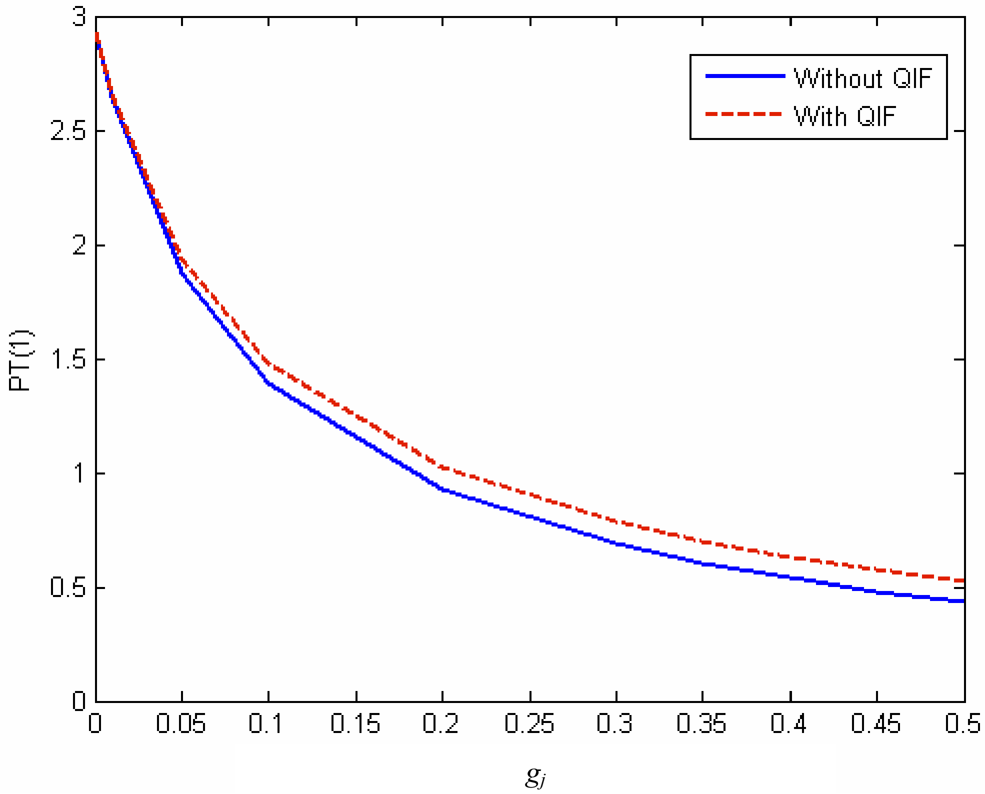

Figure 5.5 Effect of g_j on throughput rate when $h_i = 0.44$ – With and without QIF

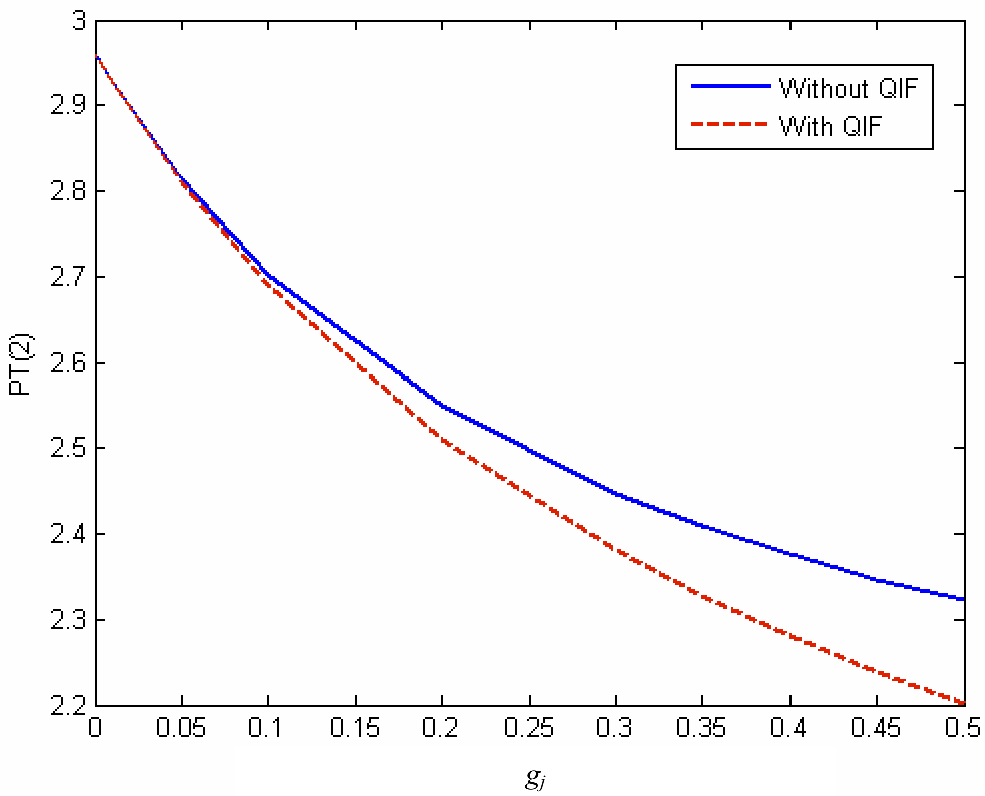

Figure 5.6 Effect of g_j on *PT(2)* when $h_i = 0.44$ – With and without QIF

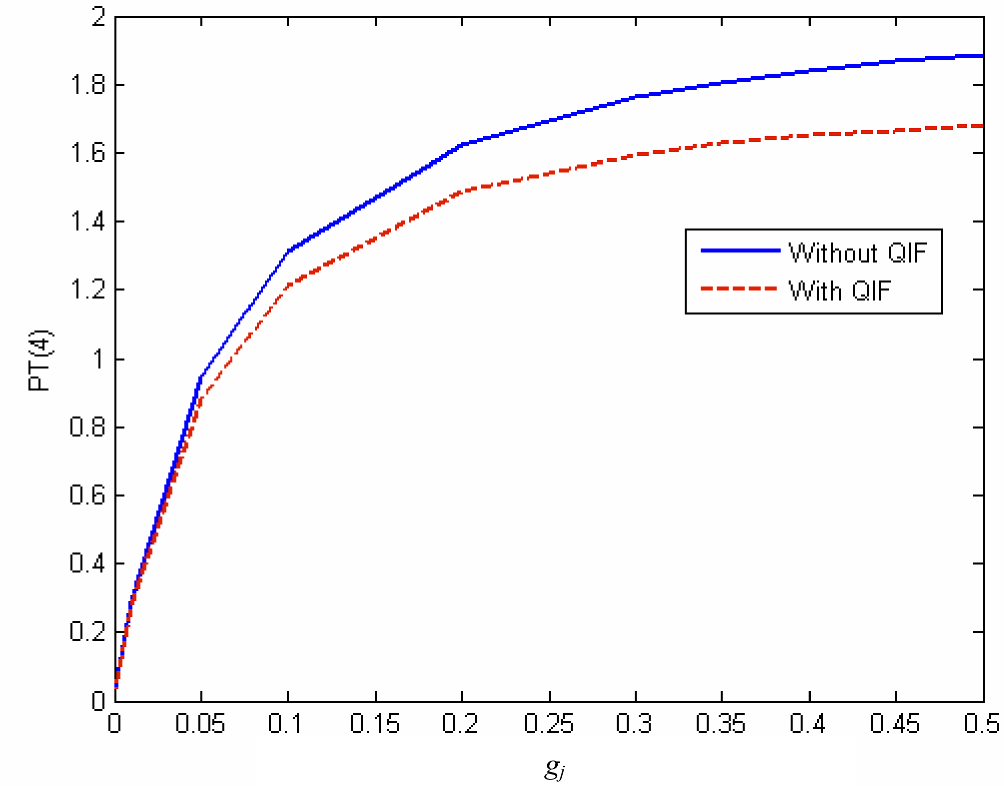

Figure 5.7 Effect of g_j on PT(4) when $h_i = 0.44$ – With and without QIF

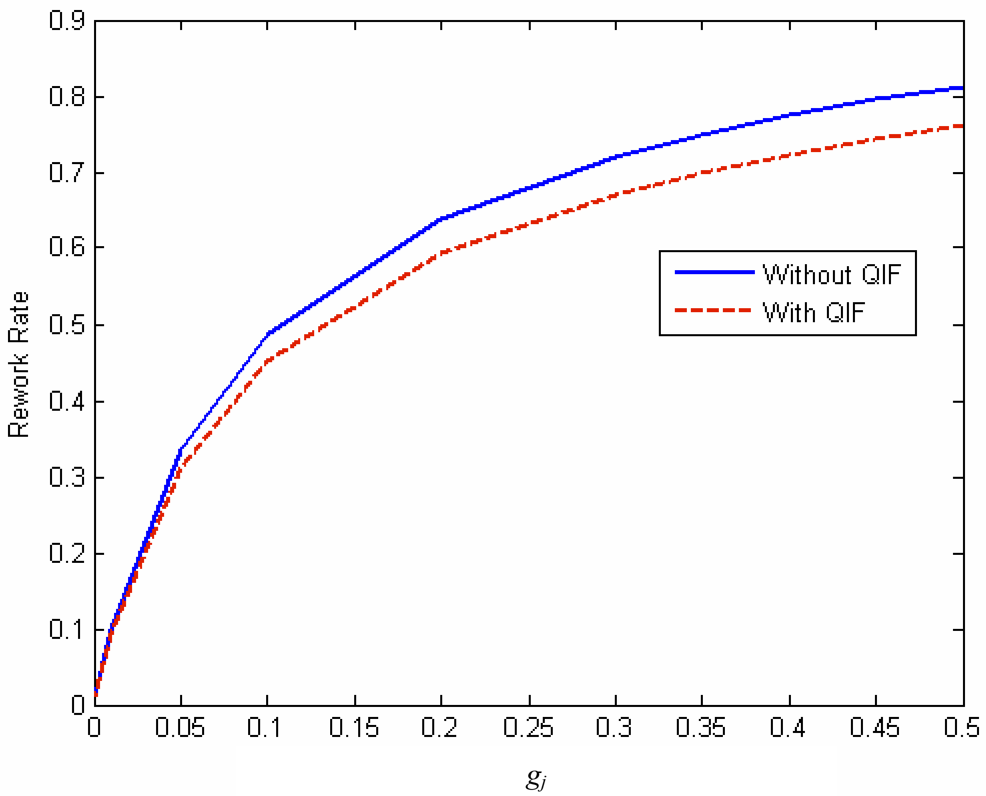

Figure 5.8 Effect of g_j on α when $h_i = 0.44$ - With and without QIF

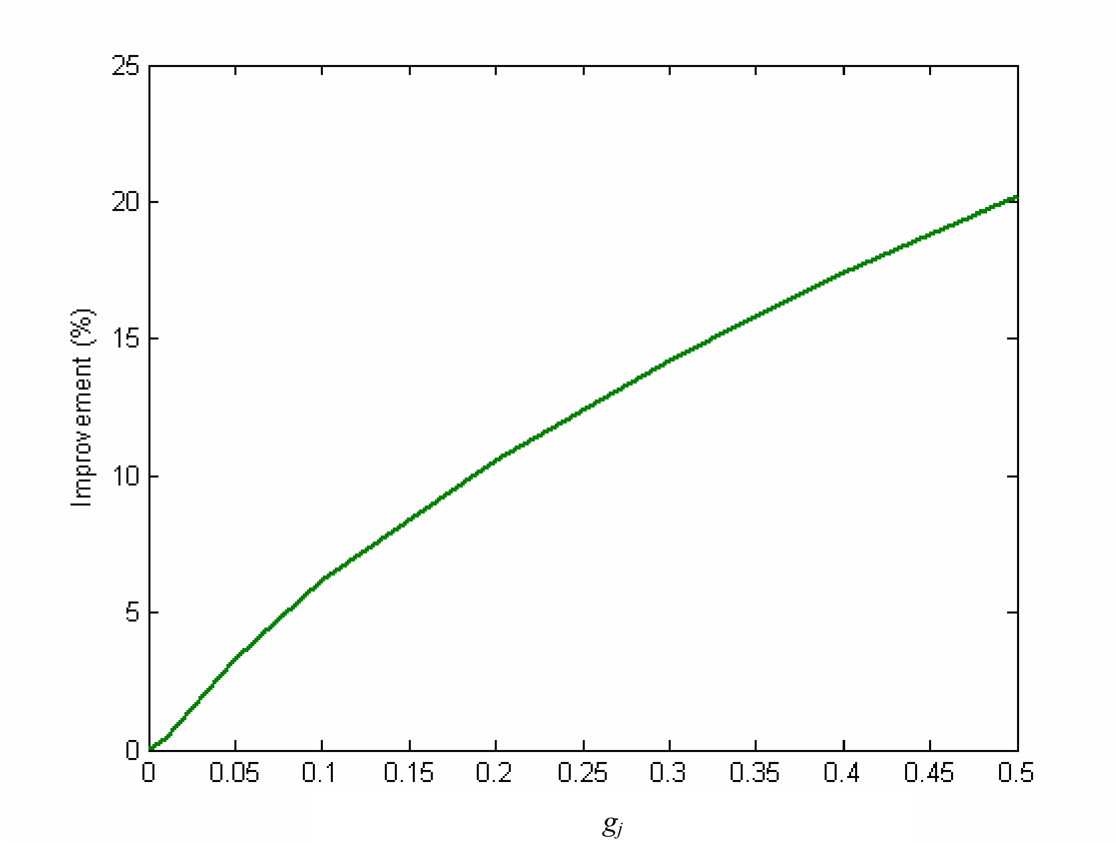

Figure 5.9 Improvement of production rate with QIF when $h_i = 0.44$

- ***Case* 2:** The parameters used in the experiments are in Table 5.4.

Table 5.4. The machine parameters used in the second case of section 5.2.1.1

	Machine	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	3.000	0.900	0.015	0.010		55
	2	3.000	0.900	0.015	0.010		55
	3	3.000	0.900	0.015	0.010		55
	4	**3.000**	**0.900**	**0.45**	**0.010**		**55**
	5	3.000	0.900	0.015	0.010		55
	6	3.000	0.900	0.015	0.010		55
	7	**3.000**	**0.900**	**0.45**	**0.010**		**55**
	8	3.000	0.900		0.010		55
REWORK LINE	*9*	3.000	0.950		0.010		55
	10	3.000	0.950		0.010		55
	11	3.000	0.950		0.010		55

Figures 5.10-5.14 are graphs of the performance of rework loop production system with both QIF and without QIF for $h_i = 0.44$, $i = 4,7$ and $h_k = 0.005$, $k = 1,2,3,5,6$. The throughput rate of the rework loop system both with QIF and without QIF is depicted in Figure 5.10.

Figure 5.11, 5.12 and 5.13 demonstrate the effect of quality failure rates of the machines on throughput rate of serial Line 2, serial Line 4 and the rework rate. Figure 5.14 shows the effect of quality failure rates of on the improvement of the production rate of the system when we use QIF.

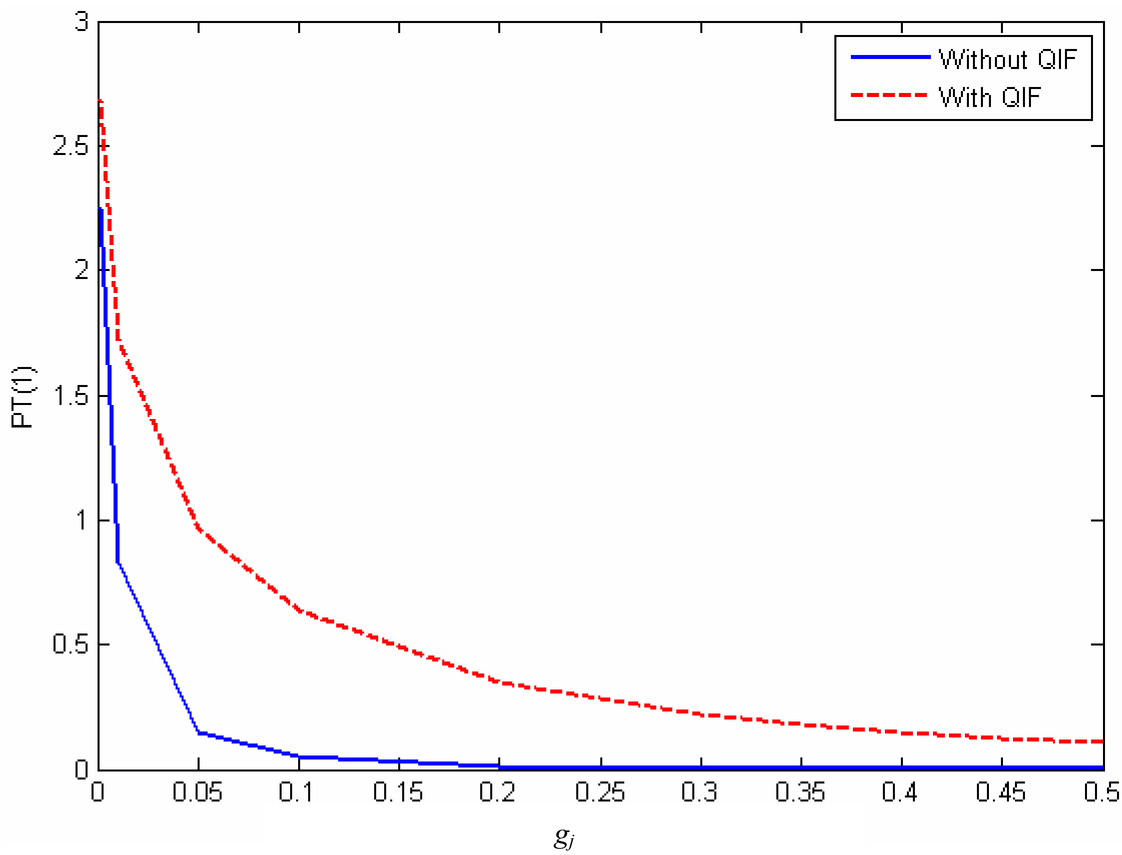

Figure 5.10. Effect of g_j on throughput rate when $h_i = 0.44$ and $h_k = 0.005$ – With and without QIF

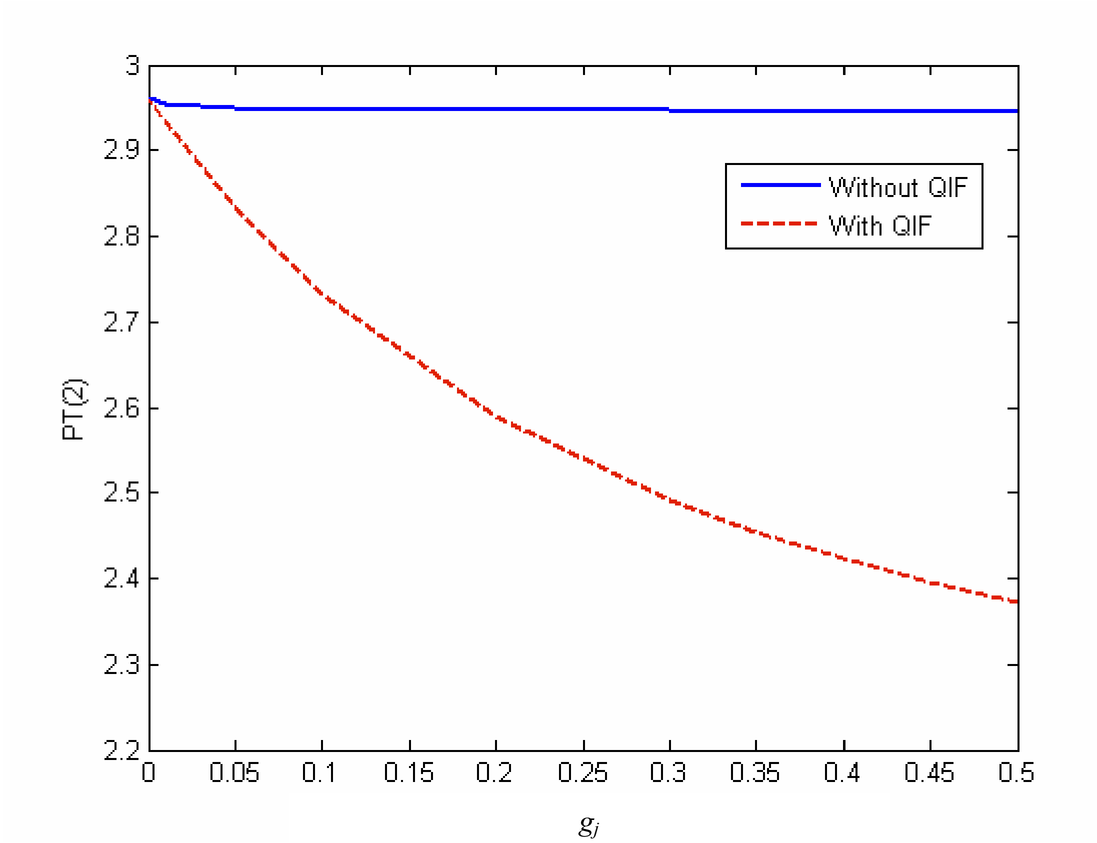

Figure 5.11 Effect of g_j on *PT(2)* when $h_i = 0.44$ and $h_k = 0.005$ – With and without QIF

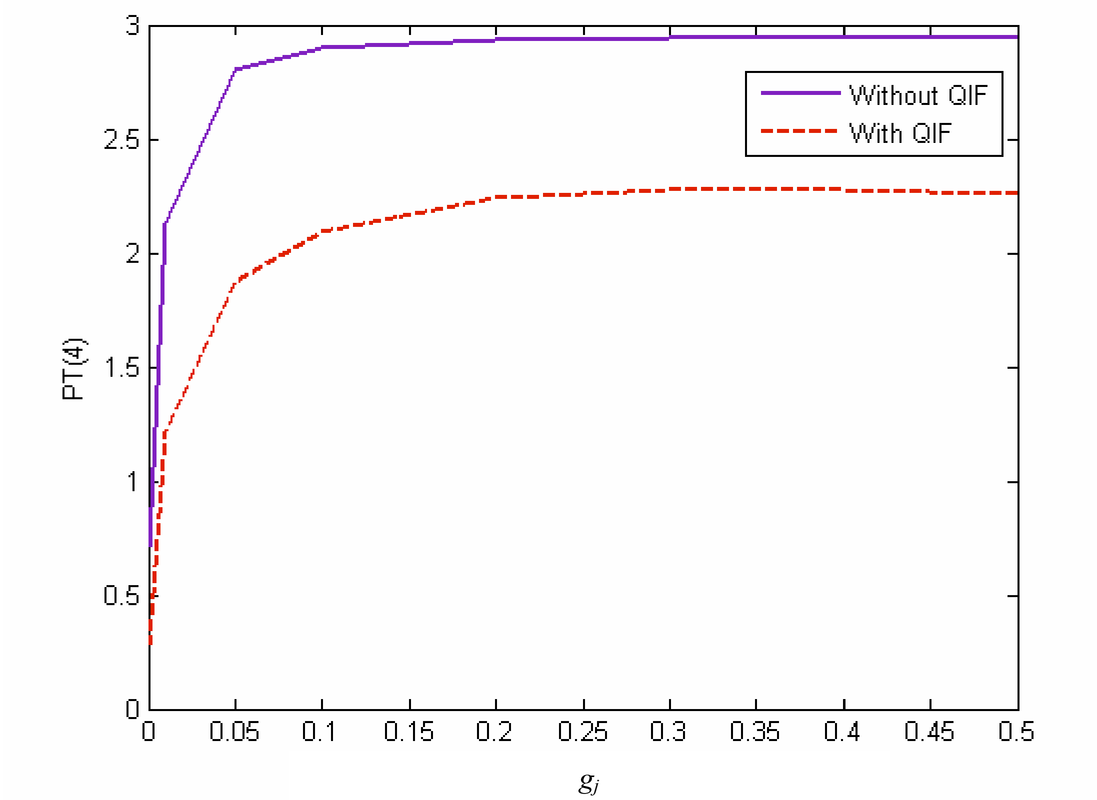

Figure 5.12 Effect of g_j on *PT(4)* when $h_i = 0.44$ and $h_k = 0.005$ – With and without QIF

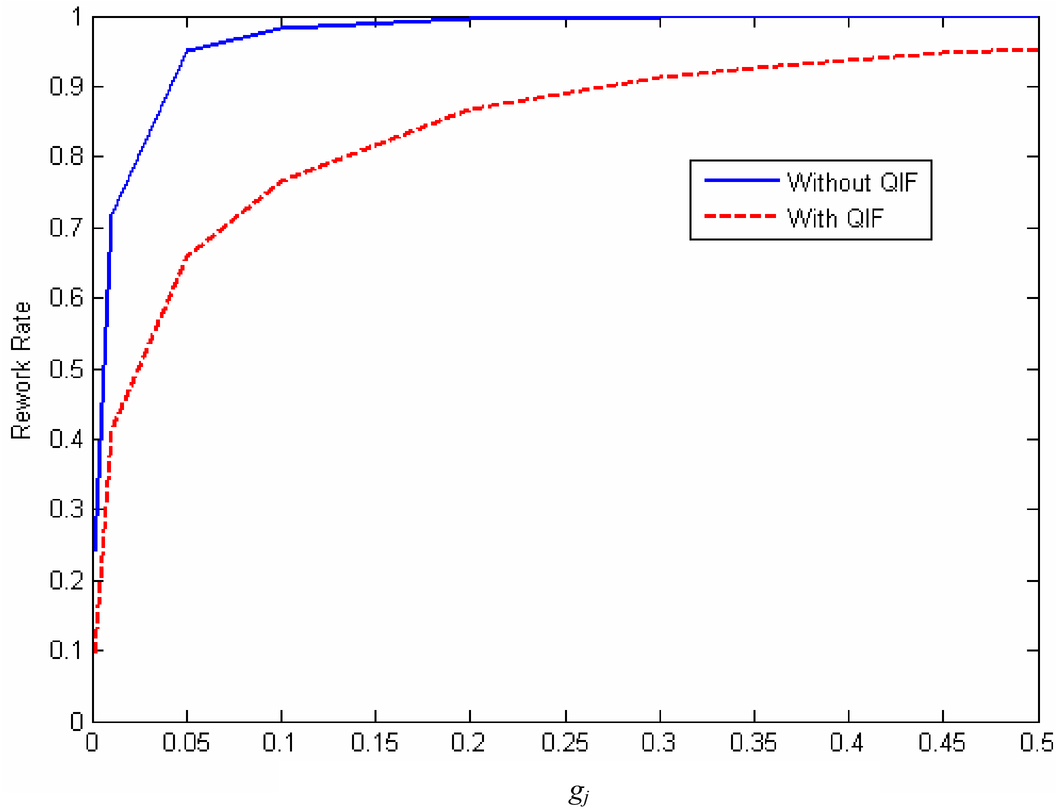

Figure 5.13 Effect of g_j on α when $h_i = 0.44$ and $h_k = 0.005$ – With and without QIF

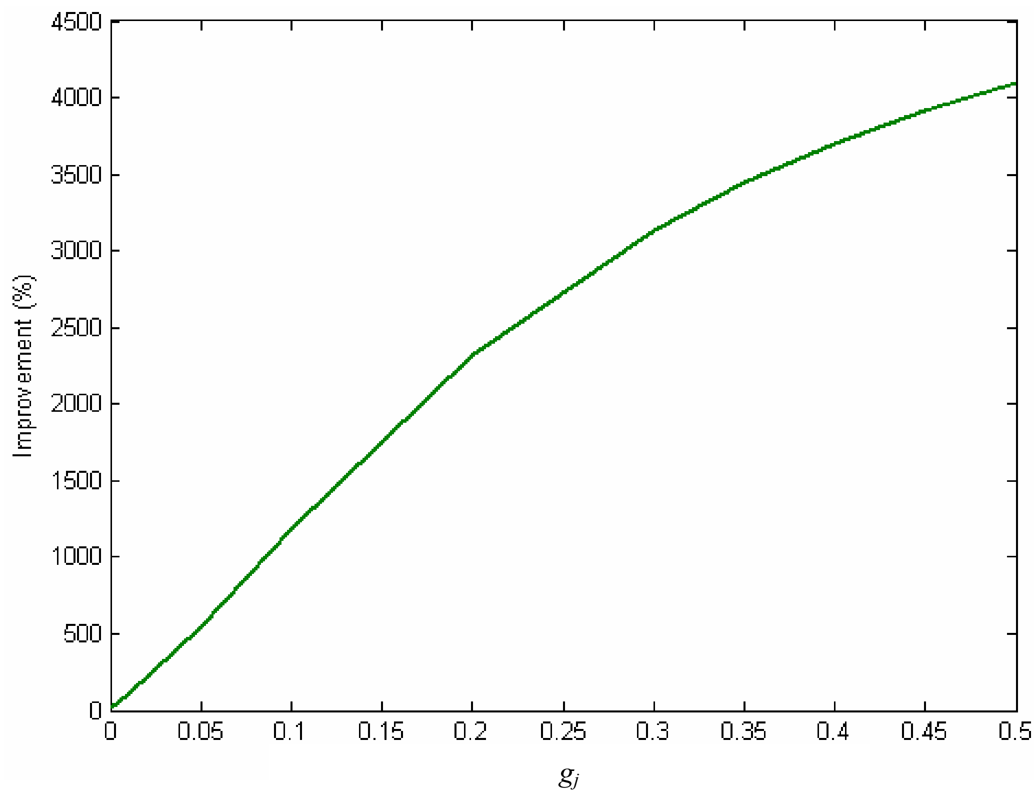

Figure 5.14 Improvement of production rate with QIF when $h_i = 0.44$ and $h_k = 0.005$

Case ***3:*** The parameters used in the experiments are in Table 5.5.

Table 5.5. The machine parameters used in the third case of section 5.2.1.1

	Machine	μ	r	f	p	g	N
MAIN LINE	*1*	3.000	0.900	0.450	0.010		55
	2	3.000	0.900	0.450	0.010		55
	3	3.000	0.900	0.450	0.010		55
	4	**3.000**	**0.900**	**0.015**	**0.010**		**55**
	5	3.000	0.900	0.450	0.010		55
	6	3.000	0.900	0.450	0.010		55
	7	**3.000**	**0.900**	**0.015**	**0.010**		**55**
	8	3.000	0.900		0.010		55
REWORK LINE	*9*	3.000	0.950		0.010		55
	10	3.000	0.950		0.010		55
	11	3.000	0.950		0.010		55

The throughput rate of the rework loop system both with QIF and without QIF is depicted in Figure 5.15 for $h_i = 0.005$, $i = 4,7$ and $h_k = 0.44$, $k = 1,2,3,5,6$. Figure 5.16 shows the effect of quality failure rates of on the improvement of the throughput rate of the system when we use QIF for the third case.

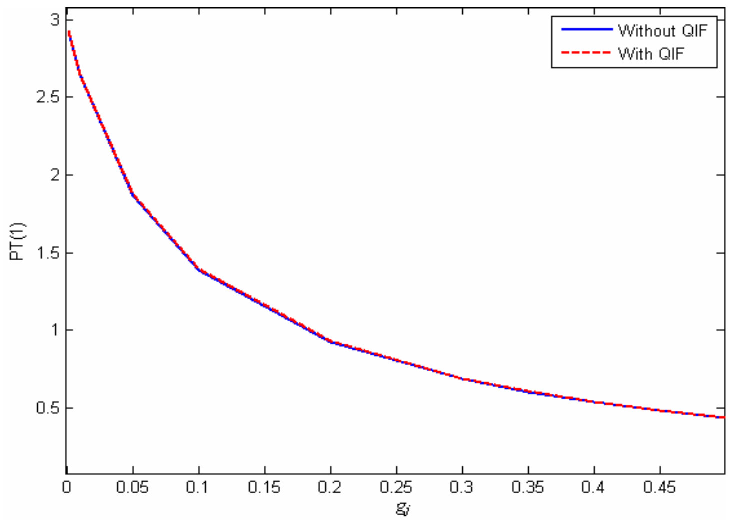

Figure 5.15 Effect of g_j on *PT(1)* when $h_i = 0.005$ and $h_k = 0.44$ – With and without QIF

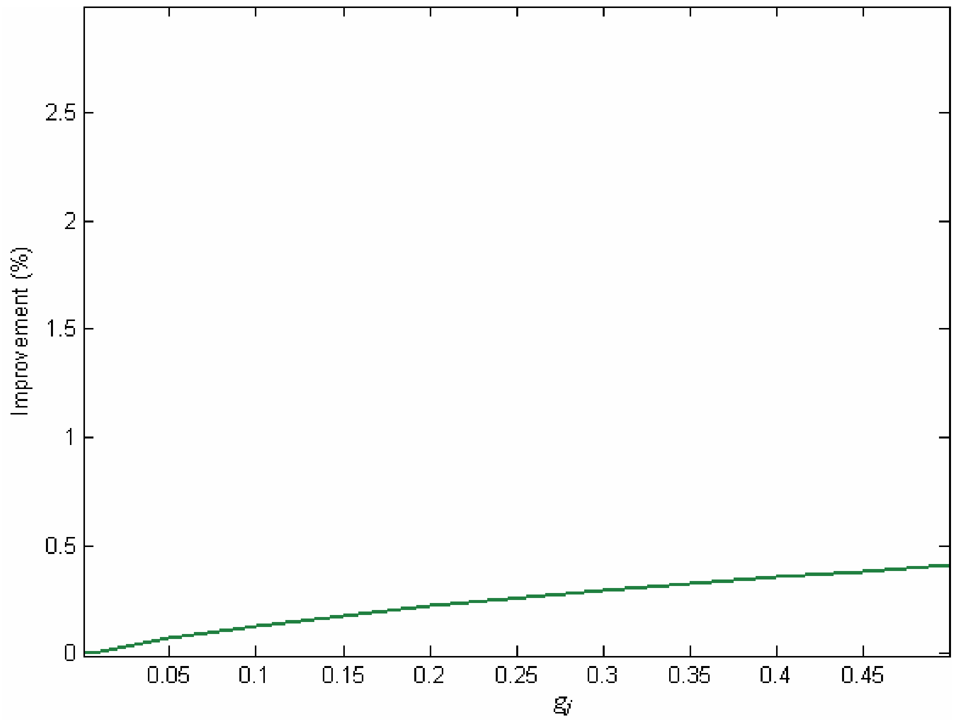

Figure 5.16 Improvement of production rate with QIF when $h_i = 0.005$ and $h_k = 0.44$

It can be observed from the three cases that QIF significantly increases the production rate of the system while quality failure rates of the main line is increasing. In Figure 5.15, the throughput rate of the rework system without QIF is very close to the throughput rate of the rework system with QIF since the h rate of the split and merge machines are low ($h_i = 0.005, \quad i = 4,7$). However, there is still an improvement even in the third case when there is QIF in the system. Note that, in the second case the throughput rate improvement is significantly higher than the other cases since in the second case, h rates of split and merge machines are high while h rates of other machines are low.

<u>5.2.1.2. Change in buffer size.</u> Next a sensitivity study with respect to the buffer capacities is carried out. As stated earlier, the mean time to detect a bad part is a function of the buffer size when there is QIF in the system. For the serial transfer lines with QIF, if the size of the buffers is large, then more bad parts will accumulate in the buffer, and therefore mean time to detect a bad part will increase and this leads to a decrease in the effective throughput rate. In this section we observe the behavior of the throughput rate of the rework loop system when we increase the buffers capacities from 5 to 85. We consider three cases and carry out 10 experiments for each case. In the first case, h rates of split

machine, merge machine, machines in serial Line 1 and Line 2 are high in the experiments, viz. $h_i = 0.44 \quad for \quad i = 4,7$ and $h_k = 0.35 \quad for \quad k = 1,2,3,5,6$. In the second case, *h* rates of split and merge machines are high but *h* rates of machines in Line 1 and Line 2 are low, viz. $h_i = 0.44 \quad for \quad i = 4,7$ and $h_k = 0.02 \quad for \quad k = 1,2,3,5,6$. In the last case, *h* rates of split and merge machines are low but *h* rates of machines in Line 1 and Line 2 are high, viz. $h_i = 0.04 \quad for \quad i = 4,7$ and $h_k = 0.35 \quad for \quad k = 1,2,3,5,6$. In all experiments, we alter the buffer capacities of all machines in the system, i.e. $N_j, \ j = 1,...,11,$ from 5 to 80 simultaneously while other parameters are held constant for each case.

- ***Case* 1:** The parameters used in the experiments are in Table 5.6

Table 5.6. The machine parameters used in the first case of section 5.2.1.2

	Machine	**μ**	**r**	**f**	**p**	**g**	**N**
MAIN LINE	*1*	4.000	0.900	0.450	0.100	0.100	
	2	4.000	0.900	0.450	0.100	0.100	
	3	4.000	0.900	0.450	0.100	0.100	
	4	**4.000**	**0.900**	**0.45**	**0.010**	**0.000**	
	5	4.000	0.900	0.450	0.100	0.100	
	6	4.000	0.900	0.450	0.100	0.100	
	7	**4.000**	**0.900**	**0.45**	**0.010**	**0.000**	
	8	4.000	0.900		0.100		
REWORK LINE	*9*	4.000	0.950		0.100		
	10	4.000	0.950		0.100		
	11	4.000	0.950		0.100		

Figures 5.17-5.21 are graphs of the performance of rework loop production system with both QIF and without QIF for $h_i = 0.44 \ for \ i = 4,7$ and $h_k = 0.35 \quad for \quad k = 1,2,3,5,6$. The thorughput rate of the rework loop system, viz *PT(1)* both with QIF and without QIF is depicted in Figure 5.17.

Figure 5.18, 5.19 and 5.20 demonstrate the effect of buffer sizes on throughput rate of serial Line 2, viz. *PT(2)*, serial Line 4, viz. *PT(4)*, and the rework rate,viz. α. Figure

5.21 shows the effect of buffer capacities on the throughput rate improvement when there is QIF in the system.

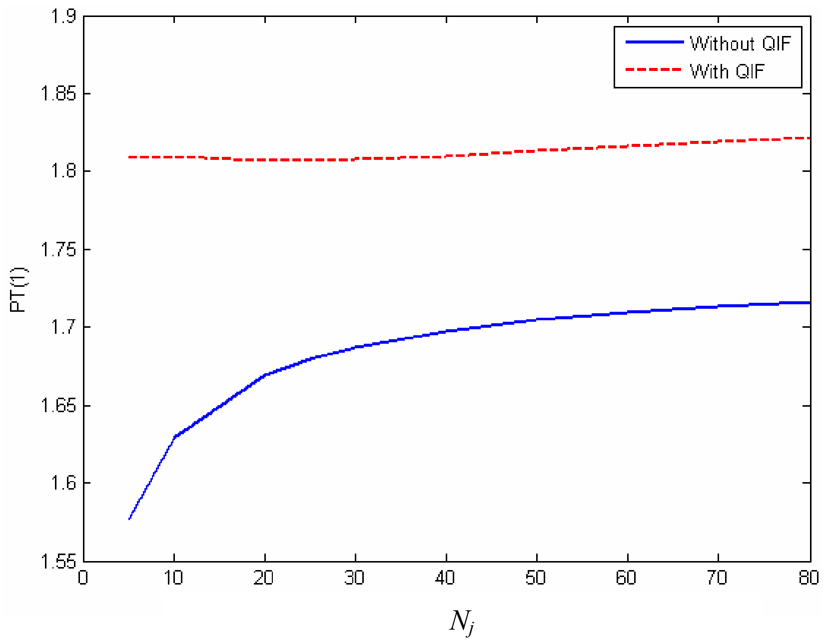

Figure 5.17 Effect of N_j on *PT(1)* when $h_i = 0.44$ and $h_k = 0.35$ – With and without QIF

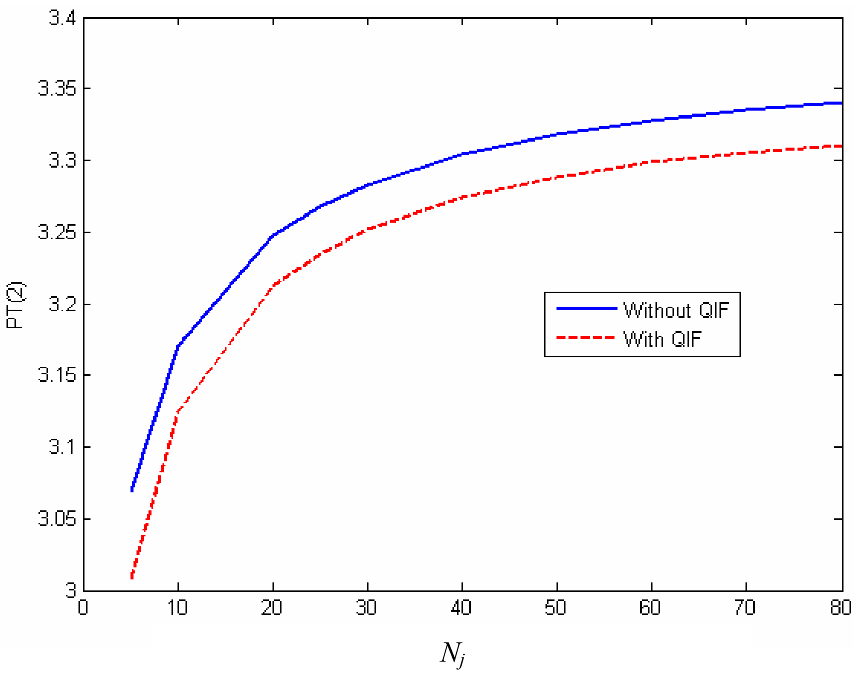

Figure 5.18 Effect of N_j on *PT(2)* when $h_i = 0.44$ and $h_k = 0.35$ – With and without QIF

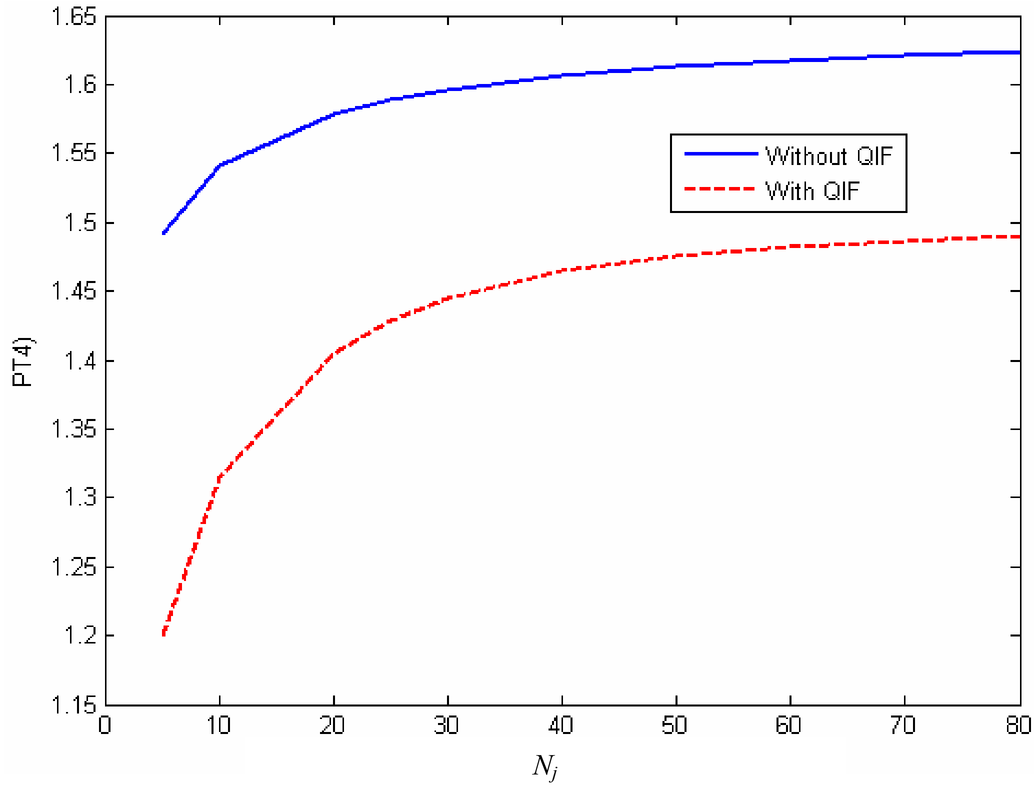

Figure 5.19 Effect of N_j on *PT(4)* when $h_i = 0.44$ and $h_k = 0.35$ – With and without QIF

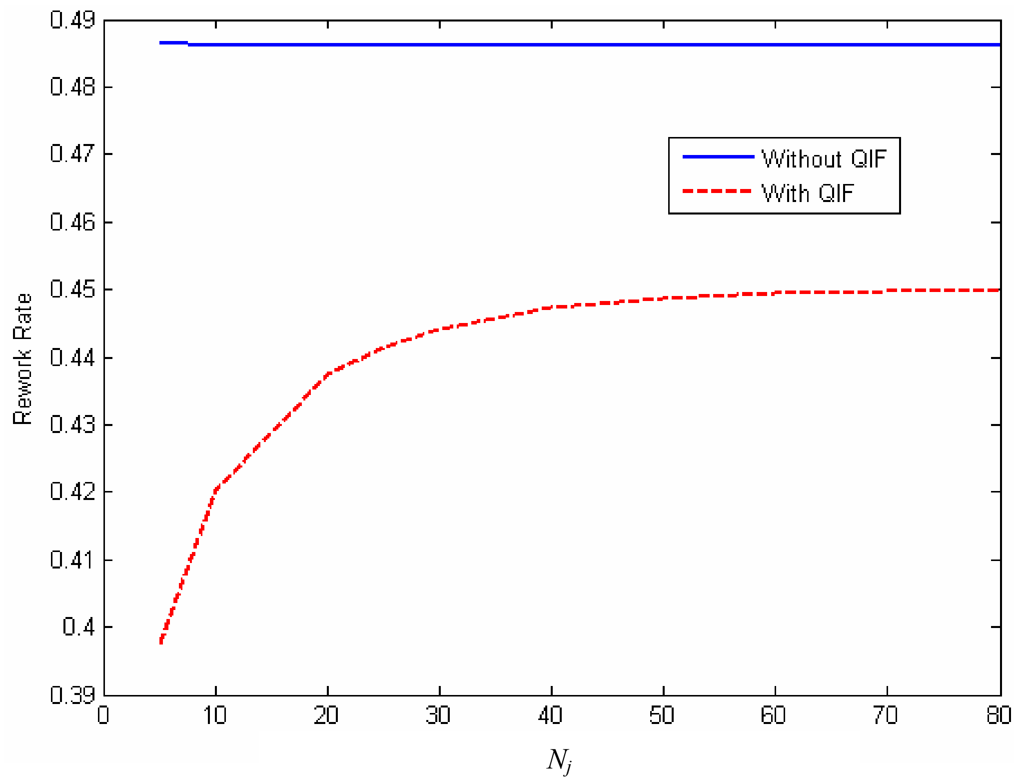

Figure 5.20 Effect of N_j on α when $h_i = 0.44$ and $h_k = 0.35$ – With and without QIF

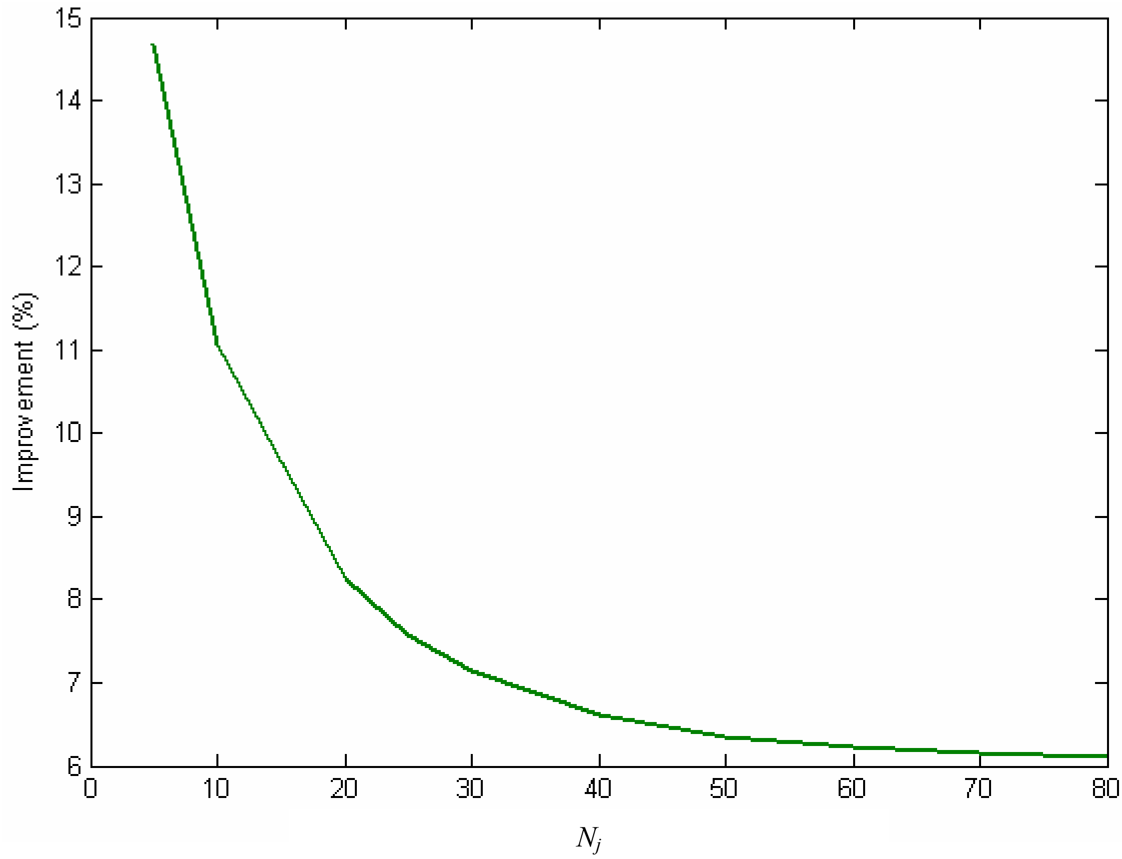

Figure 5.21 Improvement of throughput rate with QIF when $h_i = 0.44$ and $h_k = 0.35$

- ***Case* 2:** The parameters used in the experiments are in Table 5.7

Table 5.7. The machine parameters used in the second case of section 5.2.1.2

	Machine	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	4.000	0.900	0.120	0.100	0.100	
	2	4.000	0.900	0.120	0.100	0.100	
	3	4.000	0.900	0.120	0.100	0.100	
	4	**4.000**	**0.900**	**0.45**	**0.010**	**0.000**	
	5	4.000	0.900	0.120	0.100	0.100	
	6	4.000	0.900	0.120	0.100	0.100	
	7	**4.000**	**0.900**	**0.45**	**0.010**	**0.000**	
	8	4.000	0.900		0.100		
REWORK LINE	*9*	4.000	0.950		0.100		
	10	4.000	0.950		0.100		
	11	4.000	0.950		0.100		

Figures 5.22-5.26 are graphs of the performance of rework loop production system with both QIF and without QIF for $h_i = 0.44 \ \ for \ \ i = 4,7$ and $h_k = 0.02 \ \ for \ \ k = 1,2,3,5,6$. The throughput rate of the rework loop system, viz *PT(1)* both with QIF and without QIF is depicted in Figure 5.22.

Figure 5.23, 5.24 and 5.25 demonstrate the effect of buffer sizes on throughput rate of serial Line 2, viz. *PT(2)*, serial Line 4, viz. *PT(4)*, and the rework rate,viz. α. Figure 5.26 shows the effect of buffer capacities on the throughput rate improvement when there is QIF in the system.

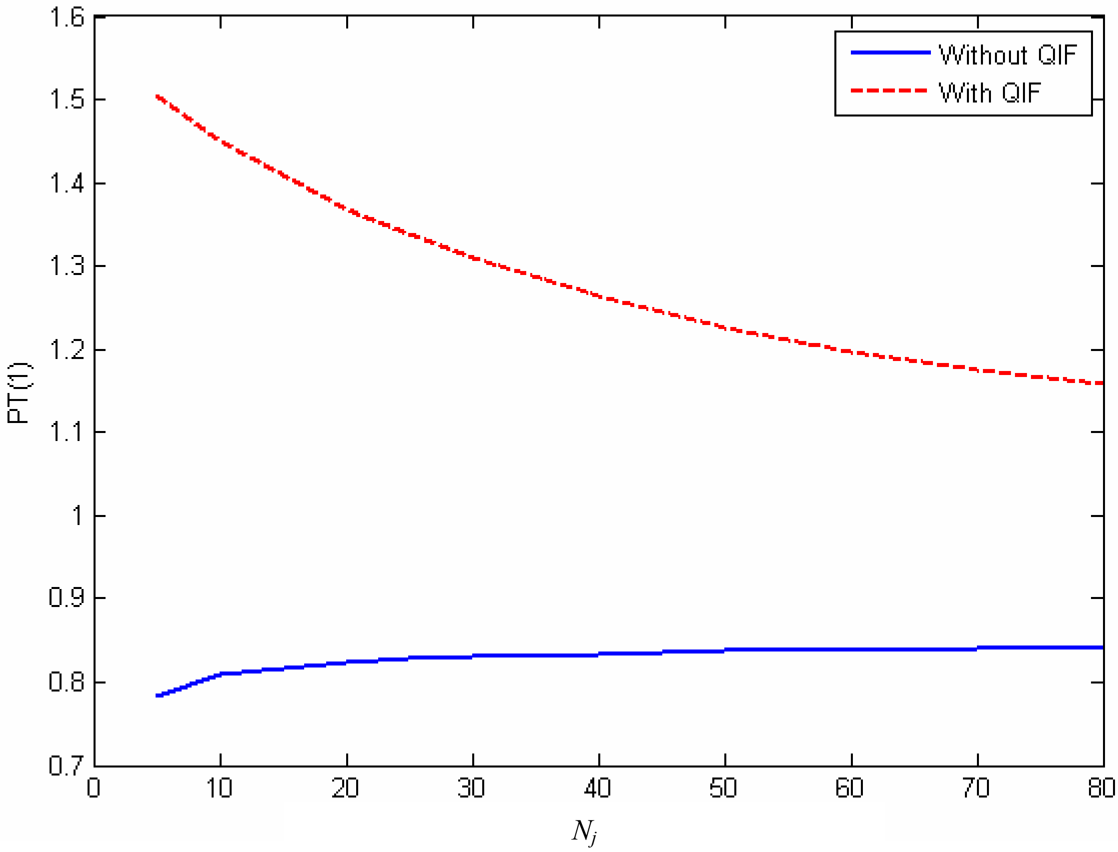

Figure 5.22 Effect of N_j on *PT(1)* when $h_i = 0.44$ and $h_k = 0.02$ – With and without QIF

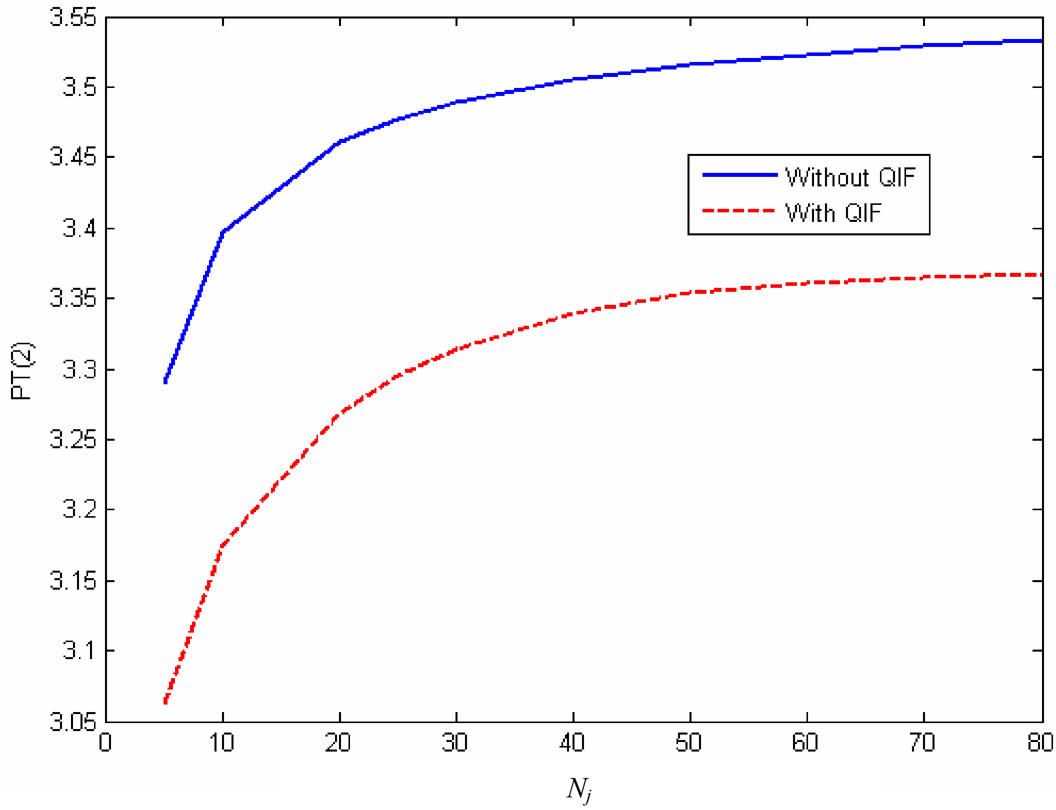

Figure 5.23 Effect of N_j on *PT(2)* when $h_i = 0.44$ and $h_k = 0.02$ – With and without QIF

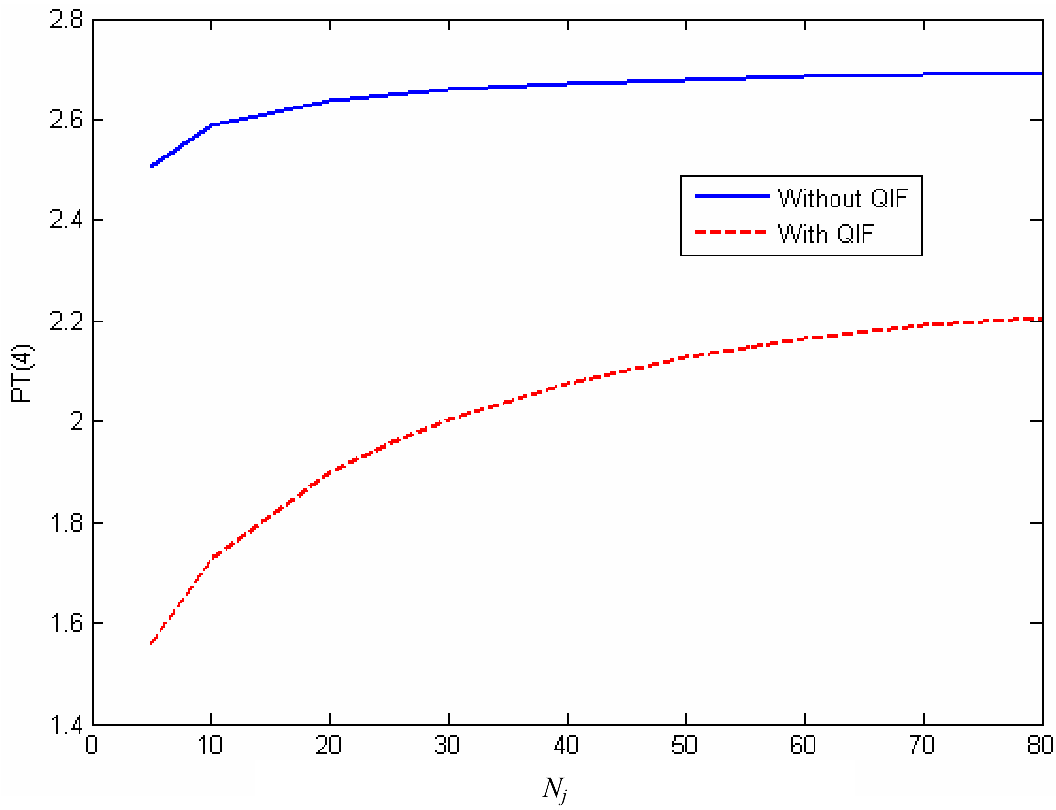

Figure 5.24 Effect of N_j on *PT(4)* when $h_i = 0.44$ and $h_k = 0.02$ – With and without QIF

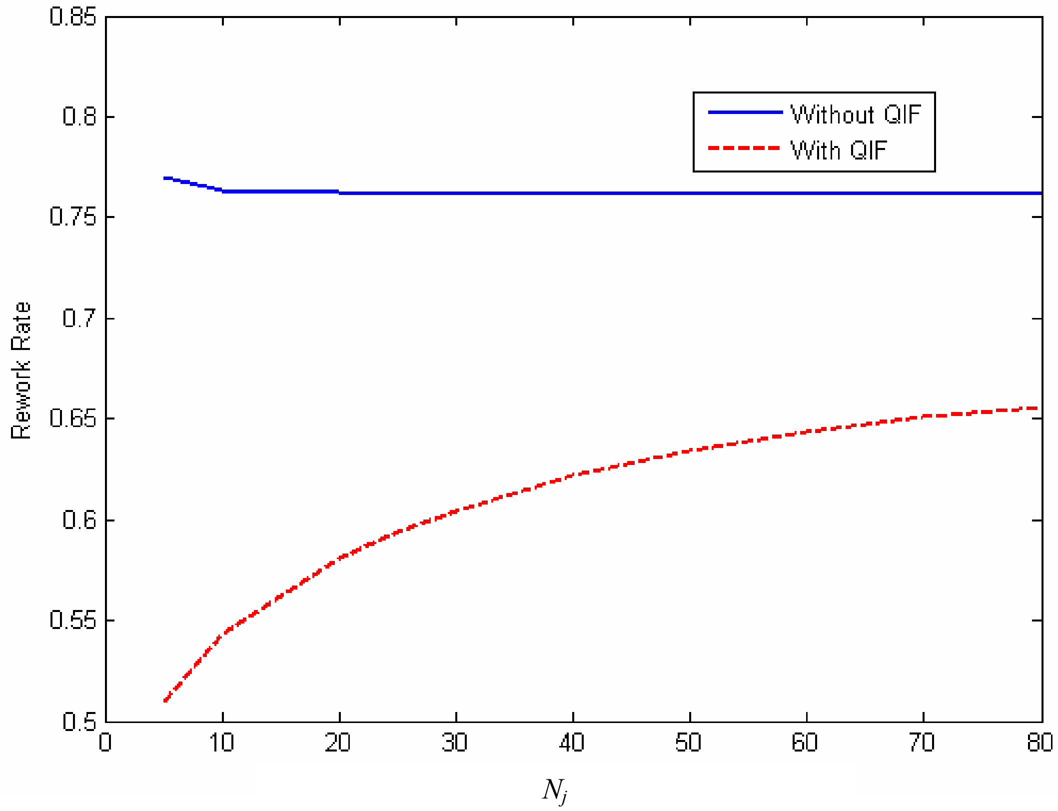

Figure 5.25 Effect of N_j on α when $h_i = 0.44$ and $h_k = 0.02$ – With and without QIF

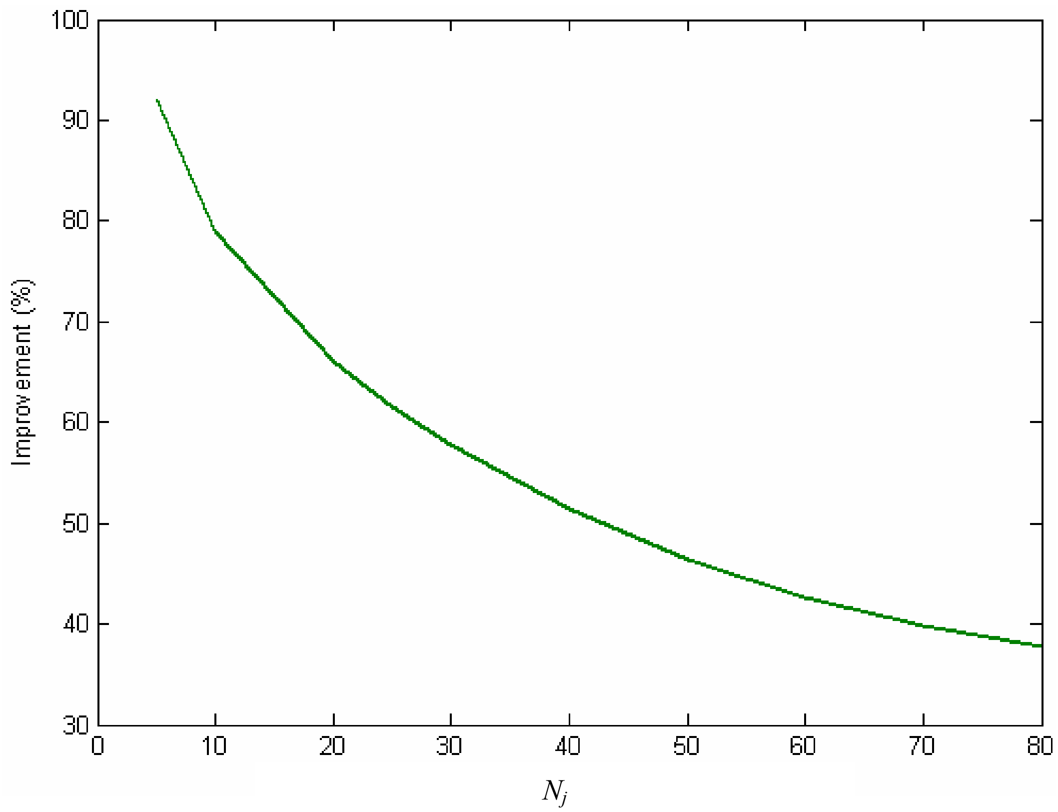

Figure 5.26 Improvement of throughput rate with QIF when $h_i = 0.44$ and $h_k = 0.02$

In the first and second cases, the throughput rate of the rework loop without QIF increases as the buffers sizes increase but the throughput rate of the rework system with QIF decreases as the buffers sizes increase. This is due to the fact that, when the buffers sizes are small, the merge and the split machines detect the bad parts in less time and this causes a significant decrease in the rework rate.

- ***Case* 3:** The parameters used in the experiments are in Table 5.8

Table 5.8. The machine parameters used in the third case of section 5.2.1.2

	Machine	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	4.000	0.900	0.450	0.100	0.100	
	2	4.000	0.900	0.450	0.100	0.100	
	3	4.000	0.900	0.450	0.100	0.100	
	4	**4.000**	**0.900**	**0.05**	**0.010**	**0.000**	
	5	4.000	0.900	0.450	0.100	0.100	
	6	4.000	0.900	0.450	0.100	0.100	
	7	**4.000**	**0.900**	**0.05**	**0.010**	**0.000**	
	8	4.000	0.900		0.100		
REWORK LINE	*9*	4.000	0.950		0.100		
	10	4.000	0.950		0.100		
	11	4.000	0.950		0.100		

Figures 5.27-5.31 are graphs of the performance of rework loop production system with both QIF and without QIF for $h_i = 0.04 \;\; for \;\; i = 4,7$ and $h_k = 0.35 \;\; for \;\; k = 1,2,3,5,6$. The throughput rate of the rework loop system, viz *PT(1)* both with QIF and without QIF is depicted in Figure 5.27.

Figure 5.28, 5.29 and 5.30 demonstrate the effect of buffer sizes on throughput rate of serial Line 2, viz. *PT(2)*, serial Line 4, viz. *PT(4)*, and the rework rate,viz. α . Figure 5.31 shows the effect of buffer capacities on the throughput rate improvement when there is QIF in the system.

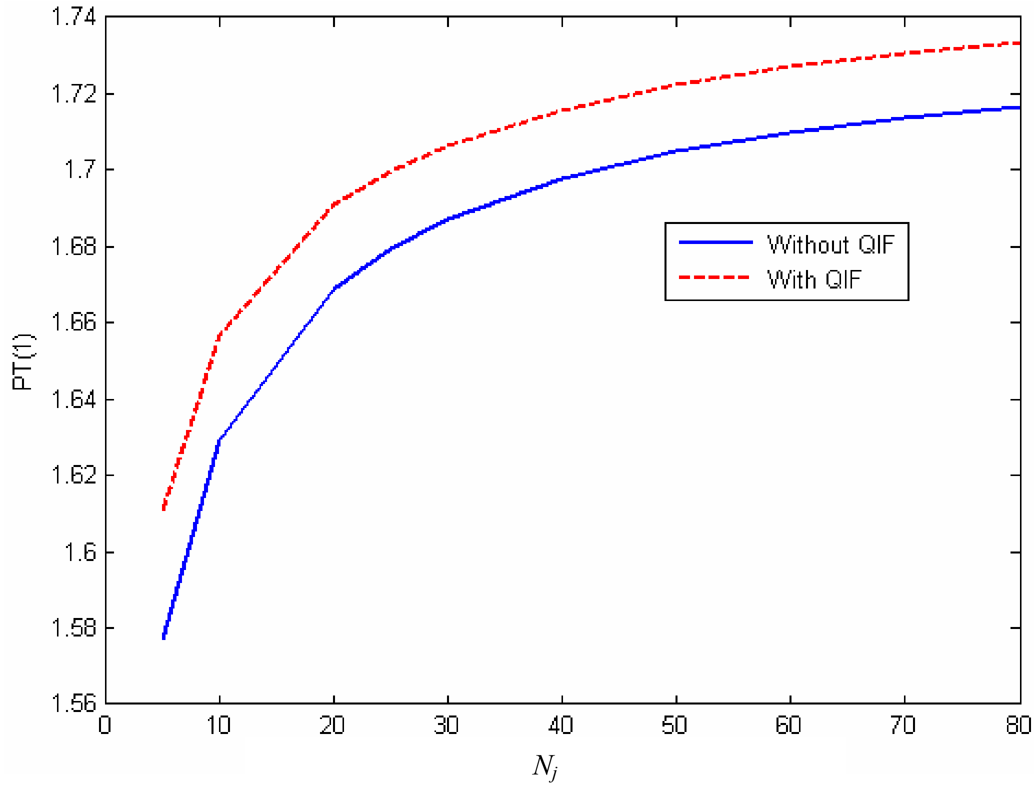

Figure 5.27 Effect of N_j on *PT(1)* when $h_i = 0.04$ and $h_k = 0.35$ – With and without QIF

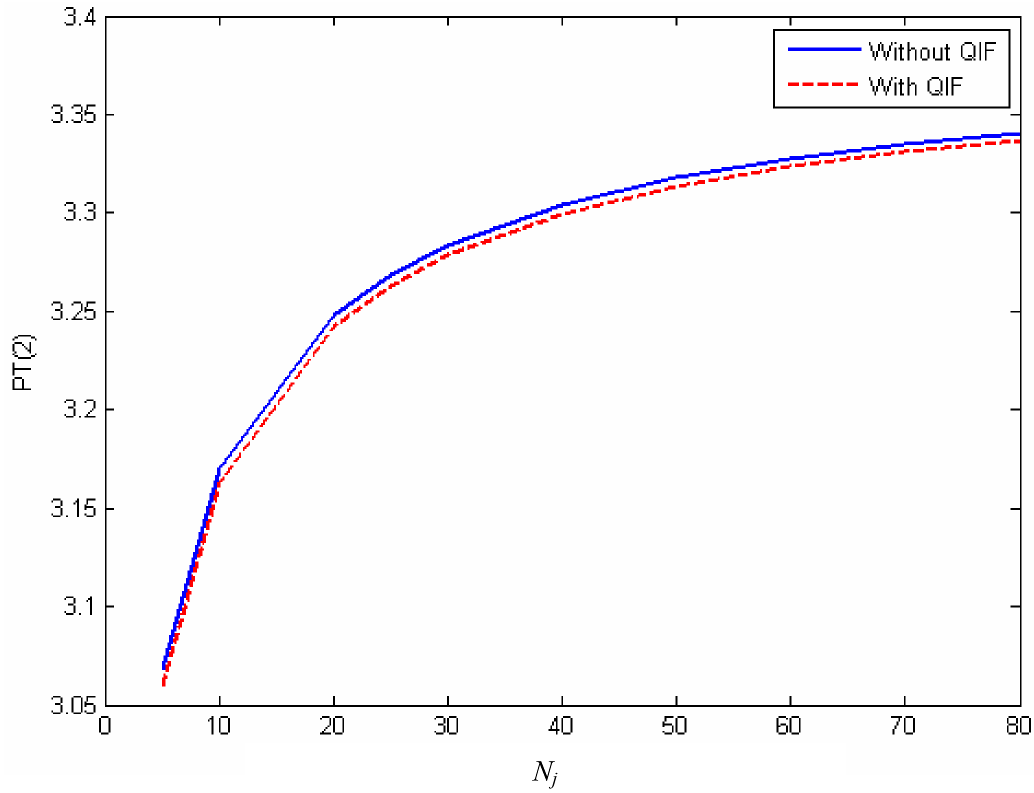

Figure 5.28 Effect of N_j on *PT(2)* when $h_i = 0.04$ and $h_k = 0.35$ – With and without QIF

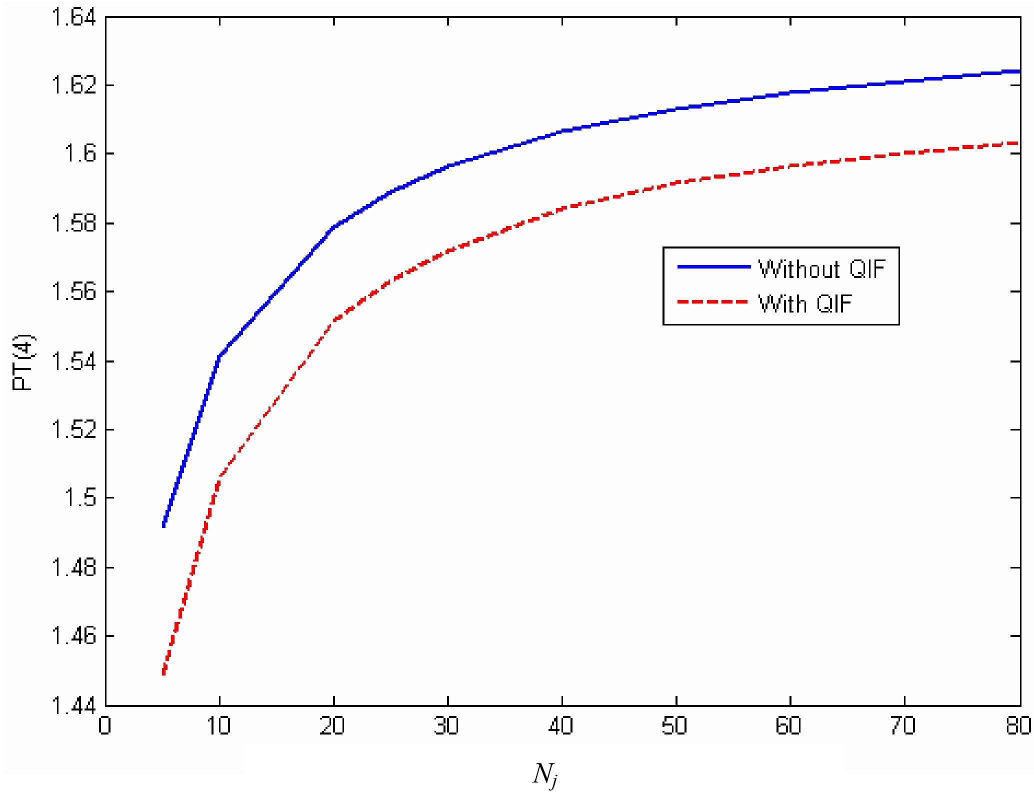

Figure 5.29 Effect of N_j on *PT(4)* when $h_i = 0.04$ and $h_k = 0.35$ – With and without QIF

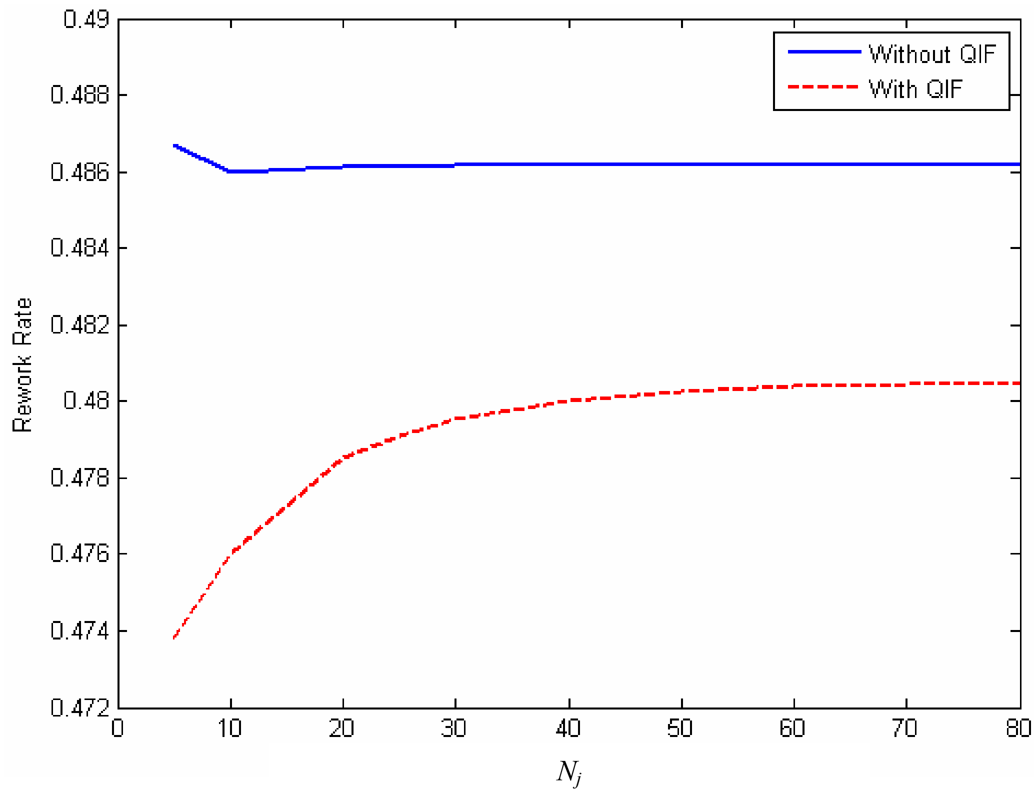

Figure 5.30 Effect of N_j on α when $h_i = 0.04$ and $h_k = 0.35$ – With and without QIF

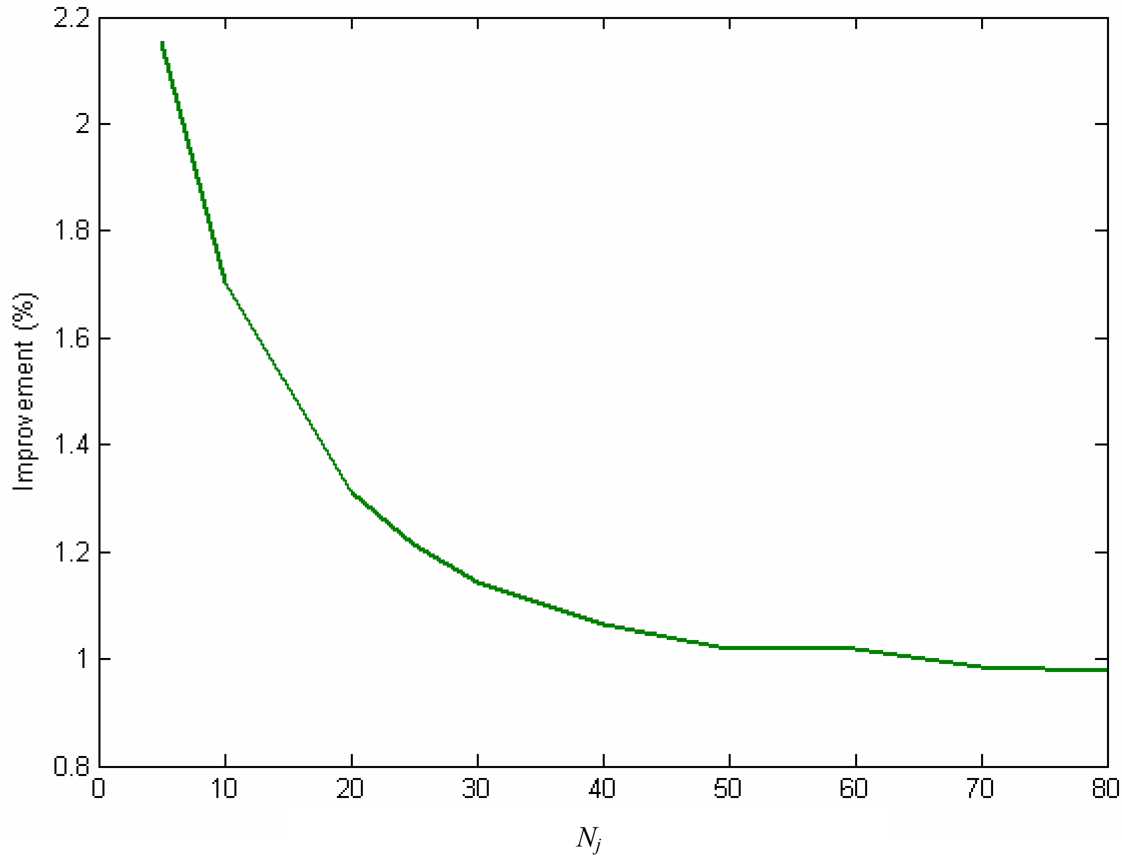

Figure 5.31 Improvement of throughput rate with QIF when $h_i = 0.04$ and $h_k = 0.35$

In the previous cases, the throughput rate of the rework loop without QIF increases as the buffers sizes increase but the throughput rate of the rework system with QIF decreases as the buffers sizes increase. This is due to the fact that, when the buffers sizes are small, the merge and the split machines detect the bad parts in less time and this causes a significant decrease in the rework rate. But when buffers sizes get larger, it takes longer time to detect defective parts for the merge and split machines. Therefore the rework rate increases. So, when there is QIF in the rework system, the percent improvement in throughput rate of the system decreases as the sizes of the buffers are increasing. In the third case, although the throughput rate of the rework system with QIF increases as the buffers sizes get larger, the percent improvement in throughput rate of the system decreases in this case as well. In the third case, the throughput rate of the rework system increases while the buffers capacities are increasing since the *h* rates of the split machine and the merge machine are low. These observations are consistent with Kim, (2005).

5.2.2. Cases where QIF Decreases the Production Rate of the Rework Loop System

In this section, we analyze the cases where QIF decreases the production rate of the rework loop system. Here, we consider two cases and we perform 10 experiments for each case. In the experiments of each case, we increase the service rates of the machine in decomposed serial Line 1, i.e. $\mu_j, \quad j = 1,2,3$, from 1 to 10. In other words, a sensitivity study is done with respect to service rates of the machine in serial Line 1. In the first case, the machines in Line 1 produces bad parts frequently, viz. $g_i = 0.4, \quad for \quad i = 1,2,3$. In the second case, the yield of the serial Line 1 is high, viz. $g_i = 0.01, \quad for \quad i = 1,2,3$. We investigate the improvement of the production rate in each case when there is QIF in the rework loop system. The improvement is calculated using (5.1).

In all experiments, there are 8 machines in the main line and 3 machines in the rework line. Also, the fourth machine is the merge machine and the seventh machine is the split machine for all experiments.

- ***Case 1:*** The parameters used in the experiments are in Table 5.9

Table 5.9. The machine parameters used in the first case of section 5.2.2

	Machine	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*		0.900	0.200	0.100	0.400	55
	2		0.900	0.200	0.100	0.400	55
	3		0.900	0.200	0.100	0.400	55
	4	**9.000**	**0.900**	**0.450**	**0.200**	**0.000**	55
	5	9.000	0.900	0.300	0.100	0.010	55
	6	9.000	0.900	0.300	0.100	0.010	55
	7	**9.000**	**0.900**	**0.350**	**0.200**	**0.000**	55
	8	9.000	0.900		0.010		55
REWORK LINE	*9*	9.000	0.950		0.010		55
	10	9.000	0.950		0.010		55
	11	9.000	0.950		0.010		55

Figures 5.32-5.35 are graphs of the performance of rework loop production system with both QIF and without QIF for $g_i = 0.4, \quad for \quad i = 1,2,3$. The throughput rate of the rework loop system, viz *PT(1)* both with QIF and without QIF is depicted in Figure 5.32.

Figure 5.33 and 5.34 demonstrate the effect of service rates of machines in Line 1 on throughput rate of serial Line 2, viz. *PT(2)* and serial Line 4, viz. *PT(4)*. Figure 5.35 shows the service rates of machines in Line 1 on the throughput rate improvement when there is QIF in the system.

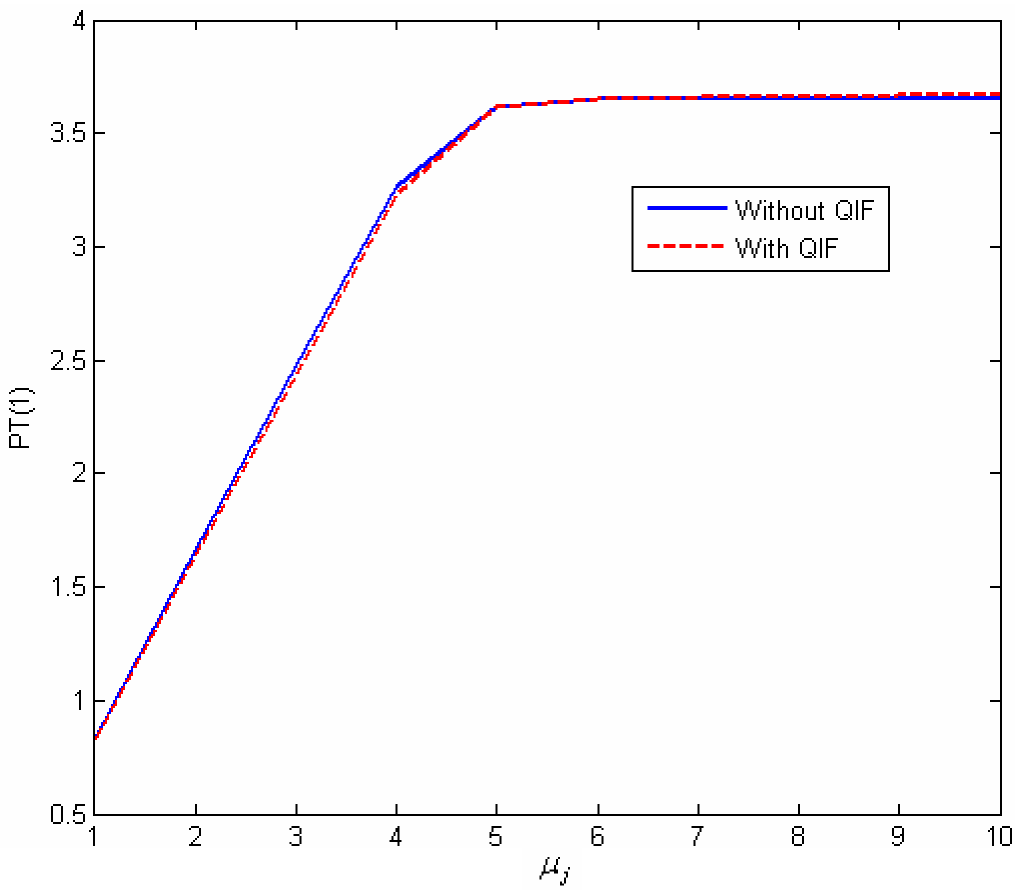

Figure 5.32 Effect of μ_j on *PT(1)* when $g_i = 0.4$ – With and without QIF

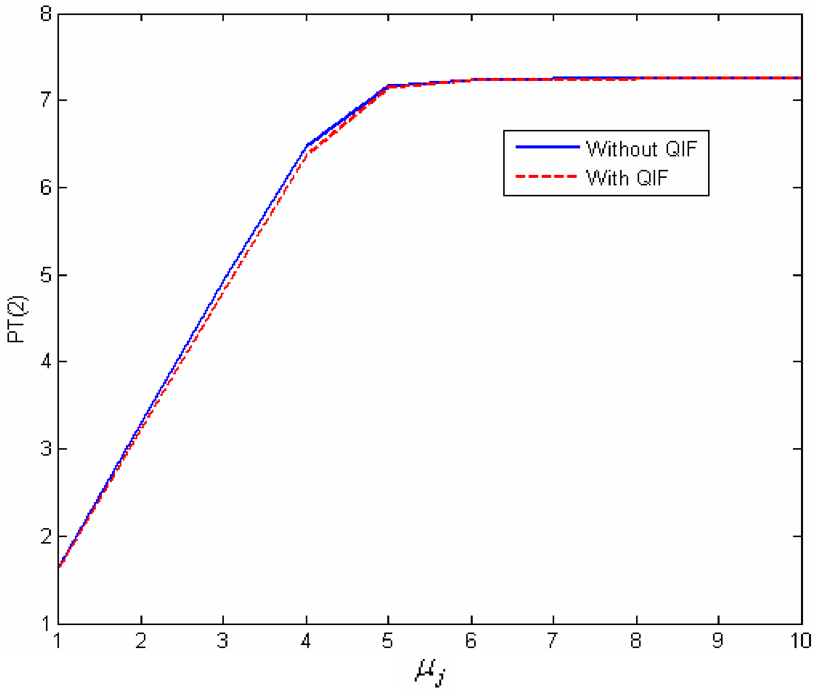

Figure 5.33 Effect of μ_j on *PT(2)* when $g_i = 0.4$ – With and without QIF

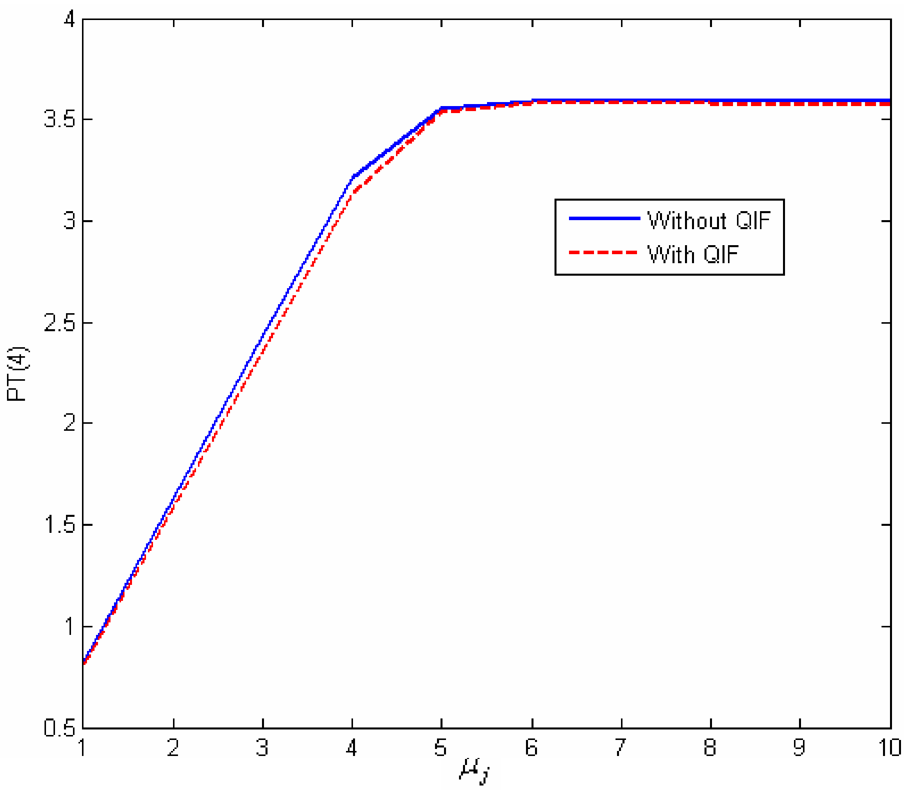

Figure 5.34 Effect of μ_j on *PT(4)* when $g_i = 0.4$ – With and without QIF

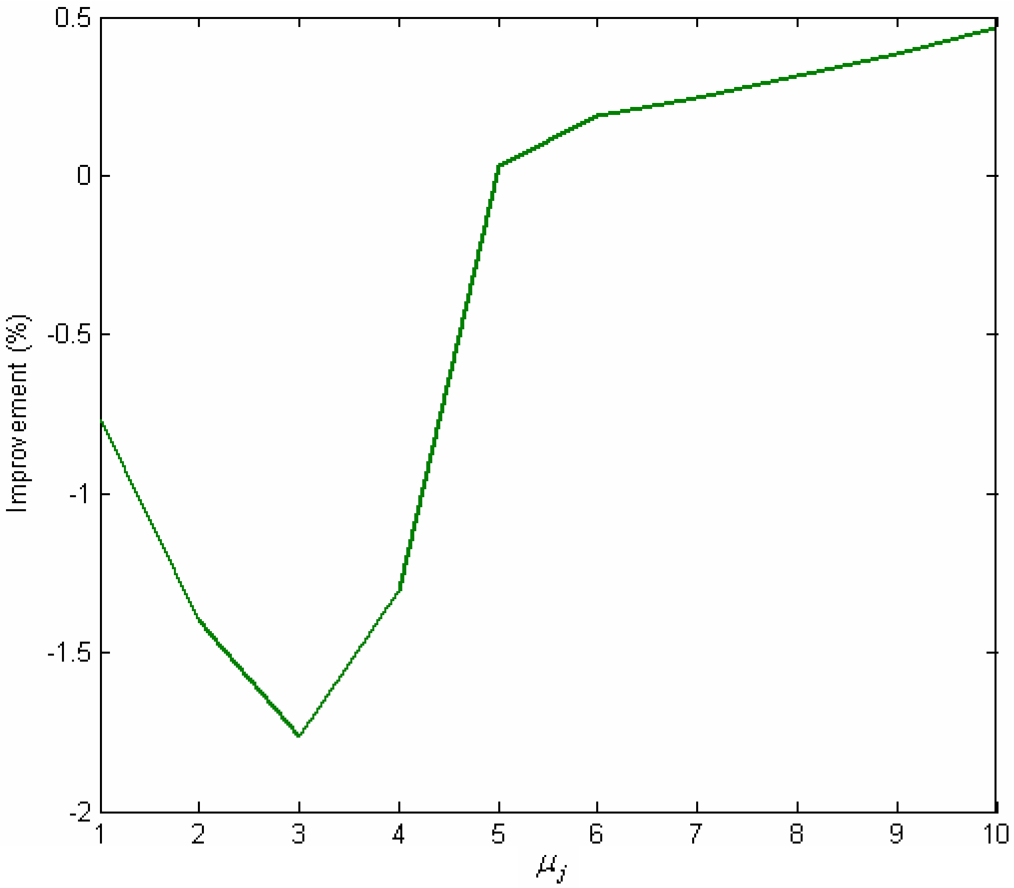

Figure 5.35 Improvement of throughput rate with QIF when $g_i = 0.4$

__Case__ 2: The parameters used in the experiments are in Table 5.9

Table 5.10. The machine parameters used in the second case of section 5.2.2

	Machine	***μ***	***r***	***f***	***p***	***g***	***N***
MAIN LINE	*1*		0,900	0,200	0,100	0,010	55
	2		0,900	0,200	0,100	0,010	55
	3		0,900	0,200	0,100	0,010	55
	4	**9,000**	**0,900**	**0,45**	**0,200**	**0,000**	55
	5	9,000	0,900	0,3	0,100	0,010	55
	6	9,000	0,900	0,3	0,100	0,010	55
	7	**9,000**	**0,900**	**0,35**	**0,200**	**0,000**	55
	8	9,000	0,900		0,010		55
REWORK LINE	*9*	9,000	0,950		0,010		55
	10	9,000	0,950		0,010		55
	11	9,000	0,950		0,010		55

Figures 5.36-5.39 are graphs of the performance of rework loop production system with both QIF and without QIF for $g_i = 0.01, \quad for \quad i = 1,2,3$. The throughput rate of the rework loop system, viz *PT(1)* both with QIF and without QIF is depicted in Figure 5.36.

Figure 5.37 and 5.38 demonstrate the effect of service rates of machines in Line 1 on throughput rate of serial Line 2, viz. *PT(2)* and serial Line 4, viz. *PT(4)*. Figure 5.39 shows the impact of service rates of machines in Line 1 on the throughput rate improvement when there is QIF in the system.

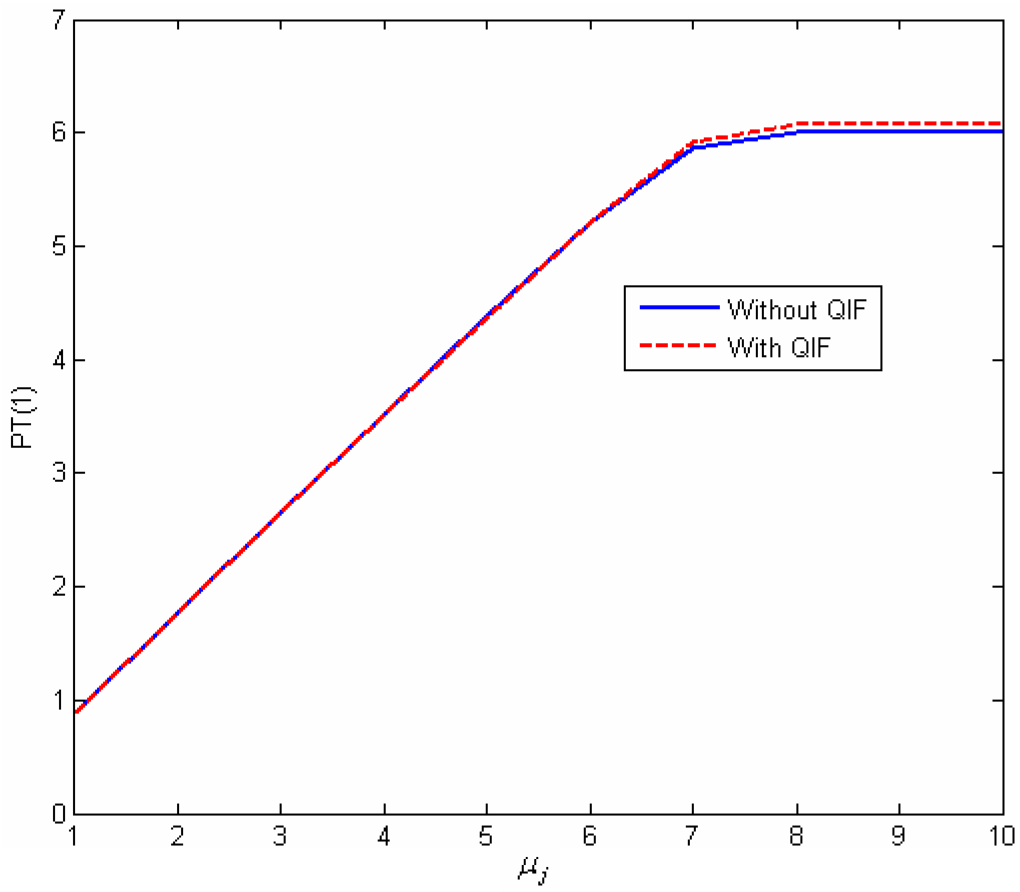

Figure 5.36 Effect of μ_j on *PT(1)* when $g_i = 0.01$ – With and without QIF

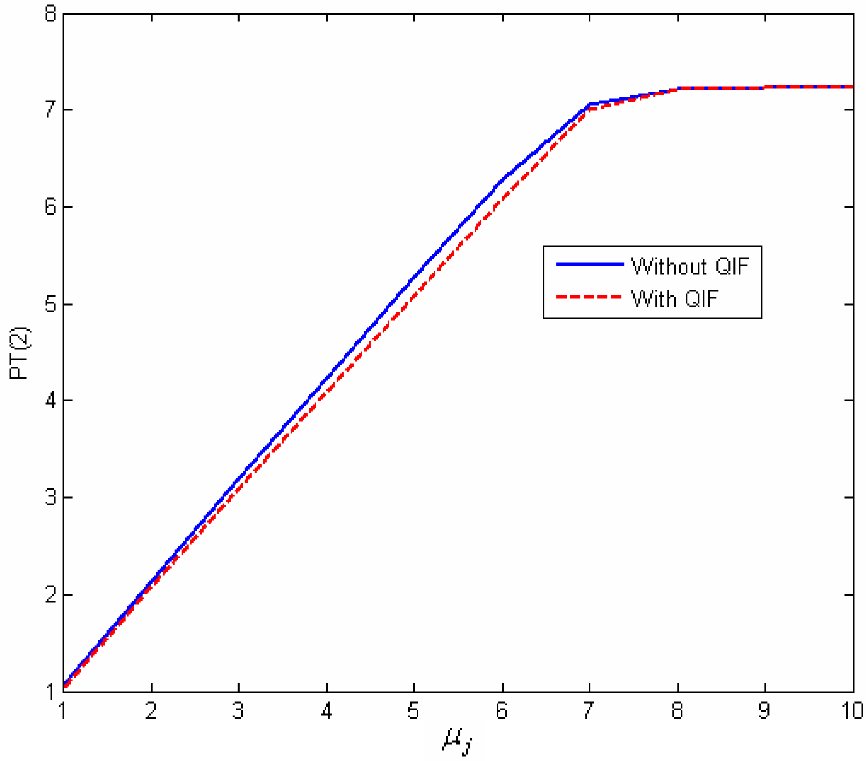

Figure 5.37 Effect of μ_j on *PT(2)* when $g_i = 0.01$ – With and without QIF

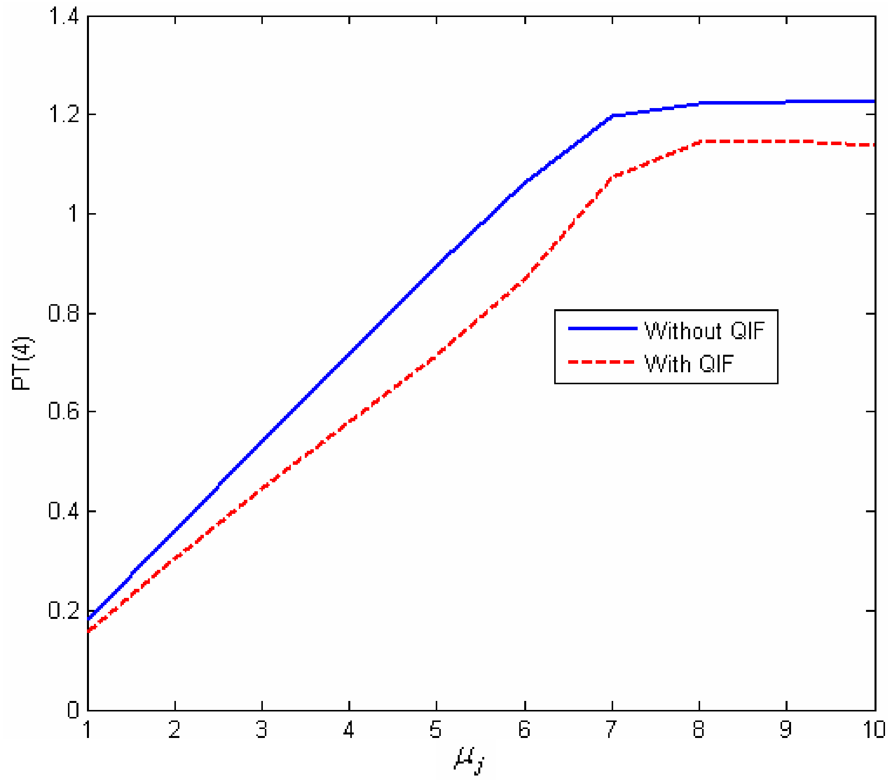

Figure 5.38 Effect of μ_j on *PT(4)* when $g_i = 0.01$ – With and without QIF

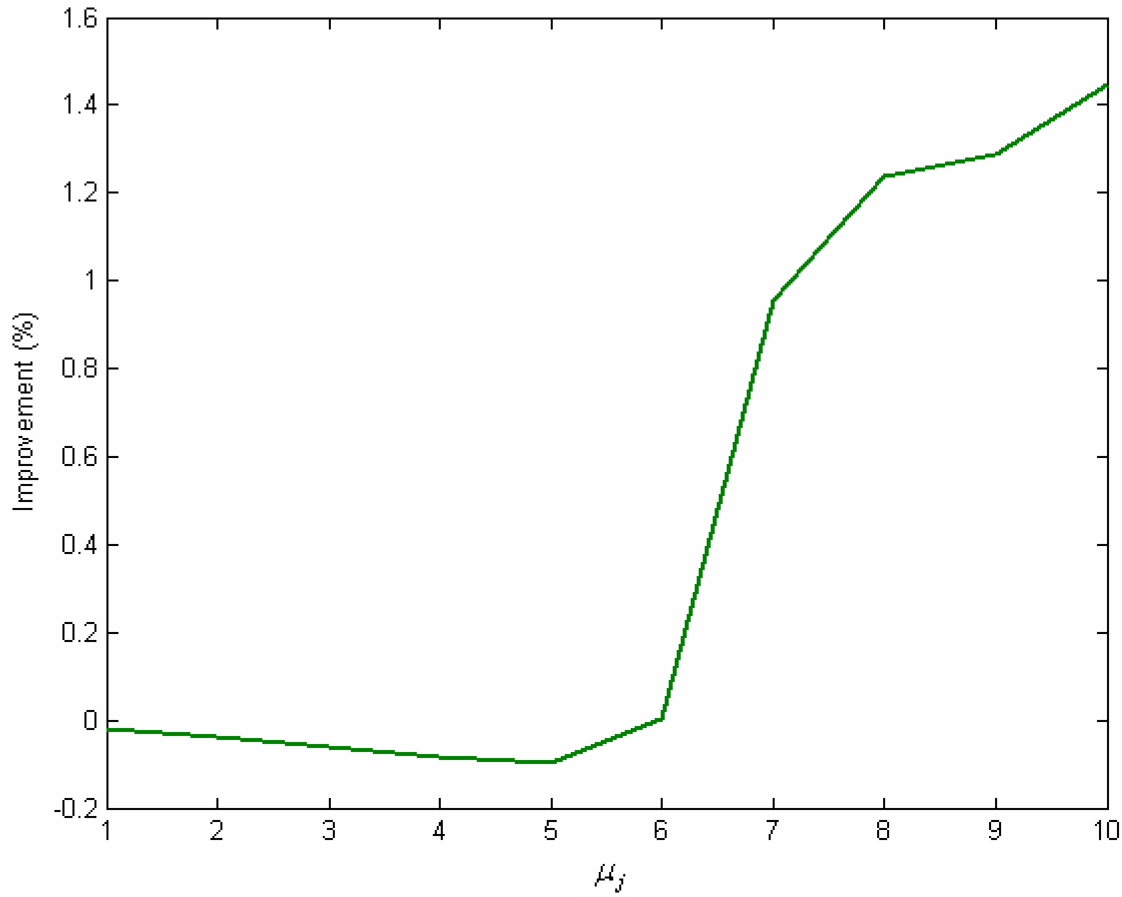

Figure 5.39 Improvement of throughput rate with QIF when $g_i = 0.01$

If the isolated production rate of some machine in a transfer line is much smaller than those of other machines, the system production rate will be mainly dominated by that bottleneck machine, and the efficiencies of other machines will approach zero. If the isolated total throughput rate of serial Line 1, which is calculated without any change in the parameters of the merge machine, is very close to the throughput rate of the rework system, then the QIF is harmful in terms of productivity.

In the rework loop system, the throughput rate of the system more depends on the efficiency of machines in Line 1 rather than the other decomposed serial Lines. When the isolated efficiency of Line 1 is significantly less than isolated efficiencies of other serial lines, stopping the machines in Line 1 so as to increase the yield does not increase the throughput rate because the bad parts are fixed very fast according to speed of Line 1.

5.3. Effect of the Number of Rework Lines on Production Systems with Rework Loops

In this section we concentrate on the effect of the number of rework lines on rework loop systems.

In production systems, not only an operational failure of a machine causes loss of capacity, but also a defective parts decrease the use of capacity since it occupies a robust machine. In the rework loops, when we increase the number of rework lines, the mean time to detect a bad part and fixing it decrases, so the use of capacity increases since the machines are not occupied by defective parts more often.

In this section, we compare three kind of production systems; a rework loop system with one rework line, a rework loop system with two rework lines and a rework loop system with three rework lines. Mainly, we consider a rework production system which has 14 machines for the main line and 3 machines for reworks.

In the rework loop system with one rework line, there is one split and one merge machine in the main line, and three machines in the rework line. M_3 is the merge machine and M_{13} is the split machine in the main line (Figure 5.40).

In the rework loop system with two rework lines, there are two split and two merge machines in the main line. In the first rework line, there is one machine, and in the second rework line there is two machine. M_3 is the first merge machine, M_5 is the first split machine, M_7 is the second merge machine and M_{13} is the second split machine (Figure 5.41).

In the rework loop system with three rework lines, there are three split and three merge machines in the main line. There are one machine in each rework line. M_3 is the first merge machine, M_5 is the first split machine, M_7 is the second merge machine, M_9 is second split machine, M_{11} is the third merge machine and M_{13} is the third split machine (Figure 5.42)

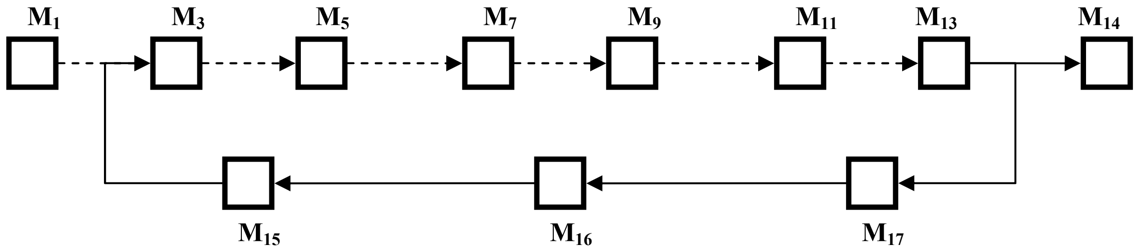

Figure 5.40 Rework loop system with one rework line

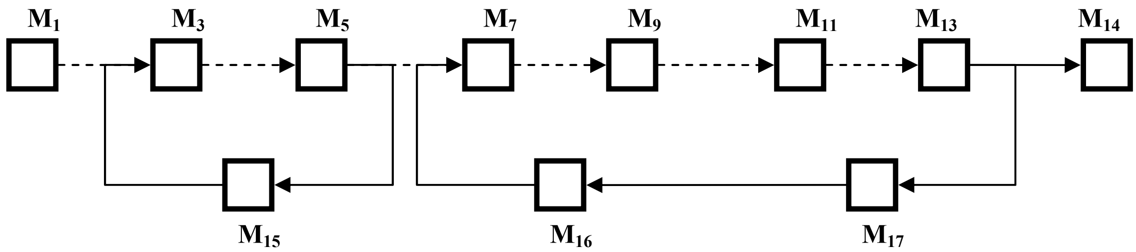

Figure 5.41 Rework loop system with two rework lines

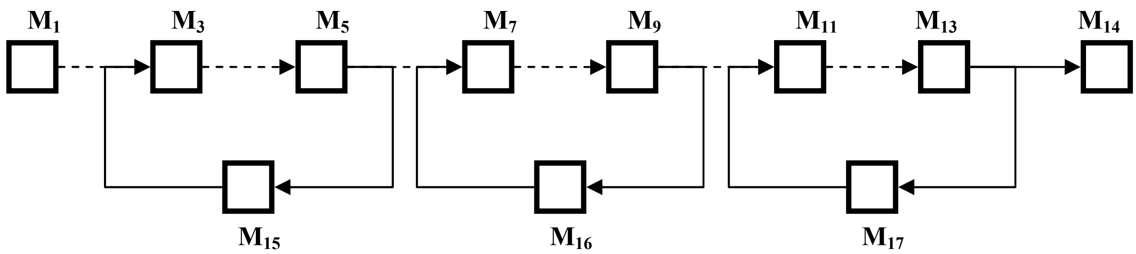

Figure 5.42 Rework loop system with three rework lines

We compare these three topologies within three cases. In each case, we make 10 experiments and calculate the throughput rates of the production topologies described above in all experiments.

In the experiments of the first case, the quality failure rates of the machines in the main line are increased from 0.01 to 0.45. The fix parameters that used in the experiments are in Table 5.11.

Table 5.11. The fix parameters used in the first case of section 5.3

	Machine	***μ***	***r***	***f***	***p***	***g***
MAIN LINE	*1*	4.000	0.900	0.35	0.100	
	2	4.000	0.900	0.350	0.100	
	3	4.000	0.900		0.100	
	4	4.000	0.900	0.350	0.100	
	5	4.000	0.900		0.100	
	6	4.000	0.900	0.350	0.100	
	7	4.000	0.900		0.100	
	8	4.000	0.900	0.350	0.100	
	9	4.000	0.900		0.100	
	10	4.000	0.900	0.350	0.100	
	11	4.000	0.900	0.350	0.100	
	12	4.000	0.900		0.100	
	13	4.000	0.900	0.350	0.100	
	14	4.000	0.900	0.350	0.100	
REWORK MACHINES	*15*	4.000	0.900	0.350	0.100	
	16	4.000	0.900	0.350	0.100	
	17	4.000	0.900	0.350	0.100	

Figure 5.43 demonstrate the comparison of the throughput rates of the rework production system with one rework line, the rework production system with two rework lines and the rework production system with three rework lines.

In the second case, we consider a main line which has a low yield. We alter the service rates of both the machines in main line and rework machines from 1 to 10 in the experiments and calculate the throughput rates of each topology. The fix parameters that used in the experiments are in Table 5.12.

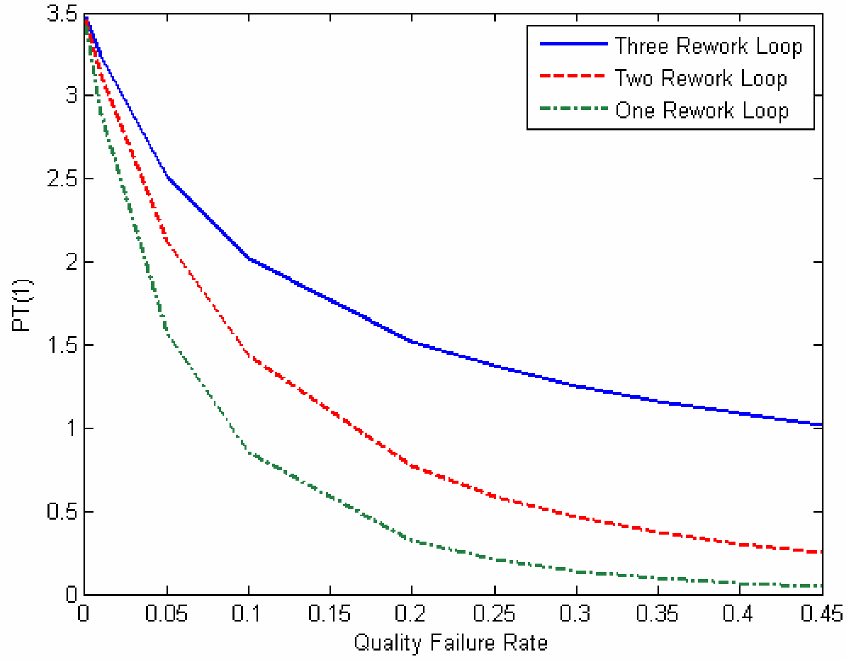

Figure 5.43 Effect of the number of rework lines for the first case of section 5.3

Table 5.12. The fix parameters used in the second case of section 5.3

	Machine	*μ*	*r*	*f*	*p*	*g*
MAIN LINE	*1*		0.900	0.350	0.100	0.350
	2		0.900	0.350	0.100	0.350
	3		0.900		0.100	
	4		0.900	0.350	0.100	0.350
	5		0.900		0.100	
	6		0.900	0.350	0.100	0.350
	7		0.900		0.100	
	8		0.900	0.350	0.100	0.350
	9		0.900		0.100	
	10		0.900	0.350	0.100	0.350
	11		0.900	0.350	0.100	
	12		0.900		0.100	0.350
	13		0.900	0.350	0.100	
	14		0.900	0.350	0.100	
REWORK MACHINES	*15*		0.900	0.350	0.100	
	16		0.900	0.350	0.100	
	17		0.900	0.350	0.100	

Figure 5.44 demonstrate the comparison of the throughput rates of the rework production system with one rework line ("One Rewok Loop"), the rework production system with two rework lines ("Two Rewok Loop") and the rework production system with three rework lines ("Three Rewok Loop") for the second case.

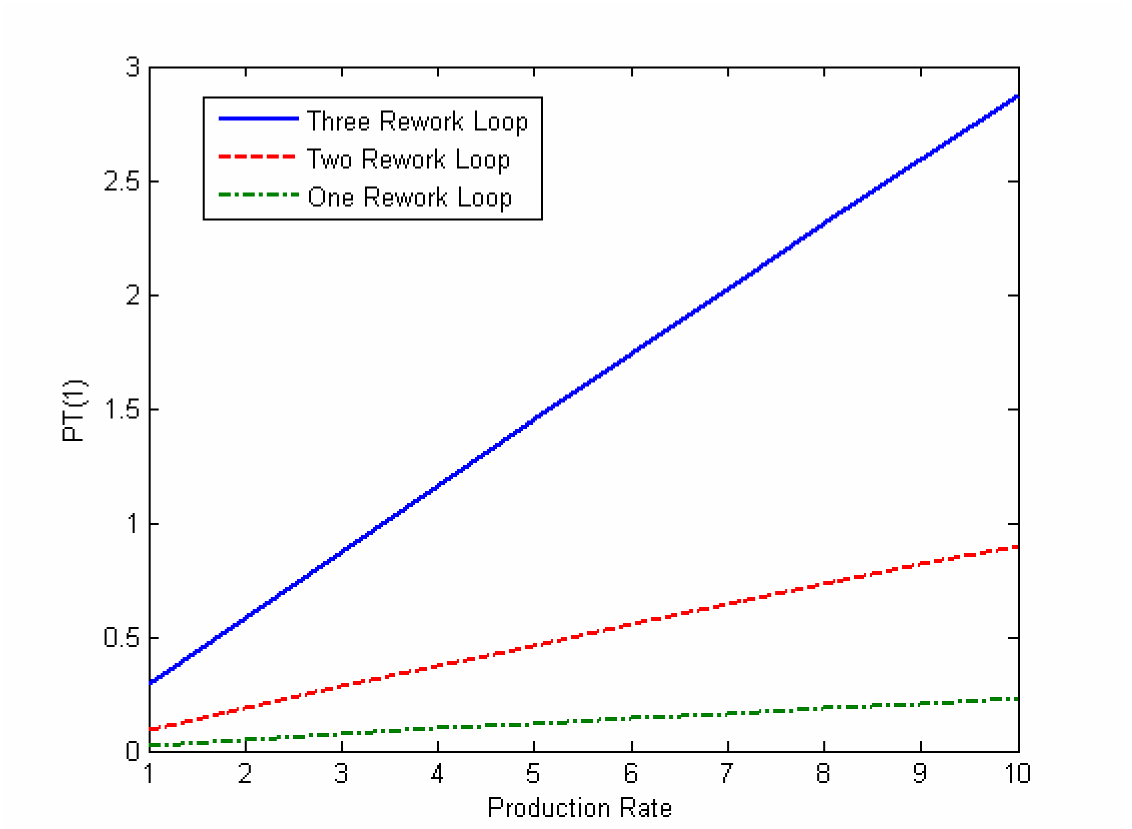

Figure 5.44 Effect of the number of rework lines for the second case of section 5.3

In the third case, we consider a main line which has a high yield. We alter the service rates of both the machines in main line and rework machines from 1 to 10 in the experiments and calculate the throughput rates of each topology. The fix parameters that used in the experiments are in Table 5.13.

Figure 5.45 demonstrate the comparison of the throughput rates of the rework production system with one rework line ("One Rewok Loop"), the rework production system with two rework lines ("Two Rewok Loop") and the rework production system with three rework lines ("Three Rewok Loop") for the third case.

Table 5.13. The fix parameters used in the third case of section 5.3

	Machine	***μ***	***r***	***f***	***p***	***g***
MAIN LINE	*1*		0.900	0.350	0.100	0.01
	2		0.900	0.350	0.100	0.01
	3		0.900		0.100	
	4		0.900	0.350	0.100	0.01
	5		0.900		0.100	
	6		0.900	0.350	0.100	0.01
	7		0.900		0.100	
	8		0.900	0.350	0.100	0.01
	9		0.900		0.100	
	10		0.900	0.350	0.100	0.01
	11		0.900	0.350	0.100	
	12		0.900		0.100	0.01
	13		0.900	0.350	0.100	
	14		0.900	0.350	0.100	
REWORK MACHINES	*15*		0.900	0.350	0.100	
	16		0.900	0.350	0.100	
	17		0.900	0.350	0.100	

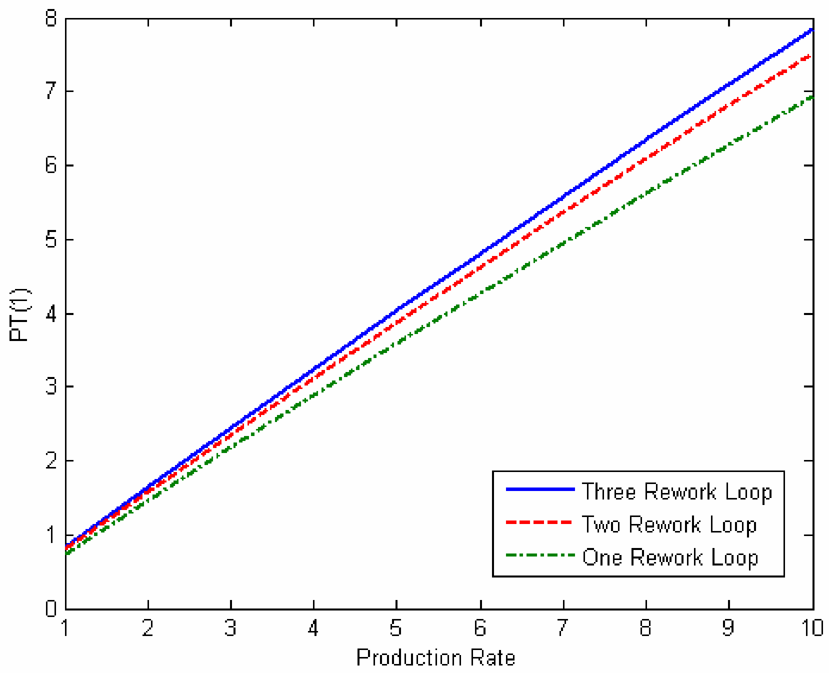

Figure 5.45 Effect of the number of rework lines for the third case of section 5.3

6. CONCLUSION

In this thesis, we have constructed solution algorithms for approximate performance analysis of rework loop production systems both with and without *quality information feedback* and we have also analyzed the rework loop production systems with multiple rework lines. In the literature, there exist solution techniques for the analysis of rework loop production systems. However, these works have not considered the machines having quality failures. They assign a constant rework rate in their algorithms. In this work we have stated the rework rate with respect to the quality failure rates of the machines.

Another purpose of this thesis is to compare rework loop production systems without QIF to rework system with QIF and serial transfer lines with QIF by making numerical studies. We also investigate the effect of the number of rework lines on the throughput rate of the rework loop system.

We decomposed the rework production system into four serial lines (Figure 4.1). The throughput rate of the system is equal to throughput rate of both serial Line 1 and Line 3. Also, the difference of the throughput rates of Line 2 and Line 4 gives us the throughput rate of the system. Having quality information feedback means having more inspections than otherwise. Therefore, machines tend to stop more frequently and the total throughput rate of the line decreases. As a result, whenever there is QIF in the rework system, the throughput rates of the Line 2 and Line 4 decrease. In contrast, the numerical experiments showed that the throughput rate of the serial Line 1 increases in more cases when threre is QIF in the rework system. This is due to the fact that, in the production systems with rework loops, if the production systems have QIF, the rework rate, i.e. α, will decrease since the machines produce good parts more frequently. Li (2004) states that the production rate of the rework loop system is a monotonically decreasing function of the α. For this reason the throughput rate of the system increases whenever we use QIF in the rework production system except one special situation.

In the experiments, it is observed that there are some critical cases which decrease the throughput of the rework system In the rework system, what comes out from the production system, i.e. the total amount of finished parts, is what serial Line 1 transfers into the production system. Therefore, quality information feedback decreases the throughput rate of the system in the cases where the serial Line 1 is the bottleneck of the system, since the machines in Line 1 stops more frequently and the total throughput rate of serial Line 1 decreases.

We have observed that when we increase the number of rework line in the production system the throughput rate of the system increases since the mean time to detect a bad part decreases and the use of capacity increases since the machines are not occupied by defective parts more often.

Mainly, we compared these production systems topologies in terms of the throughput rates. However, we must take into account the costs in comparison of these production topologies which is promised future research.

APPENDIX A: PARAMETERS AND RATES FOR REWORK LOOP SIMULATIONS STUDIES

Table A.1. Machine and buffer parameters for rework loop validation

CASE 1	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	2,00	0,80	0,30	0,30	0,25	40
	2	2,00	0,80	0,50	0,05	0,20	10
	3	2,00	0,65	0,35	0,10	0,15	45
	4	2,00	0,80	0,50	0,25	0,00	45
	5	2,00	0,80	0,15	0,05	0,05	40
	6	3,00	0,60	0,35	0,25	0,10	10
	7	2,00	0,90	0,45	0,15	0,00	35
	8	3,00	0,70	0,35	0,15		40
REWORK	*9*	3,00	0,75		0,05		45
	10	3,00	0,95		0,20		25
	11	3,00	0,95		0,05		25

CASE 2	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	3,00	0,75	0,30	0,20	0,25	50
	2	3,00	0,75	0,40	0,20	0,25	20
	3	3,00	0,95	0,35	0,20	0,10	10
	4	3,00	0,95	0,40	0,30	0,00	10
	5	2,00	0,85	0,30	0,30	0,25	20
	6	2,00	0,60	0,25	0,10	0,10	35
	7	3,00	0,65	0,50	0,40	0,00	10
	8	3,00	0,75	0,35	0,30		20
REWORK	*9*	3,00	0,60		0,15		40
	10	4,00	0,70		0,10		55
	11	3,00	0,75		0,15		30

CASE 3	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	3,00	0,70	0,40	0,05	0,01	45
	2	2,00	0,60	0,40	0,05	0,01	10
	3	2,00	0,80	0,40	0,10	0,25	40
	4	2,00	0,75	0,35	0,05	0,00	55
	5	2,00	0,75	0,50	0,15	0,30	30
	6	3,00	0,75	0,30	0,15	0,05	25
	7	3,00	0,85	0,35	0,30	0,00	30
	8	3,00	0,65	0,35	0,15		10
REWORK	*9*	3,00	0,60		0,05		40
	10	3,00	0,60		0,30		20
	11	3,00	0,60		0,30		30

CASE 4	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	2,00	0,70	0,50	0,10	0,01	55
	2	2,00	0,85	0,35	0,15	0,30	40
	3	2,00	0,80	0,40	0,30	0,00	20
	4	2,00	0,90	0,50	0,20	0,00	55
	5	2,00	0,90	0,40	0,05	0,20	20
	6	3,00	0,85	0,30	0,15	0,30	50
	7	3,00	0,70	0,35	0,30	0,00	10
	8	3,00	0,95	0,35	0,05		45
REWORK	*9*	4,00	0,60		0,30		55
	10	4,00	0,60		0,05		50
	11	2,00	0,75		0,25		55

CASE 5	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	2,00	0,80	0,35	0,25	0,01	50
	2	3,00	0,80	0,40	0,25	0,03	10
	3	4,00	0,70	0,30	0,25	0,20	40
	4	2,00	0,80	0,30	0,10	0,00	20
	5	3,00	0,95	0,15	0,15	0,01	55
	6	3,00	0,75	0,45	0,35	0,01	20
	7	3,00	0,85	0,30	0,10	0,00	20
	8	2,00	0,70	0,50	0,15		30
REWORK	*9*	3,00	0,85		0,20		35
	10	3,00	0,85		0,25		50
	11	2,00	0,60		0,15		50

CASE 6	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	3,00	0,95	0,15	0,10	0,20	20
	2	4,00	0,80	0,65	0,20	0,15	35
	3	3,00	0,85	0,45	0,25	0,20	25
	4	2,00	0,70	0,35	0,30	0,00	20
	5	3,00	0,95	0,30	0,25	0,15	55
	6	2,00	0,65	0,35	0,10	0,01	10
	7	3,00	0,65	0,20	0,15	0,00	55
	8	2,00	0,65	0,25	0,10		45
REWORK	*9*	2,00	0,75		0,25		50
	10	2,00	0,65		0,05		40
	11	2,00	0,75		0,30		35

Table A.2. Machine and buffer parameters for rework loop validation – continued

CASE 7	M	μ	r	f	p	g	N
MAIN LINE	1	3,00	0,75	0,45	0,05	0,30	55
	2	2,00	0,60	0,45	0,05	0,15	35
	3	3,00	0,70	0,50	0,20	0,05	40
	4	2,00	0,70	0,30	0,30	0,00	25
	5	2,00	0,65	0,45	0,05	0,10	50
	6	3,00	0,65	0,45	0,30	0,15	20
	7	2,00	0,65	0,45	0,10	0,00	20
	8	3,00	0,60	0,35	0,10		40
REWORK	9	2,00	0,75		0,30		40
	10	4,00	0,80		0,10		40
	11	3,00	0,80		0,30		30

CASE 8	M	μ	r	f	p	g	N
MAIN LINE	1	4,00	0,80	0,45	0,25	0,15	45
	2	4,00	0,75	0,45	0,35	0,10	35
	3	4,00	0,70	0,35	0,20	0,01	45
	4	4,00	0,60	0,45	0,10	0,00	25
	5	4,00	0,60	0,45	0,15	0,25	25
	6	4,00	0,75	0,40	0,25	0,25	25
	7	4,00	0,95	0,40	0,15	0,00	35
	8	4,00	0,65		0,05		50
REWORK	9	2,00	0,75		0,05		10
	10	2,00	0,70		0,30		40
	11	2,00	0,70		0,30		30

CASE 9	M	μ	r	f	p	g	N
MAIN LINE	1	3,00	0,85	0,50	0,05	0,10	55
	2	3,00	0,90	0,50	0,20	0,20	50
	3	3,00	0,95	0,45	0,25	0,25	35
	4	3,00	0,65	0,40	0,05	0,00	10
	5	3,00	0,65	0,35	0,30	0,30	20
	6	4,00	0,65	0,50	0,20	0,25	30
	7	2,00	0,95	0,50	0,15	0,00	40
	8	3,00	0,65	0,35	0,30		55
REWORK	9	2,00	0,70		0,30		10
	10	2,00	0,90		0,30		10
	11	3,00	0,85		0,25		35

CASE 10	M	μ	r	f	p	g	N
MAIN LINE	1	2,00	0,60	0,35	0,20	0,20	35
	2	3,00	0,95	0,35	0,10	0,05	10
	3	3,00	0,90	0,30	0,10	0,15	50
	4	2,00	0,60	0,45	0,25	0,00	55
	5	2,00	0,75	0,40	0,25	0,15	40
	6	3,00	0,70	0,50	0,15	0,05	10
	7	2,00	0,75	0,40	0,15	0,00	30
	8	4,00	0,75	0,35	0,10		55
REWORK	9	3,00	0,90		0,20		35
	10	2,00	0,75		0,05		25
	11	3,00	0,95		0,20		30

CASE 11	M	μ	r	f	p	g	N
MAIN LINE	1	4,00	0,80	0,30	0,15	0,15	20
	2	2,00	0,85	0,50	0,20	0,01	30
	3	4,00	0,60	0,40	0,10	0,25	40
	4	2,00	0,70	0,45	0,10	0,00	35
	5	3,00	0,65	0,35	0,25	0,25	40
	6	4,00	0,75	0,50	0,30	0,10	45
	7	3,00	0,75	0,35	0,05	0,00	10
	8	2,00	0,70	0,35	0,10		40
REWORK	9	2,00	0,70		0,20		25
	10	4,00	0,70		0,15		30
	11	2,00	0,75		0,05		35

CASE 12	M	μ	r	f	p	g	N
MAIN LINE	1	2,00	0,75	0,35	0,20	0,30	10
	2	3,00	0,60	0,30	0,15	0,00	25
	3	3,00	0,80	0,45	0,15	0,25	25
	4	2,00	0,80	0,50	0,30	0,00	25
	5	2,00	0,85	0,50	0,30	0,10	40
	6	2,00	0,95	0,50	0,05	0,01	55
	7	2,00	0,95	0,50	0,30	0,00	25
	8	4,00	0,75	0,35	0,05		55
REWORK	9	3,00	0,85		0,30		45
	10	2,00	0,70		0,05		10
	11	3,00	0,65		0,25		30

CASE 13	M	μ	r	f	p	g	N
MAIN LINE	1	3,00	0,85	0,45	0,20	0,15	40
	2	4,00	0,85	0,50	0,25	0,10	50
	3	3,00	0,80	0,45	0,10	0,20	35
	4	4,00	0,95	0,50	0,25	0,00	55
	5	2,00	0,80	0,50	0,20	0,05	25
	6	4,00	0,70	0,35	0,30	0,20	30
	7	2,00	0,70	0,35	0,10	0,00	30
	8	2,00	0,65	0,35	0,10		25
REWORK	9	2,00	0,60		0,15		45
	10	2,00	0,60		0,25		55
	11	4,00	0,65		0,20		10

CASE 14	M	μ	r	f	p	g	N
MAIN LINE	1	2,00	0,85	0,30	0,05	0,20	50
	2	2,00	0,85	0,30	0,30	0,00	30
	3	2,00	0,70	0,50	0,10	0,15	35
	4	3,00	0,80	0,30	0,20	0,00	45
	5	2,00	0,95	0,35	0,20	0,01	30
	6	4,00	0,65	0,35	0,05	0,30	10
	7	2,00	0,65	0,35	0,20	0,00	10
	8	3,00	0,75	0,35	0,25		45
REWORK	9	4,00	0,85		0,30		25
	10	2,00	0,75		0,20		10
	11	2,00	0,95		0,05		10

Table A.3. Machine and buffer parameters for rework loop validation – continued

CASE 15	M	μ	r	f	p	g	N	CASE 19	M	μ	r	f	p	g	N
MAIN LINE	*1*	2,00	0,95	0,30	0,15	0,00	35	MAIN LINE	*1*	3,00	0,90	0,40	0,10	0,04	35
	2	2,00	0,85	0,40	0,05	0,15	55		*2*	3,00	0,90	0,40	0,10	0,04	35
	3	2,00	0,65	0,40	0,10	0,20	20		*3*	3,00	0,90	0,40	0,10	0,10	35
	4	2,00	0,85	0,40	0,15	0,00	55		*4*	3,00	0,85	0,58	0,15	0,00	40
	5	3,00	0,75	0,45	0,10	0,00	55		*5*	3,00	0,90	0,40	0,10	0,03	45
	6	2,00	0,90	0,50	0,15	0,25	35		*6*	3,00	0,90	0,40	0,10	0,15	35
	7	3,00	0,70	0,50	0,30	0,00	10		*7*	3,00	0,90	0,40	0,03	0,00	40
	8	2,00	0,85	0,35	0,25		20		*8*	3,00	0,85	0,40	0,15		30
REWORK	*9*	2,00	0,65		0,25		40	REWORK	*9*	3,00	0,90		0,15		30
	10	2,00	0,85		0,20		35		*10*	3,00	0,90		0,10		50
	11	2,00	0,95		0,20		45		*11*	3,00	0,90		0,10		50

CASE 16	M	μ	r	f	p	g	N	CASE 20	M	μ	r	f	p	g	N
MAIN LINE	*1*	1,00	0,90	0,33	0,22	0,03	10	MAIN LINE	*1*	1,00	0,10	0,02	0,01	0,01	20
	2	1,00	0,53	0,33	0,25	0,22	10		*2*	1,00	0,20	0,02	0,01	0,01	20
	3	1,00	0,75	0,05	0,05	0,35	30		*3*	1,00	0,20	0,02	0,01	0,01	20
	4	1,00	0,82	0,35	0,25	0,00	10		*4*	1,00	0,20	0,02	0,01	0,00	20
	5	1,00	0,75	0,25	0,20	0,35	55		*5*	1,00	0,20	0,02	0,01	0,01	20
	6	3,00	0,70	0,58	0,30	0,00	50		*6*	1,00	0,10	0,02	0,01	0,01	20
	7	2,00	0,75	0,05	0,03	0,00	20		*7*	1,00	0,20	0,02	0,01	0,00	20
	8	1,00	0,53	0,35	0,15		45		*8*	1,00	0,20		0,01	0,00	20
REWORK	*9*	1,00	0,70		0,25		55	REWORK	*9*	1,00	0,50		0,10		10
	10	1,00	0,75		0,20		50		*10*	1,00	0,50		0,10		10
	11	1,00	0,70		0,10		40		*11*	1,00	0,50		0,10		10

CASE 17	M	μ	r	f	p	g	N	CASE 21	M	μ	r	f	p	g	N
MAIN LINE	*1*	4,00	0,85	0,15	0,02	0,00	55	MAIN LINE	*1*	4,00	0,90	0,50	0,20	0,20	25
	2	1,00	0,75	0,33	0,15	0,05	20		*2*	4,00	0,85	0,40	0,25	0,20	20
	3	1,00	0,66	0,58	0,15	0,35	10		*3*	4,00	0,90	0,60	0,20	0,10	35
	4	1,00	0,90	0,25	0,20	0,00	45		*4*	4,00	0,80	0,60	0,25	0,00	28
	5	1,00	0,90	0,35	0,30	0,35	20		*5*	4,00	0,90	0,40	0,20	0,25	30
	6	3,00	0,75	0,33	0,05	0,10	35		*6*	4,00	0,80	0,55	0,52	0,30	25
	7	3,00	0,75	0,33	0,25	0,00	30		*7*	4,00	0,80	0,60	0,22	0,00	30
	8	1,00	0,66	0,35	0,20		20		*8*	4,00	0,90		0,25		35
REWORK	*9*	3,00	0,82		0,10		20	REWORK	*9*	4,00	0,20		0,85		45
	10	4,00	0,66		0,05		55		*10*	4,00	0,20		0,70		25
	11	4,00	0,75		0,30		25		*11*	4,00	0,20		0,80		30

CASE 18	M	μ	r	f	p	g	N	CASE 22	M	μ	r	f	p	g	N
MAIN LINE	*1*	1,00	0,90	0,25	0,22	0,35	25	MAIN LINE	*1*	3,00	0,10	0,03	0,02	0,10	30
	2	1,00	0,75	0,33	0,15	0,35	25		*2*	3,00	0,10	0,03	0,02	0,01	30
	3	2,00	0,82	0,25	0,20	0,22	20		*3*	3,00	0,10	0,03	0,02	0,10	30
	4	1,00	0,85	0,58	0,15	0,00	40		*4*	3,00	0,10	0,30	0,02	0,00	30
	5	1,00	0,90	0,45	0,20	0,03	45		*5*	3,00	0,10	0,03	0,02	0,10	30
	6	3,00	0,53	0,15	0,15	0,05	35		*6*	3,00	0,10	0,03	0,02	0,01	30
	7	1,00	0,53	0,05	0,03	0,00	40		*7*	3,00	0,10	0,30	0,02	0,00	30
	8	1,00	0,85	0,35	0,15		30		*8*	3,00	0,10		0,02	0,00	30
REWORK	*9*	1,00	0,53		0,15		30	REWORK	*9*	3,00	0,50		0,10		20
	10	4,00	0,82		0,30		50		*10*	3,00	0,50		0,10		20
	11	2,00	0,75		0,30		50		*11*	3,00	0,50		0,10		20

Table A.4. Machine and buffer parameters for rework loop validation – continued

CASE 23	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	4,00	0,90	0,30	0,02	0,40	25
	2	4,00	0,85	0,20	0,03	0,40	20
	3	4,00	0,90	0,30	0,10	0,50	35
	4	4,00	0,80	0,20	0,10	0,00	28
	5	4,00	0,90	0,20	0,02	0,50	30
	6	4,00	0,80	0,20	0,05	0,40	25
	7	4,00	0,80	0,20	0,12	0,00	30
	8	4,00	0,90		0,12		35
REWORK	*9*	4,00	0,80		0,10		45
	10	4,00	0,80		0,10		25
	11	4,00	0,80		0,10		30

CASE 24	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	4,00	0,82	0,25	0,04	0,05	35
	2	2,00	0,66	0,15	0,10	0,35	40
	3	1,00	0,75	0,25	0,05	0,10	50
	4	1,00	0,85	0,33	0,05	0,00	40
	5	1,00	0,53	0,25	0,10	0,35	10
	6	1,00	0,70	0,58	0,10	0,22	50
	7	2,00	0,75	0,58	0,05	0,00	50
	8	1,00	0,66		0,10		40
REWORK	*9*	2,00	0,53		0,25		10
	10	1,00	0,70		0,05		50
	11	1,00	0,82		0,20		35

CASE 25	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	4,00	0,90	0,25	0,22	0,05	35
	2	4,00	0,90	0,50	0,30	0,35	30
	3	2,00	0,85	0,50	0,40	0,10	35
	4	4,00	0,85	0,35	0,30	0,00	25
	5	3,00	0,85	0,15	0,15	0,35	40
	6	3,00	0,90	0,33	0,30	0,22	45
	7	1,00	0,66	0,35	0,30	0,00	20
	8	2,00	0,75		0,15		30
REWORK	*9*	3,00	0,90		0,25		30
	10	1,00	0,90		0,30		40
	11	2,00	0,85		0,30		45

CASE 26	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	2,00	0,90	0,00	0,00	0,40	55
	2	2,00	0,90	0,00	0,00	0,40	55
	3	2,00	0,90	0,00	0,00	0,40	55
	4	5,00	0,90	0,45	0,01	0,00	55
	5	4,00	0,90	0,33	0,02	0,01	50
	6	4,00	0,90	0,33	0,02	0,00	35
	7	5,00	0,90	0,35	0,03	0,00	45
	8	4,00	0,90		0,01		60
REWORK	*9*	3,00	0,95		0,01		50
	10	3,00	0,95		0,01		50
	11	3,00	0,95		0,01		50

CASE 27	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	3,00	0,70	0,33	0,03	0,10	50
	2	2,00	0,82	0,58	0,20	0,01	30
	3	2,00	0,75	0,35	0,30	0,10	35
	4	2,00	0,53	0,05	0,02	0,00	10
	5	1,00	0,85	0,58	0,25	0,22	25
	6	4,00	0,82	0,25	0,25	0,03	30
	7	2,00	0,70	0,05	0,03	0,00	50
	8	2,00	0,53		0,10		35
REWORK	*9*	1,00	0,70		0,30		50
	10	2,00	0,90		0,05		45
	11	2,00	0,70		0,15		30

CASE 28	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	4,00	0,53	0,33	0,03	0,10	40
	2	2,00	0,66	0,58	0,20	0,10	35
	3	2,00	0,70	0,15	0,03	0,35	50
	4	4,00	0,66	0,05	0,02	0,00	40
	5	2,00	0,45	0,58	0,25	0,01	40
	6	3,00	0,55	0,25	0,25	0,10	30
	7	2,00	0,75	0,05	0,03	0,00	25
	8	1,00	0,70		0,20		55
REWORK	*9*	2,00	0,30		0,50		35
	10	1,00	0,30		0,20		25
	11	4,00	0,30		0,20		45

CASE 29	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	2,00	0,66	0,58	0,24	0,03	30
	2	1,00	0,75	0,58	0,15	0,00	20
	3	1,00	0,75	0,33	0,15	0,10	10
	4	3,00	0,66	0,50	0,30	0,00	20
	5	1,00	0,70	0,05	0,03	0,35	55
	6	2,00	0,53	0,58	0,15	0,03	10
	7	4,00	0,53	0,33	0,30	0,00	50
	8	4,00	0,82		0,15		35
REWORK	*9*	3,00	0,82		0,10		35
	10	1,00	0,66		0,20		50
	11	3,00	0,75		0,05		40

CASE 30	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	1,00	0,70	0,25	0,10	0,01	45
	2	1,00	0,82	0,15	0,15	0,05	45
	3	1,00	0,90	0,33	0,30	0,15	55
	4	2,00	0,70	0,25	0,10	0,00	10
	5	1,00	0,70	0,33	0,15	0,05	50
	6	3,00	0,75	0,33	0,15	0,01	25
	7	1,00	0,66	0,33	0,25	0,00	45
	8	1,00	0,82		0,30		50
REWORK	*9*	1,00	0,66		0,30		20
	10	1,00	0,70		0,25		55
	11	1,00	0,82		0,05		45

APPENDIX B: PARAMETERS AND RATES FOR MULTIPLE LOOP SIMULATIONS STUDIES

Table B.1. Machine and buffer parameters for multiple loop validation

CASE 1	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	2.00	0.80	0.25	0.20	0.30	30
	2	3.00	0.85	0.40	0.25	0.50	40
	3	2.00	0.80	0.45	0.01	0.00	35
	4	2.00	0.80	0.40	0.15	0.01	30
	5	2.00	0.95	0.30	0.20	0.00	30
	6	3.00	0.80	0.30	0.25	0.40	45
	7	2.00	0.85	0.40	0.12	0.00	35
	8	2.00	0.70	0.40	0.12	0.20	35
	9	3.00	0.90	0.40	0.10	0.00	45
	10	2.00	0.80	0.30	0.10	0.20	25
	11	2.00	0.95	0.40	0.15	0.00	30
	12	3.00	0.60	0.35	0.10	0.20	30
	13	2.00	0.80	0.40	0.25	0.00	45
	14	2.00	0.95	0.50	0.10	0.00	50
REWORK 1	*15*	4.00	0.80	0.40	0.20	0.00	45
REWORK 2	*16*	4.00	0.80	0.40	0.10	0.00	45
REWORK 3	*17*	4.00	0.75	0.40	0.15	0.00	30
							40
							45

CASE 3	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	4.00	0.60	0.15	0.10	0.15	35
	2	4.00	0.75	0.30	0.30	0.15	10
	3	3.00	0.75	0.45	0.10	0.00	45
	4	3.00	0.95	0.40	0.25	0.01	45
	5	3.00	0.35	0.40	0.20	0.00	50
	6	3.00	0.65	0.25	0.20	0.05	40
	7	4.00	0.55	0.40	0.30	0.00	40
	8	3.00	0.35	0.40	0.10	0.00	55
	9	3.00	0.45	0.40	0.30	0.00	35
	10	4.00	0.65	0.50	0.10	0.30	45
	11	4.00	0.75	0.35	0.20	0.00	50
	12	3.00	0.90	0.15	0.10	0.20	50
	13	3.00	0.85	0.25	0.05	0.00	35
	14	2.00	0.65	0.35	0.05	0.00	10
REWORK 1	*15*	3.00	0.95	0.40	0.05	0.00	10
REWORK 2	*16*	4.00	0.80	0.35	0.05	0.00	20
REWORK 3	*17*	4.00	0.65	0.30	0.03	0.00	30
							55
							40

CASE 2	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	2.00	0.65	0.50	0.20	0.00	10
	2	4.00	0.60	0.25	0.15	0.15	40
	3	2.00	0.45	0.30	0.30	0.00	20
	4	2.00	0.65	0.30	0.15	0.10	40
	5	2.00	0.35	0.40	0.01	0.00	55
	6	4.00	0.80	0.45	0.25	0.05	35
	7	4.00	0.45	0.40	0.20	0.00	50
	8	2.00	0.55	0.45	0.15	0.10	35
	9	2.00	0.90	0.45	0.15	0.00	50
	10	2.00	0.45	0.50	0.15	0.15	25
	11	3.00	0.65	0.30	0.15	0.00	55
	12	3.00	0.65	0.35	0.25	0.30	55
	13	4.00	0.55	0.40	0.25	0.00	10
	14	2.00	0.95	0.30	0.25	0.00	30
REWORK 1	*15*	3.00	0.95		0.30	0.00	40
REWORK 2	*16*	4.00	0.75		0.01	0.00	10
REWORK 3	*17*	3.00	0.90		0.15	0.00	55
							30
							10

CASE 4	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	2.00	0.80	0.50	0.20	0.05	25
	2	2.00	0.65	0.15	0.05	0.15	20
	3	4.00	0.90	0.50	0.15	0.00	40
	4	2.00	0.60	0.40	0.25	0.15	30
	5	2.00	0.55	0.15	0.01	0.00	35
	6	2.00	0.80	0.40	0.10	0.25	40
	7	2.00	0.75	0.50	0.01	0.00	10
	8	4.00	0.55	0.40	0.30	0.25	45
	9	4.00	0.80	0.50	0.30	0.00	35
	10	2.00	0.90	0.45	0.20	0.05	10
	11	4.00	0.75	0.30	0.25	0.00	10
	12	2.00	0.90	0.25	0.05	0.20	40
	13	3.00	0.60	0.45	0.01	0.00	20
	14	2.00	0.80	0.35	0.01	0.00	30
REWORK 1	*15*	4.00	0.65		0.15	0.00	10
REWORK 2	*16*	2.00	0.65		0.05	0.00	55
REWORK 3	*17*	2.00	0.80		0.10	0.00	50
							20
							35

Table B.2. Machine and buffer parameters for multiple loop validation – continued

CASE 5	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	2.00	0.65	0.15	0.05	0.30	50
	2	3.00	0.45	0.50	0.01	0.20	35
	3	4.00	0.80	0.40	0.25	0.00	40
	4	4.00	0.95	0.15	0.03	0.15	40
	5	2.00	0.90	0.50	0.10	0.00	45
	6	2.00	0.35	0.30	0.20	0.10	20
	7	4.00	0.55	0.45	0.15	0.00	10
	8	4.00	0.45	0.35	0.30	0.15	45
	9	3.00	0.80	0.40	0.20	0.00	10
	10	4.00	0.80	0.30	0.10	0.00	40
	11	3.00	0.75	0.50	0.30	0.00	55
	12	2.00	0.60	0.30	0.01	0.20	55
	13	3.00	0.80	0.35	0.25	0.00	10
	14	3.00	0.95	0.35	0.20	0.00	20
REWORK 1	*15*	3.00	0.55		0.20	0.00	20
REWORK 2	*16*	3.00	0.75		0.30	0.00	35
REWORK 3	*17*	3.00	0.65		0.01	0.00	40
							50
							45

CASE 7	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	2.00	0.75	0.30	0.30	0.05	55
	2	3.00	0.55	0.28	0.20	0.05	10
	3	4.00	0.55	0.25	0.10	0.00	45
	4	2.00	0.35	0.35	0.01	0.25	55
	5	2.00	0.55	0.40	0.25	0.00	20
	6	2.00	0.35	0.35	0.10	0.01	50
	7	3.00	0.90	0.50	0.25	0.00	25
	8	3.00	0.65	0.40	0.25	0.01	45
	9	3.00	0.55	0.40	0.20	0.00	20
	10	3.00	0.65	0.15	0.02	0.30	35
	11	3.00	0.55	0.40	0.30	0.00	10
	12	3.00	0.65	0.50	0.15	0.01	10
	13	4.00	0.60	0.40	0.05	0.00	10
	14	2.00	0.65	0.30	0.20	0.00	55
REWORK 1	*15*	3.00	0.80		0.20	0.00	55
REWORK 2	*16*	2.00	0.80		0.20	0.00	55
REWORK 3	*17*	3.00	0.35		0.30	0.00	20
							10
							35

CASE 6	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	4.00	0.55	0.30	0.01	0.20	45
	2	2.00	0.55	0.50	0.10	0.01	55
	3	4.00	0.35	0.30	0.15	0.00	20
	4	2.00	0.80	0.50	0.01	0.05	10
	5	4.00	0.60	0.35	0.10	0.00	30
	6	4.00	0.35	0.30	0.30	0.15	35
	7	4.00	0.75	0.45	0.30	0.00	10
	8	4.00	0.35	0.25	0.25	0.00	40
	9	4.00	0.65	0.25	0.20	0.00	25
	10	3.00	0.60	0.15	0.15	0.10	50
	11	3.00	0.65	0.35	0.25	0.00	45
	12	3.00	0.35	0.25	0.20	0.05	30
	13	3.00	0.65	0.35	0.10	0.00	50
	14	2.00	0.75	0.15	0.01	0.00	10
REWORK 1	*15*	2.00	0.55		0.01	0.00	30
REWORK 2	*16*	3.00	0.90		0.20	0.00	30
REWORK 3	*17*	3.00	0.90		0.30	0.00	45
							45
							25

CASE 8	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	4.00	0.55	0.25	0.10	0.01	35
	2	2.00	0.35	0.40	0.30	0.25	40
	3	3.00	0.55	0.45	0.01	0.00	50
	4	3.00	0.35	0.35	0.05	0.30	25
	5	3.00	0.35	0.35	0.30	0.00	35
	6	2.00	0.65	0.35	0.30	0.10	55
	7	3.00	0.60	0.50	0.25	0.00	10
	8	4.00	0.55	0.50	0.01	0.10	10
	9	3.00	0.95	0.50	0.25	0.00	35
	10	3.00	0.55	0.45	0.05	0.05	35
	11	3.00	0.45	0.45	0.10	0.00	55
	12	2.00	0.55	0.35	0.30	0.20	25
	13	4.00	0.80	0.30	0.30	0.00	35
	14	2.00	0.65	0.50	0.10	0.00	55
REWORK 1	*15*	3.00	0.80		0.20	0.00	30
REWORK 2	*16*	2.00	0.45		0.15	0.00	35
REWORK 3	*17*	2.00	0.65		0.05	0.00	10
							30
							35

Table B.3. Machine and buffer parameters for multiple loop validation – continued

CASE 9	*M*	*μ*	*r*	*f*	*p*	*g*	*N*	CASE 11	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	4.00	0.65	0.30	0.02	0.30	50	MAIN LINE	*1*	3.00	0.45	0.45	0.20	0.20	20
	2	4.00	0.65	0.35	0.05	0.00	45		*2*	3.00	0.60	0.35	0.05	0.05	10
	3	4.00	0.90	0.25	0.03	0.00	10		*3*	4.00	0.65	0.45	0.10	0.00	10
	4	4.00	0.75	0.25	0.01	0.25	10		*4*	2.00	0.55	0.50	0.25	0.30	55
	5	4.00	0.90	0.50	0.15	0.00	30		*5*	2.00	0.35	0.35	0.10	0.00	10
	6	4.00	0.35	0.30	0.15	0.05	30		*6*	3.00	0.60	0.45	0.01	0.20	50
	7	4.00	0.90	0.40	0.05	0.00	50		*7*	3.00	0.95	0.35	0.15	0.00	30
	8	4.00	0.35	0.40	0.25	0.00	40		*8*	3.00	0.90	0.35	0.20	0.01	10
	9	4.00	0.55	0.40	0.05	0.00	20		*9*	2.00	0.80	0.50	0.15	0.00	35
	10	4.00	0.45	0.40	0.20	0.00	20		*10*	3.00	0.60	0.30	0.30	0.30	35
	11	4.00	0.60	0.50	0.20	0.00	35		*11*	4.00	0.60	0.40	0.20	0.00	20
	12	4.00	0.55	0.15	0.15	0.30	50		*12*	3.00	0.35	0.30	0.01	0.05	55
	13	4.00	0.90	0.40	0.10	0.00	50		*13*	2.00	0.90	0.40	0.25	0.00	25
	14	4.00	0.55	0.40	0.15	0.00	10		*14*	3.00	0.95	0.25	0.30	0.00	50
REWORK 1	*15*	3.00	0.35		0.15	0.00	10	REWORK 1	*15*	2.00	0.60		0.30	0.00	10
REWORK 2	*16*	3.00	0.60		0.20	0.00	35	REWORK 2	*16*	2.00	0.65		0.20	0.00	25
REWORK 3	*17*	3.00	0.65		0.05	0.00	50	REWORK 3	*17*	3.00	0.95		0.15	0.00	25
							10								50
							40								30

CASE 10	*M*	*μ*	*r*	*f*	*p*	*g*	*N*	CASE 12	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	3.00	0.45	0.45	0.30	0.15	10	MAIN LINE	*1*	4.00	0.60	0.45	0.30	0.10	40
	2	3.00	0.55	0.30	0.25	0.05	10		*2*	3.00	0.95	0.45	0.01	0.30	55
	3	4.00	0.65	0.35	0.15	0.00	55		*3*	2.00	0.90	0.40	0.15	0.00	40
	4	3.00	0.65	0.35	0.15	0.30	50		*4*	2.00	0.75	0.25	0.25	0.05	30
	5	2.00	0.90	0.25	0.01	0.00	10		*5*	4.00	0.45	0.35	0.30	0.00	25
	6	3.00	0.95	0.40	0.20	0.00	10		*6*	4.00	0.55	0.40	0.01	0.15	45
	7	2.00	0.75	0.15	0.01	0.00	10		*7*	2.00	0.45	0.35	0.30	0.00	30
	8	4.00	0.35	0.15	0.05	0.01	45		*8*	2.00	0.90	0.40	0.01	0.10	55
	9	4.00	0.90	0.40	0.10	0.00	40		*9*	2.00	0.95	0.15	0.04	0.00	45
	10	3.00	0.55	0.40	0.01	0.01	10		*10*	2.00	0.65	0.30	0.01	0.01	25
	11	2.00	0.35	0.35	0.30	0.00	30		*11*	3.00	0.45	0.35	0.15	0.00	30
	12	3.00	0.55	0.15	0.10	0.10	55		*12*	3.00	0.90	0.35	0.30	0.30	55
	13	2.00	0.90	0.45	0.20	0.00	10		*13*	3.00	0.45	0.40	0.10	0.00	55
	14	2.00	0.95	0.50	0.05	0.00	10		*14*	2.00	0.55	0.45	0.30	0.00	30
REWORK 1	*15*	2.00	0.75		0.05	0.00	45	REWORK 1	*15*	3.00	0.35		0.05	0.00	25
REWORK 2	*16*	3.00	0.75		0.01	0.00	50	REWORK 2	*16*	3.00	0.95		0.01	0.00	50
REWORK 3	*17*	3.00	0.55		0.15	0.00	35	REWORK 3	*17*	3.00	0.75		0.30	0.00	50
							25								10
							25								40

Table B.4. Machine and buffer parameters for multiple loop validation – continued

CASE 13	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	4.00	0.80	0.45	0.15	0.10	50
	2	3.00	0.80	0.15	0.05	0.30	50
	3	4.00	0.60	0.30	0.01	0.00	45
	4	2.00	0.80	0.50	0.15	0.30	45
	5	4.00	0.35	0.50	0.10	0.00	30
	6	4.00	0.75	0.25	0.20	0.30	30
	7	3.00	0.90	0.40	0.25	0.00	40
	8	2.00	0.45	0.40	0.15	0.00	50
	9	4.00	0.35	0.45	0.01	0.00	20
	10	3.00	0.45	0.30	0.10	0.20	50
	11	2.00	0.35	0.40	0.20	0.00	55
	12	2.00	0.35	0.50	0.10	0.25	30
	13	2.00	0.55	0.35	0.25	0.00	35
	14	2.00	0.35	0.45	0.01	0.00	20
REWORK 1	*15*	3.00	0.35		0.10	0.00	30
REWORK 2	*16*	3.00	0.35		0.20	0.00	20
REWORK 3	*17*	3.00	0.35		0.15	0.00	35
							40
							30

CASE 15	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	2.00	0.45	0.45	0.10	0.10	55
	2	2.00	0.90	0.35	0.20	0.20	55
	3	2.00	0.45	0.25	0.10	0.00	35
	4	2.00	0.35	0.35	0.25	0.25	30
	5	2.00	0.95	0.40	0.15	0.00	30
	6	4.00	0.75	0.40	0.10	0.05	55
	7	3.00	0.35	0.35	0.01	0.00	40
	8	3.00	0.90	0.45	0.25	0.15	25
	9	3.00	0.80	0.30	0.20	0.00	40
	10	2.00	0.95	0.50	0.10	0.05	35
	11	3.00	0.60	0.15	0.01	0.00	50
	12	2.00	0.55	0.25	0.20	0.01	45
	13	3.00	0.75	0.40	0.01	0.00	10
	14	3.00	0.35	0.15	0.30	0.00	50
REWORK 1	*15*	4.00	0.75		0.01	0.00	20
REWORK 2	*16*	4.00	0.55		0.20	0.00	20
REWORK 3	*17*	2.00	0.95		0.25	0.00	20
							30
							55

CASE 14	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	3.00	0.55	0.45	0.30	0.00	55
	2	2.00	0.65	0.15	0.10	0.25	45
	3	3.00	0.35	0.45	0.20	0.00	50
	4	3.00	0.75	0.30	0.10	0.20	10
	5	2.00	0.55	0.45	0.15	0.00	55
	6	2.00	0.80	0.40	0.01	0.10	30
	7	2.00	0.75	0.45	0.30	0.00	30
	8	4.00	0.75	0.50	0.01	0.05	40
	9	3.00	0.90	0.35	0.20	0.00	35
	10	2.00	0.60	0.30	0.15	0.20	20
	11	2.00	0.75	0.50	0.10	0.00	10
	12	3.00	0.55	0.50	0.01	0.30	35
	13	3.00	0.45	0.35	0.01	0.00	20
	14	4.00	0.65	0.45	0.05	0.00	10
REWORK 1	*15*	4.00	0.95		0.15	0.00	50
REWORK 2	*16*	4.00	0.95		0.30	0.00	40
REWORK 3	*17*	3.00	0.45		0.30	0.00	25
							55
							20

CASE 16	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	3.00	0.80	0.25	0.10	0.30	30
	2	4.00	0.65	0.15	0.02	0.30	30
	3	3.00	0.45	0.35	0.30	0.00	20
	4	3.00	0.60	0.45	0.05	0.10	25
	5	2.00	0.90	0.15	0.01	0.00	10
	6	3.00	0.45	0.45	0.20	0.25	55
	7	3.00	0.65	0.15	0.03	0.00	55
	8	3.00	0.75	0.15	0.15	0.00	20
	9	3.00	0.75	0.50	0.20	0.00	50
	10	4.00	0.80	0.15	0.15	0.30	30
	11	2.00	0.60	0.45	0.01	0.00	55
	12	3.00	0.35	0.35	0.10	0.10	35
	13	3.00	0.75	0.30	0.30	0.00	35
	14	2.00	0.35	0.45	0.20	0.00	10
REWORK 1	*15*	4.00	0.95		0.20	0.00	30
REWORK 2	*16*	3.00	0.35		0.30	0.00	10
REWORK 3	*17*	3.00	0.90		0.01	0.00	20
							35
							35

Table B.5. Machine and buffer parameters for multiple loop validation – continued

CASE 17	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	3.00	0.75	0.35	0.25	0.05	35
	2	2.00	0.35	0.30	0.20	0.05	20
	3	3.00	0.60	0.30	0.15	0.00	10
	4	3.00	0.75	0.50	0.01	0.20	20
	5	2.00	0.65	0.45	0.20	0.00	20
	6	3.00	0.80	0.25	0.20	0.05	40
	7	3.00	0.80	0.35	0.10	0.00	40
	8	2.00	0.45	0.45	0.05	0.15	45
	9	2.00	0.45	0.50	0.20	0.00	40
	10	3.00	0.35	0.35	0.10	0.25	10
	11	4.00	0.95	0.30	0.05	0.00	10
	12	3.00	0.45	0.35	0.01	0.20	40
	13	3.00	0.90	0.15	0.10	0.00	30
	14	3.00	0.35	0.45	0.15	0.00	10
REWORK 1	*15*	3.00	0.60		0.05	0.00	45
REWORK 2	*16*	3.00	0.55		0.30	0.00	25
REWORK 3	*17*	3.00	0.65		0.01	0.00	55
							55
							30

CASE 19	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	2.00	0.90	0.25	0.01	0.30	10
	2	2.00	0.35	0.30	0.30	0.05	55
	3	4.00	0.75	0.40	0.01	0.00	10
	4	2.00	0.75	0.45	0.01	0.15	20
	5	2.00	0.35	0.50	0.05	0.00	20
	6	2.00	0.65	0.15	0.10	0.05	20
	7	2.00	0.55	0.15	0.05	0.00	30
	8	3.00	0.35	0.35	0.01	0.01	10
	9	3.00	0.75	0.25	0.01	0.00	55
	10	4.00	0.55	0.35	0.25	0.15	20
	11	4.00	0.35	0.50	0.25	0.00	20
	12	3.00	0.90	0.35	0.25	0.01	40
	13	3.00	0.35	0.40	0.30	0.00	50
	14	2.00	0.75	0.50	0.25	0.00	35
REWORK 1	*15*	2.00	0.65		0.20	0.00	10
REWORK 2	*16*	2.00	0.80		0.01	0.00	50
REWORK 3	*17*	4.00	0.60		0.30	0.00	10
							10
							50

CASE 18	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	3.00	0.80	0.02	0.01	0.20	45
	2	3.00	0.60	0.03	0.02	0.20	10
	3	3.00	0.35	0.35	0.30	0.00	30
	4	4.00	0.55	0.03	0.02	0.02	25
	5	4.00	0.80	0.40	0.25	0.00	10
	6	4.00	0.75	0.50	0.10	0.02	50
	7	3.00	0.35	0.45	0.15	0.00	20
	8	3.00	0.75	0.25	0.25	0.02	50
	9	3.00	0.35	0.35	0.05	0.00	10
	10	4.00	0.90	0.50	0.01	0.30	45
	11	4.00	0.95	0.45	0.25	0.00	40
	12	4.00	0.75	0.45	0.15	0.01	40
	13	3.00	0.90	0.30	0.25	0.00	40
	14	3.00	0.90	0.05	0.01	0.00	40
REWORK 1	*15*	2.00	0.80		0.05	0.00	45
REWORK 2	*16*	2.00	0.55		0.15	0.00	45
REWORK 3	*17*	2.00	0.35		0.01	0.00	40
							25
							35

CASE 20	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	3.00	0.35	0.35	0.25	0.05	10
	2	2.00	0.60	0.50	0.01	0.01	25
	3	4.00	0.95	0.35	0.01	0.00	45
	4	3.00	0.90	0.40	0.20	0.01	40
	5	2.00	0.55	0.40	0.30	0.00	55
	6	2.00	0.45	0.15	0.10	0.30	10
	7	2.00	0.60	0.25	0.15	0.00	25
	8	3.00	0.35	0.30	0.30	0.10	50
	9	3.00	0.35	0.35	0.05	0.00	25
	10	4.00	0.95	0.25	0.01	0.00	20
	11	3.00	0.75	0.45	0.25	0.00	50
	12	4.00	0.55	0.40	0.10	0.30	10
	13	3.00	0.95	0.40	0.20	0.00	10
	14	2.00	0.55	0.45	0.01	0.00	45
REWORK 1	*15*	3.00	0.60		0.25	0.00	50
REWORK 2	*16*	3.00	0.60		0.01	0.00	25
REWORK 3	*17*	3.00	0.35		0.10	0.00	10
							45
							10

Table B.6. Machine and buffer parameters for multiple loop validation – continued

CASE 21	M	μ	r	f	p	g	N
MAIN LINE	*1*	3.00	0.45	0.30	0.25	0.10	10
	2	2.00	0.95	0.35	0.05	0.20	55
	3	3.00	0.65	0.35	0.30	0.00	20
	4	3.00	0.45	0.35	0.20	0.15	25
	5	3.00	0.75	0.35	0.25	0.00	25
	6	2.00	0.65	0.30	0.20	0.10	45
	7	2.00	0.60	0.50	0.10	0.00	40
	8	3.00	0.35	0.30	0.25	0.20	45
	9	3.00	0.35	0.45	0.01	0.00	10
	10	2.00	0.80	0.30	0.30	0.00	30
	11	3.00	0.75	0.40	0.20	0.00	40
	12	3.00	0.65	0.35	0.05	0.20	55
	13	4.00	0.80	0.30	0.15	0.00	10
	14	2.00	0.55	0.50	0.20	0.00	35
REWORK 1	*15*	3.00	0.45		0.30	0.00	30
REWORK 2	*16*	4.00	0.35		0.25	0.00	55
REWORK 3	*17*	2.00	0.45		0.05	0.00	40
							25
							40

CASE 23	M	μ	r	f	p	g	N
MAIN LINE	*1*	3.00	0.90	0.40	0.20	0.30	35
	2	2.00	0.80	0.15	0.15	0.01	20
	3	4.00	0.90	0.35	0.20	0.00	45
	4	2.00	0.45	0.15	0.01	0.15	30
	5	3.00	0.60	0.40	0.05	0.00	40
	6	2.00	0.80	0.50	0.30	0.25	50
	7	2.00	0.90	0.50	0.15	0.00	30
	8	2.00	0.55	0.35	0.25	0.30	40
	9	4.00	0.55	0.45	0.05	0.00	45
	10	3.00	0.60	0.30	0.10	0.15	10
	11	3.00	0.65	0.50	0.10	0.00	30
	12	4.00	0.65	0.40	0.10	0.15	40
	13	2.00	0.80	0.15	0.05	0.00	50
	14	3.00	0.45	0.25	0.15	0.00	20
REWORK 1	*15*	2.00	0.35		0.30	0.00	20
REWORK 2	*16*	2.00	0.80		0.01	0.00	35
REWORK 3	*17*	2.00	0.95		0.30	0.00	45
							45
							30

CASE 22	M	μ	r	f	p	g	N
MAIN LINE	*1*	4.00	0.95	0.40	0.15	0.30	10
	2	4.00	0.95	0.40	0.15	0.05	40
	3	4.00	0.95	0.40	0.15	0.00	20
	4	4.00	0.95	0.40	0.15	0.10	40
	5	4.00	0.95	0.40	0.15	0.00	40
	6	4.00	0.95	0.40	0.15	0.01	40
	7	4.00	0.95	0.40	0.15	0.00	10
	8	4.00	0.95	0.40	0.15	0.00	55
	9	4.00	0.95	0.40	0.15	0.00	40
	10	4.00	0.95	0.40	0.15	0.15	10
	11	4.00	0.95	0.40	0.15	0.00	45
	12	4.00	0.95	0.40	0.15	0.00	10
	13	4.00	0.95	0.40	0.15	0.00	30
	14	4.00	0.95	0.40	0.15	0.00	25
REWORK 1	*15*	4.00	0.95		0.15	0.00	55
REWORK 2	*16*	4.00	0.95		0.15	0.00	55
REWORK 3	*17*	4.00	0.95		0.15	0.00	40
							35
							25

CASE 24	M	μ	r	f	p	g	N
MAIN LINE	*1*	4.00	0.65	0.25	0.20	0.15	30
	2	3.00	0.45	0.25	0.20	0.00	45
	3	2.00	0.80	0.15	0.05	0.00	55
	4	2.00	0.65	0.15	0.01	0.15	45
	5	3.00	0.95	0.50	0.20	0.00	10
	6	2.00	0.75	0.25	0.03	0.05	20
	7	2.00	0.35	0.40	0.15	0.00	20
	8	2.00	0.35	0.15	0.10	0.15	50
	9	2.00	0.95	0.35	0.01	0.00	55
	10	4.00	0.35	0.35	0.30	0.05	25
	11	2.00	0.35	0.30	0.20	0.00	45
	12	3.00	0.60	0.45	0.05	0.20	45
	13	4.00	0.35	0.35	0.20	0.00	25
	14	3.00	0.35	0.45	0.10	0.00	55
REWORK 1	*15*	2.00	0.60		0.01	0.00	40
REWORK 2	*16*	2.00	0.80		0.25	0.00	20
REWORK 3	*17*	4.00	0.90		0.15	0.00	20
							10
							10

Table B.7. Machine and buffer parameters for multiple loop validation – continued

CASE 25	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	4.00	0.90	0.35	0.25	0.30	55
	2	3.00	0.80	0.30	0.10	0.05	10
	3	2.00	0.45	0.30	0.10	0.00	25
	4	2.00	0.75	0.15	0.03	0.05	10
	5	2.00	0.60	0.15	0.15	0.00	40
	6	2.00	0.75	0.35	0.15	0.15	40
	7	4.00	0.60	0.25	0.10	0.00	10
	8	3.00	0.80	0.40	0.30	0.05	45
	9	4.00	0.55	0.35	0.10	0.00	35
	10	2.00	0.45	0.35	0.15	0.30	10
	11	2.00	0.45	0.45	0.20	0.00	35
	12	2.00	0.80	0.25	0.25	0.30	30
	13	2.00	0.90	0.30	0.10	0.00	30
	14	2.00	0.90	0.50	0.01	0.00	30
REWORK 1	*15*	3.00	0.55		0.20	0.00	35
REWORK 2	*16*	2.00	0.45		0.25	0.00	40
REWORK 3	*17*	2.00	0.55		0.25	0.00	30
							35
							10

CASE 27	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	4.00	0.35	0.45	0.20	0.30	50
	2	3.00	0.95	0.45	0.15	0.05	25
	3	2.00	0.80	0.30	0.25	0.00	10
	4	2.00	0.35	0.15	0.10	0.05	20
	5	4.00	0.60	0.45	0.30	0.00	10
	6	2.00	0.35	0.45	0.05	0.00	20
	7	4.00	0.55	0.45	0.20	0.00	40
	8	3.00	0.95	0.30	0.05	0.05	20
	9	2.00	0.55	0.15	0.10	0.00	25
	10	4.00	0.65	0.15	0.03	0.30	45
	11	2.00	0.45	0.25	0.15	0.00	30
	12	3.00	0.75	0.25	0.01	0.30	10
	13	4.00	0.90	0.15	0.02	0.00	35
	14	2.00	0.65	0.40	0.30	0.00	55
REWORK 1	*15*	4.00	0.65		0.15	0.00	50
REWORK 2	*16*	4.00	0.75		0.05	0.00	30
REWORK 3	*17*	4.00	0.45		0.15	0.00	10
							55
							30

CASE 26	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	4.00	0.75	0.45	0.20	0.25	45
	2	3.00	0.35	0.50	0.30	0.10	35
	3	3.00	0.75	0.25	0.20	0.00	35
	4	2.00	0.55	0.50	0.01	0.10	40
	5	2.00	0.45	0.45	0.05	0.00	20
	6	2.00	0.60	0.45	0.10	0.15	10
	7	2.00	0.35	0.50	0.25	0.00	20
	8	2.00	0.35	0.40	0.10	0.01	40
	9	2.00	0.35	0.35	0.20	0.00	40
	10	2.00	0.55	0.35	0.30	0.15	25
	11	2.00	0.55	0.30	0.01	0.00	45
	12	2.00	0.60	0.45	0.15	0.25	50
	13	3.00	0.65	0.40	0.05	0.00	35
	14	2.00	0.35	0.35	0.10	0.00	50
REWORK 1	*15*	2.00	0.35		0.05	0.00	45
REWORK 2	*16*	2.00	0.80		0.10	0.00	50
REWORK 3	*17*	2.00	0.35		0.30	0.00	20
							25
							35

CASE 28	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	4.00	0.60	0.35	0.30	0.15	40
	2	3.00	0.75	0.30	0.10	0.15	30
	3	4.00	0.90	0.15	0.10	0.00	10
	4	3.00	0.90	0.35	0.10	0.01	45
	5	3.00	0.65	0.40	0.20	0.00	35
	6	3.00	0.60	0.15	0.02	0.00	40
	7	3.00	0.45	0.30	0.10	0.00	25
	8	3.00	0.35	0.30	0.30	0.05	45
	9	4.00	0.60	0.45	0.01	0.00	55
	10	2.00	0.35	0.40	0.25	0.30	10
	11	2.00	0.65	0.45	0.20	0.00	40
	12	4.00	0.95	0.40	0.30	0.20	10
	13	3.00	0.75	0.50	0.20	0.00	40
	14	2.00	0.35	0.15	0.05	0.00	50
REWORK 1	*15*	3.00	0.35		0.10	0.00	55
REWORK 2	*16*	3.00	0.60		0.10	0.00	10
REWORK 3	*17*	4.00	0.35		0.01	0.00	25
							10
							35

Table B.8. Machine and buffer parameters for multiple loop validation – continued

CASE 29	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	1.00	0.65	0.30	0.10	0.30	35
	2	1.00	0.65	0.35	0.10	0.00	20
	3	1.00	0.65	0.25	0.10	0.00	35
	4	1.00	0.65	0.25	0.10	0.25	25
	5	1.00	0.65	0.50	0.10	0.00	35
	6	1.00	0.65	0.30	0.10	0.05	40
	7	1.00	0.65	0.40	0.10	0.00	45
	8	1.00	0.65	0.40	0.10	0.00	10
	9	1.00	0.65	0.40	0.10	0.00	25
	10	1.00	0.65	0.40	0.10	0.00	45
	11	1.00	0.65	0.50	0.10	0.00	40
	12	1.00	0.65	0.15	0.10	0.30	50
	13	1.00	0.65	0.40	0.10	0.00	25
	14	1.00	0.65	0.40	0.10	0.00	55
REWORK 1	*15*	1.00	0.65		0.10	0.00	30
REWORK 2	*16*	1.00	0.65		0.10	0.00	50
REWORK 3	*17*	1.00	0.65		0.10	0.00	50
							10
							50

CASE 30	*M*	*μ*	*r*	*f*	*p*	*g*	*N*
MAIN LINE	*1*	1.00	0.35	0.35	0.25	0.15	25
	2	1.00	0.55	0.15	0.03	0.30	10
	3	1.00	0.60	0.40	0.30	0.00	50
	4	1.00	0.45	0.45	0.25	0.30	20
	5	1.00	0.95	0.50	0.10	0.00	35
	6	1.00	0.95	0.40	0.15	0.00	55
	7	1.00	0.95	0.35	0.15	0.00	10
	8	1.00	0.35	0.35	0.30	0.15	50
	9	1.00	0.55	0.50	0.15	0.00	45
	10	1.00	0.65	0.15	0.15	0.20	50
	11	1.00	0.35	0.30	0.30	0.00	55
	12	1.00	0.45	0.45	0.15	0.25	10
	13	1.00	0.95	0.35	0.10	0.00	45
	14	1.00	0.60	0.35	0.01	0.00	45
REWORK 1	*15*	1.00	0.55		0.20	0.00	35
REWORK 2	*16*	1.00	0.95		0.01	0.00	20
REWORK 3	*17*	1.00	0.75		0.15	0.00	10
							30
							55

APPENDIX C: THE MATLAB CODE OF THE EXACT SOLUTION OF 2M1B

Table C.1. The Matlab code of the solution technique of 2M1B system

```
function [pro_rate,p001,p011,pN10,pN11,avg_inv,ps,pb] = conti(u1,u2,r1,r2,pr1,pr2,N)

e1=r1/(r1+pr1);
e2=r2/(r2+pr2);

a=-(u2-u1)*pr1;
b=(u2-u1)*(r1+r2)-(u2*pr1+u1*pr2);
c=u2*(r1+r2);

delta=b^2-4*a*c;


if u1==u2 | abs(u1-u2) <= 10^(-5);

    y1=(r1+r2)/(pr1+pr2);
    y2=(r1+r2)/(pr1+pr2);
    lam=(1/u1)*(r1*pr2-r2*pr1)*(1/(pr1+pr2)+1/(r1+r2));

    p(1,1,2)=u1/(r1*pr2)*(r1+r2);
    p(1,2,2)=(u1/pr2)*((r1+r2)/(pr1+pr2));
    p(N+1,2,1)=u1/(pr1*r2)*exp(lam*N)*(r1+r2);
    p(N+1,2,2)=(u1/pr1)*exp(lam*N)*((r1+r2)/(pr1+pr2));

    if lam==0
        sum = N*(y1*y2+y1+y2+1)+p(1,1,2)+p(1,2,2)+p(N+1,2,1)+p(N+1,2,2);
    end

    if lam~=0
        sum = ((exp(lam*N)/lam)*(y1*y2+y1+y2+1)-(exp(lam*0)/lam)*(y1*y2+y1+y2+1))
        +p(1,1,2)+p(1,2,2)+p(N+1,2,1)+p(N+1,2,2);
    end

    c=1/sum;

    p(1,1,2)=c*p(1,1,2);
    p(1,2,2)=c*p(1,2,2);
    p(N+1,2,1)=c*p(N+1,2,1);
    p(N+1,2,2)=c*p(N+1,2,2);

    if lam~=0
        inv=c*(((1+y1)/lam)^2)*(exp(lam*N)*(lam*N-1)+1)+N*(p(N+1,2,1)+p(N+1,2,2));
    else
        inv=(c/2)*((N*(1+y1))^2)+N*(p(N+1,2,1)+p(N+1,2,2));
    end

else

    y1(1)=(-b+sqrt(delta))/(2*a);
    y1(2)=(-b-sqrt(delta))/(2*a);
    y2(1)=(r2+r1-pr1*y1(1))/pr2;
    y2(2)=(r2+r1-pr1*y1(2))/pr2;
    lam(1)=(pr2*y2(1)-r2)*((1+y2(1))/(y2(1)*u2));
    lam(2)=(pr2*y2(2)-r2)*((1+y2(2))/(y2(2)*u2));
```

Table C.2. The Matlab code of the solution technique of 2M1B system – continued

```
if u1>u2

        if lam(1)~=0 & lam(2)~=0

            sum1=(exp(lam(1)*N)/lam(1))*(y1(1)*y2(1)+y1(1)+y2(1)+1)-(1/lam(1))
            *(y1(1)*y2(1)+y1(1)+y2(1)+1); %A1 ( f(x,x,y) )
            sum2=(exp(lam(2)*N)/lam(2))*(y1(2)*y2(2)+y1(2)+y2(2)+1)-(1/lam(2))
            *(y1(2)*y2(2)+y1(2)+y2(2)+1);%A2  ( f(x,x,y) )
            sum1=sum1+((u1-u2)/r1)*y1(1)*y2(1)+(u1/pr1)*exp(lam(1)*N)*y2(1)
            +(u1/(r2*pr1))*(r1+r2)*exp(lam(1)*N);%A1
            sum2=sum2+((u1-u2)/r1)*y1(2)*y2(2)+(u1/pr1)*exp(lam(2)*N)*y2(2)
            +(u1/(r2*pr1))*(r1+r2)*exp(lam(2)*N);%A2
            c(2)=y1(1)/(y1(1)*sum2-sum1*y1(2));
            c(1)=(-y1(2)*c(2))/y1(1);

            p(1,1,2)=c(1)*((u1-u2)/r1)*y1(1)*y2(1)+c(2)*((u1-u2)/r1)*y1(2)
            *y2(2); %p(0,0,1)
            p(N+1,2,2)=c(1)*(u1/pr1)*exp(lam(1)*N)*y2(1)+c(2)*(u1/pr1)
            *exp(lam(2)*N)*y2(2); %p(N,1,1)
            p(N+1,2,1)=c(1)*(u1*(r1+r2)/(r2*pr1))*exp(lam(1)*N)+c(2)*(u1*(r1+r2)
            /(r2*pr1))*exp(lam(2)*N);

            inv=N*(p(N+1,2,1)+p(N+1,2,2));

            for i=1:1:2
                inv=inv+c(i)*((1+y1(i))/lam(i))*((1+y2(i))/lam(i))*(exp(lam(i)*N)
                *(lam(i)*N-1)+1);
            end

        elseif lam(1)==0 & lam(2)~=0

            sum1=N*(y1(1)*y2(1)+y1(1)+y2(1)+1); %A1 ( f(x,x,y) )
            sum2=(exp(lam(2)*N)/lam(2))*(y1(2)*y2(2)+y1(2)+y2(2)+1)-(1/lam(2))
            *(y1(2)*y2(2)+y1(2)+y2(2)+1);%A2  ( f(x,x,y) )
            sum1=sum1+((u1-u2)/r1)*y1(1)*y2(1)+(u1/pr1)*y2(1)+(u1/(r2*pr1))*(r1+r2);
            %A1
            sum2=sum2+((u1-u2)/r1)*y1(2)*y2(2)+(u1/pr1)*exp(lam(2)*N)*y2(2)
            +(u1/(r2*pr1))*(r1+r2)*exp(lam(2)*N);%A2

            c(2)=y1(1)/(y1(1)*sum2-sum1*y1(2));
            c(1)=(-y1(2)*c(2))/y1(1);

            p(1,1,2)=c(1)*((u1-u2)/r1)*y1(1)*y2(1)+c(2)*((u1-u2)/r1)*y1(2)*y2(2)
            ; %p(0,0,1)
            p(N+1,2,2)=c(1)*(u1/pr1)*y2(1)+c(2)*(u1/pr1)*exp(lam(2)*N)*y2(2)
            ; %p(N,1,1)
            p(N+1,2,1)=c(1)*(u1*(r1+r2)/(r2*pr1))+c(2)*(u1*(r1+r2)/(r2*pr1))
            *exp(lam(2)*N);

            inv=N*(p(N+1,2,1)+p(N+1,2,2))+ c(2)*((1+y1(2))/lam(2))*((1+y2(2))/lam(2))
            *(exp(lam(2)*N)*(lam(2)*N-1)+1) + (c(1)/2)*N^2*(1+y1(1))*(1+y2(1));

        elseif lam(2)==0 & lam(1)~=0

            sum1=(exp(lam(1)*N)/lam(1))*(y1(1)*y2(1)+y1(1)+y2(1)+1)-(1/lam(1))
            *(y1(1)*y2(1)+y1(1)+y2(1)+1); %A1 ( f(x,x,y) )
            sum2=N*(y1(2)*y2(2)+y1(2)+y2(2)+1) ;%A2  ( f(x,x,y) )
            sum1=sum1+((u1-u2)/r1)*y1(1)*y2(1)+(u1/pr1)*exp(lam(1)*N)*y2(1)
            +(u1/(r2*pr1))*(r1+r2)*exp(lam(1)*N);%A1
            sum2=sum2+((u1-u2)/r1)*y1(2)*y2(2)+(u1/pr1)*y2(2)+(u1/(r2*pr1))*(r1+r2)
            ;%A2

            c(2)=y1(1)/(y1(1)*sum2-sum1*y1(2));
            c(1)=(-y1(2)*c(2))/y1(1);
            p(1,1,2)=c(1)*((u1-u2)/r1)*y1(1)*y2(1)+c(2)*((u1-u2)/r1)*y1(2)*y2(2);
            %p(0,0,1)
```

Table C.3. The Matlab code of the solution technique of 2M1B system – continued

```
        p(N+1,2,2)=c(1)*(u1/pr1)*exp(lam(1)*N)*y2(1)+c(2)*(u1/pr1)*y2(2);
        %p(N,1,1)
        p(N+1,2,1)=c(1)*(u1*(r1+r2)/(r2*pr1))*exp(lam(1)*N)+c(2)*(u1*(r1+r2)/
        (r2*pr1));

        inv=N*(p(N+1,2,1)+p(N+1,2,2))+c(1)*((1+y1(1))/lam(1))*((1+y2(1))/lam(1))
        *(exp(lam(1)*N)*(lam(1)*N-1)+1)+(c(2)/2)*N^2*(1+y1(2))*(1+y2(2));
    end


elseif u2>u1

    if lam(1)~=0 & lam(2)~=0

        sum1=(exp(lam(1)*N)/lam(1))*(y1(1)*y2(1)+y1(1)+y2(1)+1)-(1/lam(1))
        *(y1(1)*y2(1)+y1(1)+y2(1)+1); %A1 ( f(x,x,y) )
        sum2=(exp(lam(2)*N)/lam(2))*(y1(2)*y2(2)+y1(2)+y2(2)+1)-(1/lam(2))
        *(y1(2)*y2(2)+y1(2)+y2(2)+1);%A2  ( f(x,x,y) )
        sum1=sum1+(u2/(r1*pr2))*(r1+r2)+y1(1)*u2/pr2+exp(lam(1)*N)*y1(1)
        *y2(1)*((u2-u1)/r2) ;%A1
        sum2=sum2+(u2/(r1*pr2))*(r1+r2)+y1(2)*u2/pr2+exp(lam(2)*N)*y1(2)
        *y2(2)*((u2-u1)/r2) ;%A2

        c(2)=(exp(lam(1)*N)*y2(1))/((exp(lam(1)*N)*y2(1))*sum2-sum1*exp(lam(2)*N)
        *y2(2));
        c(1)=(1-sum2*c(2))/sum1;

        p(1,1,2)=c(1)*(u2/(r1*pr2))*(r1+r2)+c(2)*(u2/(r1*pr2))*(r1+r2);
        %p(0,0,1)
        p(1,2,2)=c(1)*(u2/pr2)*y1(1)+c(2)*(u2/pr2)*y1(2);  %p(0,1,1)
        p(N+1,2,1)=c(1)*((u2-u1)/r2)*exp(lam(1)*N)*y1(1)*y2(1)+c(2)*((u2-u1)/r2)
        *exp(lam(2)*N)*y1(2)*y2(2); %p(N,1,0)

        inv=N*(p(N+1,2,1)+p(N+1,2,2));

        for i=1:1:2
            inv=inv+c(i)*((1+y1(i))/lam(i))*((1+y2(i))/lam(i))*(exp(lam(i)*N)
            *(lam(i)*N-1)+1);
        end

    elseif lam(1)==0 & lam(2)~=0

        sum1=N*(y1(1)*y2(1)+y1(1)+y2(1)+1); %A1 ( f(x,x,y) )
        sum2=(exp(lam(2)*N)/lam(2))*(y1(2)*y2(2)+y1(2)+y2(2)+1)-(1/lam(2))
        *(y1(2)*y2(2)+y1(2)+y2(2)+1);%A2  ( f(x,x,y) )
        sum1=sum1+(u2/(r1*pr2))*(r1+r2)+y1(1)*u2/pr2+y1(1)*y2(1)*((u2-u1)/r2)
        ;%A1
        sum2=sum2+(u2/(r1*pr2))*(r1+r2)+y1(2)*u2/pr2+exp(lam(2)*N)*y1(2)*y2(2)
        *((u2-u1)/r2) ;%A2

        c(2)=y1(1)/(y1(1)*sum2-sum1*y1(2));
        c(1)=(-y1(2)*c(2))/y1(1);

        p(1,1,2)=c(1)*(u2/(r1*pr2))*(r1+r2)+c(2)*(u2/(r1*pr2))*(r1+r2);
        %p(0,0,1)
        p(1,2,2)=c(1)*(u2/pr2)*y1(1)+c(2)*(u2/pr2)*y1(2);  %p(0,1,1)
        p(N+1,2,1)=c(1)*((u2-u1)/r2)*y1(1)*y2(1)+c(2)*((u2-u1)/r2)*exp(lam(2)*N)
        *y1(2)*y2(2); %p(N,1,0)

        inv=N*(p(N+1,2,1)+p(N+1,2,2))+ c(2)*((1+y1(2))/lam(2))*((1+y2(2))/lam(2))
        *(exp(lam(2)*N)*(lam(2)*N-1)+1) + (c(1)/2)*N^2*(1+y1(1))*(1+y2(1));

    elseif lam(2)==0 & lam(1)~=0

        sum1=(exp(lam(1)*N)/lam(1))*(y1(1)*y2(1)+y1(1)+y2(1)+1)-(1/lam(1))
        *(y1(1)*y2(1)+y1(1)+y2(1)+1); %A1 ( f(x,x,y) )
        sum2=N*(y1(2)*y2(2)+y1(2)+y2(2)+1) ;%A2  ( f(x,x,y) )
```

Table C.4. The Matlab code of the solution technique of 2M1B system – continued

```
        sum1=sum1+((u1-u2)/r1)*y1(1)*y2(1)+(u1/pr1)*exp(lam(1)*N)*y2(1)
        +(u1/(r2*pr1))*(r1+r2)*exp(lam(1)*N);%A1
        sum2=sum2+((u1-u2)/r1)*y1(2)*y2(2)+(u1/pr1)*y2(2)+(u1/(r2*pr1))*(r1+r2)
        ;%A2

        c(2)=y1(1)/(y1(1)*sum2-sum1*y1(2));
        c(1)=(-y1(2)*c(2))/y1(1);

        p(1,1,2)=c(1)*((u1-u2)/r1)*y1(1)*y2(1)+c(2)*((u1-u2)/r1)*y1(2)*y2(2);
        %p(0,0,1)
        p(N+1,2,2)=c(1)*(u1/pr1)*exp(lam(1)*N)*y2(1)+c(2)*(u1/pr1)*y2(2);
        %p(N,1,1)
        p(N+1,2,1)=c(1)*(u1*(r1+r2)/(r2*pr1))*exp(lam(1)*N)+c(2)*(u1*(r1+r2)/
        (r2*pr1));

        inv=N*(p(N+1,2,1)+p(N+1,2,2))+c(1)*((1+y1(1))/lam(1))*((1+y2(1))/lam(1))
        *(exp(lam(1)*N)*(lam(1)*N-1)+1)+(c(2)/2)*N^2*(1+y1(2))*(1+y2(2));
    end


elseif u2>u1

    if lam(1)~=0 & lam(2)~=0

        sum1=(exp(lam(1)*N)/lam(1))*(y1(1)*y2(1)+y1(1)+y2(1)+1)-(1/lam(1))
        *(y1(1)*y2(1)+y1(1)+y2(1)+1); %A1 ( f(x,x,y) )
        sum2=(exp(lam(2)*N)/lam(2))*(y1(2)*y2(2)+y1(2)+y2(2)+1)-(1/lam(2))
        *(y1(2)*y2(2)+y1(2)+y2(2)+1);%A2  ( f(x,x,y) )
        sum1=sum1+(u2/(r1*pr2))*(r1+r2)+y1(1)*u2/pr2+exp(lam(1)*N)*y1(1)
        *y2(1)*((u2-u1)/r2) ;%A1
        sum2=sum2+(u2/(r1*pr2))*(r1+r2)+y1(2)*u2/pr2+exp(lam(2)*N)*y1(2)
        *y2(2)*((u2-u1)/r2) ;%A2

        c(2)=(exp(lam(1)*N)*y2(1))/((exp(lam(1)*N)*y2(1))*sum2-sum1*exp(lam(2)*N)
        *y2(2));
        c(1)=(1-sum2*c(2))/sum1;

        p(1,1,2)=c(1)*(u2/(r1*pr2))*(r1+r2)+c(2)*(u2/(r1*pr2))*(r1+r2);
        %p(0,0,1)
        p(1,2,2)=c(1)*(u2/pr2)*y1(1)+c(2)*(u2/pr2)*y1(2);  %p(0,1,1)
        p(N+1,2,1)=c(1)*((u2-u1)/r2)*exp(lam(1)*N)*y1(1)*y2(1)+c(2)*((u2-u1)/r2)
        *exp(lam(2)*N)*y1(2)*y2(2); %p(N,1,0)

        inv=N*(p(N+1,2,1)+p(N+1,2,2));

        for i=1:1:2
            inv=inv+c(i)*((1+y1(i))/lam(i))*((1+y2(i))/lam(i))*(exp(lam(i)*N)
            *(lam(i)*N-1)+1);
        end

    elseif lam(1)==0 & lam(2)~=0

        sum1=N*(y1(1)*y2(1)+y1(1)+y2(1)+1); %A1 ( f(x,x,y) )
        sum2=(exp(lam(2)*N)/lam(2))*(y1(2)*y2(2)+y1(2)+y2(2)+1)-(1/lam(2))
        *(y1(2)*y2(2)+y1(2)+y2(2)+1);%A2  ( f(x,x,y) )
        sum1=sum1+(u2/(r1*pr2))*(r1+r2)+y1(1)*u2/pr2+y1(1)*y2(1)*((u2-u1)/r2)
        ;%A1
        sum2=sum2+(u2/(r1*pr2))*(r1+r2)+y1(2)*u2/pr2+exp(lam(2)*N)*y1(2)*y2(2)
        *((u2-u1)/r2) ;%A2
        c(2)=y1(1)/(y1(1)*sum2-sum1*y1(2));
        c(1)=(-y1(2)*c(2))/y1(1);

        p(1,1,2)=c(1)*(u2/(r1*pr2))*(r1+r2)+c(2)*(u2/(r1*pr2))*(r1+r2);
        %p(0,0,1)
        p(1,2,2)=c(1)*(u2/pr2)*y1(1)+c(2)*(u2/pr2)*y1(2);  %p(0,1,1)
```

Table C.5. The Matlab code of the solution technique of 2M1B system – continued

```
            p(N+1,2,1)=c(1)*((u2-u1)/r2)*y1(1)*y2(1)+c(2)*((u2-u1)/r2)*exp(lam(2)*N)
            *y1(2)*y2(2); %p(N,1,0)

            inv=N*(p(N+1,2,1)+p(N+1,2,2))+ c(2)*((1+y1(2))/lam(2))*((1+y2(2))/lam(2))
            *(exp(lam(2)*N)*(lam(2)*N-1)+1) + (c(1)/2)*N^2*(1+y1(1))*(1+y2(1));



        elseif lam(2)==0 & lam(1)~=0

            sum1=(exp(lam(1)*N)/lam(1))*(y1(1)*y2(1)+y1(1)+y2(1)+1)-(1/lam(1))*(y1(1)
            *y2(1)+y1(1)+y2(1)+1); %A1 ( f(x,x,y) )
            sum2=N*(y1(2)*y2(2)+y1(2)+y2(2)+1) ;%A2  ( f(x,x,y) )
            sum1=sum1+(u2/(r1*pr2))*(r1+r2)+y1(1)*u2/pr2+exp(lam(1)*N)*y1(1)*y2(1)
            *((u2-u1)/r2) ;%A1
            sum2=sum2+(u2/(r1*pr2))*(r1+r2)+y1(2)*u2/pr2+y1(2)*y2(2)*((u2-u1)/r2)
            ;%A2

            c(2)=y1(1)/(y1(1)*sum2-sum1*y1(2));
            c(1)=(-y1(2)*c(2))/y1(1);

            p(1,1,2)=c(1)*(u2/(r1*pr2))*(r1+r2)+c(2)*(u2/(r1*pr2))*(r1+r2)
            ; %p(0,0,1)
            p(1,2,2)=c(1)*(u2/pr2)*y1(1)+c(2)*(u2/pr2)*y1(2);  %p(0,1,1)
            p(N+1,2,1)=c(1)*((u2-u1)/r2)*exp(lam(1)*N)*y1(1)*y2(1)+c(2)*((u2-u1)/r2)
            *y1(2)*y2(2); %p(N,1,0)

            inv=N*(p(N+1,2,1)+p(N+1,2,2))+c(1)*((1+y1(1))/lam(1))*((1+y2(1))/lam(1))
            *(exp(lam(1)*N)*(lam(1)*N-1)+1)+(c(2)/2)*N^2*(1+y1(2))*(1+y2(2));

        end
    end
end


ps=p(1,1,2)+(1-u1/u2)*p(1,2,2);
pb=p(N+1,2,1)+(1-u2/u1)*p(N+1,2,2);

pro_rate1=u1*e1*(1-pb);
pro_rate2=u2*e2*(1-ps);

pro_rate=pro_rate1;
p001=p(1,1,2);
p011=p(1,2,2);
pN10=p(N+1,2,1);
pN11=p(N+1,2,2);
avg_inv=inv;
```

REFERENCES

Burman, M., 1995, *New Results In Fow Line Analysis*, Ph.D. Dissertation, Massachusetts Institute of Technology.

Buzacott, J.A. and J.G. Shanthikumar, 1993, *Stochastic Models of Manufacturing Systems*, Prentice Hall.

David, R., X. L.Xie and Y. Dallery, 1988, *Properties Of Continuous Models Of Transfer Lines With Unreliable Machines And Infinite Buffers*, Tech. Rep. LAG 88-50, April.

Gershwin, S. B., and O. Berman, 1981, "Analysis of Transfer Lines Consisting of Two Unreliable Machines with Random Processing Times and a Finite Storage Buffer", *AIIE Transactions*, Vol. 13, No. 1, pp. 2-11, March.

Gershwin, S. B., 1987, "An Efficient Decomposition Method For The Approximate Evaluation Of Tandem Queues With Finite Storage Space And Blocking", *Operations Research*, Vol. 35, No. 2, pp. 291–305, March-April.

Gershwin, S. B., 1987, "Representation And Analysis Of Transfer Lines With Machines That Have Different Processing Rates", *Annals of Operations Research*, Vol. 9, pp. 511 – 530.

Gershwin, S. B., 2002, *Manufacturing Systems Engineering*, Massachusetts Institute of Technology.

Helber, S., 1997, *Approximate Analysis of Unreliable Transfer Lines with Rework or Scrapping of Parts*, M.S. Thesis, Institut für Produktionswirtschaft und Controlling Fakultat für Betriebswirtschaftslehre Ludwig-Maximilians-Universitat München.

Lim, J.T., S. M. Meerkov and Top F., 1990, “Homogeneous, asymptotically reliable serial production lines: Theory and a case study”, *IEEE Trans. Automat. Contr.*, Vol. 35, pp. 524–534, May.

Kim, J., 2005, *Integrated Quality And Quantity Modeling Of A Production Line*, Ph. D. Dissertation, Massachusetts Institute of Technology.

Le Bihan, H. and Y. Dallery, 1998, “A Robust Decomposition Method For The Analysis Of Production Lines With Unreliable Machines And Finite Buffers”, *Technical Report LIP6*, *Annals of Operations Research*.

Li, J., 2004, “Performance Analysis Of Production System With Rework Loops”, *Report R&D-225, General Motors Research & Development Center*, Warren, MI.

Poffe, A. and S.B. Gershwin, 2005, “Integrated Quality and Quantity Modelling in a Production Line”, *Operations Research Center Working Paper*, Massachusetts Institute Of Technology.