AN ABSTRACT OF THE THESIS OF

Dan R. Staley for the degree of Master of Science in Industrial Engineering presented on September 8, 2006.
Title: General Design Rules for the Allocation of Buffers in Closed Serial Production Lines.

Abstract approved:

Redacted for Privacy

David S. Kim

One of the classic problems in industrial engineering is the buffer allocation problem. The objective of the buffer allocation problem is to maximize some line output, typically throughput, through the allocation of buffers throughout the production line. Previous work in this area has focused on either determining general design rules or developing heuristics to determine optimal buffer placement. Most of this work was done on production lines which were assumed to have an unlimited supply of jobs to the first workstation and an unlimited storage space after the last workstation (open production lines).

The purpose of this research was to study buffer allocation in closed production lines and focus on the validation and development of general design rules for buffer placement. Balanced and unbalanced lines were studied with workstations

representing manual stations, and then with workstations representing automated machines.

The general approach taken was to first identify an existing buffer allocation rule established for open lines and then determine the corresponding rule for closed production lines. Next, a set of experiments were designed to test the closed production line design rule. Finally, the design rule was validated, modified, or reformulated.

The findings of this research indicate that an even buffer allocation is optimal for balanced closed production lines. It also showed similar behavior to open production lines when a bottleneck is present, but the effect of the bottleneck is not as strong. Also, differences between reliable and unreliable lines were observed. Finally, the rules developed seemed to be consistent for short lines as well as long ones.

Several of the rules developed in this research can be utilized immediately in the design of closed production lines. Also, because of the lack of literature on closed production lines, it will serve as a good first step into the understanding of the role of buffers in closed production lines.

General Design Rules for the Allocation of Buffers in Closed Serial Production Lines

by

Dan R. Staley

A THESIS

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Master of Science

Presented September 8, 2006
Commencement June 2007

Master of Science thesis of Dan R. Staley presented on September 8, 2006.

APPROVED:

Redacted for Privacy

Major Professor, representing Industrial Engineering

Redacted for Privacy

Head of the Department of Industrial and Manufacturing Engineering

Redacted for Privacy

Dean of the Graduate School

I understand that my thesis will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my thesis to any reader upon request.

Redacted for Privacy

Dan R. Staley, Author

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF FIGURES

LIST OF FIGURES (Continued)

LIST OF FIGURES (Continued)

LIST OF TABLES

LIST OF TABLES (Continued)

LIST OF TABLES (Continued)

LIST OF APPENDICES

LIST OF APPENDIX TABLES

LIST OF APPENDIX TABLES (Continued)

LIST OF APPENDIX TABLES (Continued)

LIST OF APPENDIX TABLES (Continued)

LIST OF APPENDIX TABLES (Continued)

GENERAL DESIGN RULES FOR THE ALLOCATION OF BUFFERS IN CLOSED SERIAL PRODUCTION LINES

INTRODUCTION

This research addresses buffer allocation in closed serial production lines. A buffer is space to store work-in-progress, and buffer allocation is the specific placement of a limited number of buffers in a production line. The buffer allocation problem is a classic industrial engineering problem that has been well studied since the placement of buffers in a production line can have a significant impact on the performance (in this case throughput) of the line.

The focus of this research is the development of general design rules for the placement of buffers in closed serial production lines. A design rule, in general, holds true based on certain properties of the line. For example, the "bowl phenomenon" is a general design rule. This rule states that whenever you have a balanced (stations with the same mean processing time and processing time variation) open serial production line the buffers should be placed as evenly as possible (to maximize throughput), with the excess buffers placed symmetrically around the center of the line. By simply knowing the line is balanced, open, and serial one can design the line for optimal or near optimal throughput by following the "bowl phenomenon" design rule.

Classification of Production Lines

Since this research addresses buffer allocation in *serial production lines,* it is necessary to define what a serial production line is and some of the important differences between its many subdivisions. First, a production line is a manufacturing system consisting of two or more workstations through which parts/jobs pass in a specified order and receive processing, never passing through the same station more than once. Secondly, a serial production line is one where jobs move in a linear fashion from one workstation to the next. There are no assembly/merge steps in which a part must wait for one or more other parts before being able to receive processing. It is assumed that there are no parallel processing steps in which a job may enter one of two or more equivalent parallel positions to receive the same processing. Serial production lines can be further divided based upon the various operating characteristics of the line such as the blocking scheme, timing of job movements, production control mechanism, workstation type, and requirement for job carriers. See Figure 1 for a serial production line configuration tree. The branches in darker shades depict the line types addressed in this research.

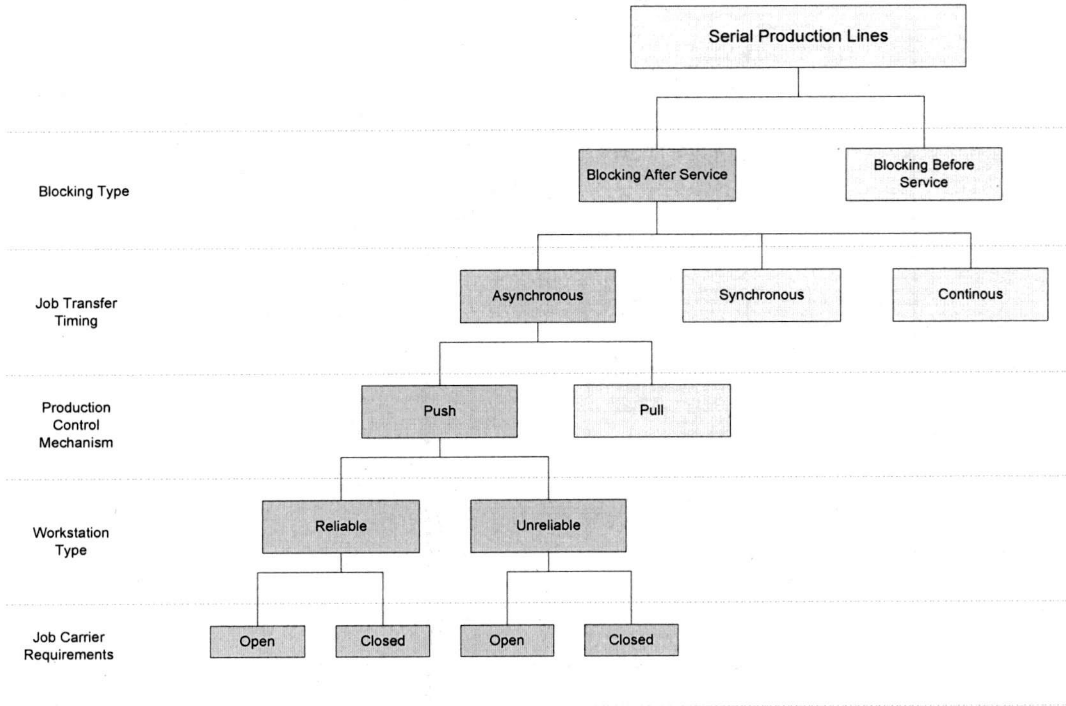

Figure 1: Serial Production Line Configuration Tree Diagram

Production lines will generally use one of two main types of blocking schemes, blocking before service (BBS) or blocking after service (BAS). The majority of production lines use BAS. In BAS, upon completion of service at a workstation, a job attempts to enter the downstream buffer. If the buffer is full then the workstation is said to be blocked and cannot start processing another job. The workstation remains blocked and the job is forced to wait until a departure occurs in the downstream buffer. This differs from BBS where prior to starting the processing of a job, a workstation checks to see if there is space available in the downstream buffer. If the buffer is full then the workstation is considered blocked and will not process the job. The workstation remains blocked until a departure occurs in the downstream buffer at which point the workstation begins processing the job. For this research, all lines are assumed to operate using the BAS scheme.

Production lines can be classified as synchronous, asynchronous, or continuous depending on the timing of part/job transfers. In synchronous lines (often referred to as transfer lines), the transfer of all parts in the line takes place simultaneously. Automated production lines are often transfer lines. In asynchronous lines, each station operates and transfers jobs at its own pace. In continuous lines, parts move at a constant continuous speed such as parts on a motorized conveyor. All production lines studied in this research are asynchronous with respect to part/job transfers and movement.

Production lines can also be separated based on their production control mechanisms. A production control mechanism is the method/logic by which jobs are released from one station to the next. There are several production control mechanisms that may be used, but the two most general are referred to as "push" and "pull" systems. Most production lines may be one or the other or, more commonly, some hybrid of both. Push systems are typically demand driven. When a system operates in push mode, a job enters the system and moves from one station to the next whenever there is room for it to do so. In pull mode, a job enters the system or moves from one station to the next based upon system status.

For example, if when a job completes processing at the final station it initiates a signal for the first station to begin processing a new job, the line is operating in pull mode. For this research, all lines are assumed to operate in push mode.

The type of workstations in the production line is also important to consider. Workstations may, in general, be classified as reliable or unreliable. Reliable lines are characterized by variable processing times, but are not subject to failures (a period of time when a job is in a workstation and no processing or material movement is taking place). A production line with manual workstations is considered a reliable line. In contrast an unreliable line may have constant or variable processing times but are also subject to failures of random duration. A production line with automated workstations (e.g., robotic welding workstations) is considered an unreliable line. Both reliable and unreliable lines will be addressed in this research.

The requirement of job carriers is one final distinction to address when looking at serial production lines. In an "open" production line jobs are moved through the line and processed in workstations without the need to be attached to a "job carrier". A sheet metal stamping line is an example of an open production line. In a closed production line, jobs must be attached or held by a job carrier as it moves through the production line and receives processing. An example of a closed system is an engine block machining line where each job is attached to a fixture to facilitate movement and to help align jobs for processing. In a closed line there is a constant number of carriers that is less than the total space available (number of stations plus the number of buffers) circulating within the system. It will be assumed that there is an unlimited supply of jobs entering the production line, and that there is unlimited space to store completed jobs once they leave the system after the last station. As a result of the assumptions, the first station in an open production line is never starved and the last station is never blocked. In a closed production line, since the number of carriers in the system is constant, once a job completes processing at the last station, the job is removed from its carrier and the carrier returns to the first station. The first station becomes starved if no carrier has been returned from the last station and the station is up and ready to process a new job. The last station becomes blocked if a job completes processing and there is no room before the first station for its carrier. This research focuses on closed production lines, but also looks at open production lines for comparison purposes.

General Approach

The objective of this research is the development of general design rules for buffer allocation in closed serial production lines. The general approach used in this research is as follows:

1. Examine established buffer allocation design rules for open production lines in the research literature.
2. Identify the corresponding buffer allocation design rule for closed production lines.
3. Design a set of experiments to examine the validity of the design rule for closed production lines.
4. Validate, modify, or generate a new design rule based on the experimental results.

Throughout this research the effectiveness of a specific buffer allocation will be evaluated based on line throughput. To determine the throughput of the production lines, computer simulation is used.

Contribution

There has been a considerable amount of research into the behavior of open serial production lines and the buffer allocation problem. However, there has been very little research addressing buffer allocation in closed serial production lines. This is a significant omission considering there are many closed production lines in use. For example, production lines that make use of AGV's (automated guidance vehicles) to transport jobs such as in the automotive industry are closed production lines. This research is a first step toward better understanding of buffer allocation in closed serial production lines.

Outline of Thesis

The first section of this thesis is the literature review, describing past research on buffer allocation in serial production lines. The second section describes the methodology. It includes descriptions of the model and the experiments performed. Next is the data analysis and results. This includes a description of the tests performed and the results obtained. Conclusions of the research follow the data analysis. Finally, the potential for future research is discussed.

LITERATURE REVIEW

Serial production lines can be quite varied so it is important to clearly identify the type of production line being studied in this research, and where this research lies relative to the overall body of production line research. In this research, the serial production lines examined all have asynchronous part transfers, finite buffers, and a "push" production control mechanism. The papers reviewed in detail will also only deal with production lines of this type. However, there exists a vast amount of research on other production line types as well. There are many papers dealing with synchronous part transfers. A few that address buffer effects and buffer allocations include Buzacott (1967), Sheskin (1976), and Yamashina and Okamura (1983). There are also many papers that address other production control mechanisms. Some of these papers include Atwater and Chakavorty (1996) and Kirkavak and Dincer (1999) who studied pull production lines, Blackstone Jr. (2004) who looked at a CONWIP (constant WIP) system, and Kadipasaoglu et al. (2000) who looked at DBR (Drum-Rope-Buffer) lines. These papers are only relevant in that they show areas for potential extensions to the research presented in this document.

The bulk of the past research on buffer allocation for asynchronous push serial production lines falls into one of two categories. The first category is the development of heuristics and algorithms for buffer allocation optimization. The second category is the development of general design rules and guidelines for buffer allocation. The research on heuristics and algorithms is extensive, but since the focus of this research is on general design rules, the first part of the literature review will only briefly cover heuristics and algorithms. The remainder of the literature review will go into detail reviewing papers that focus on general design rules for buffer allocation in asynchronous serial production lines.

Algorithms and Heuristics

Due to the complexity of solving buffer allocation optimization problems for longer lines (for example a 10 workstation line with total buffer capacity of 20 has over 10 million possible allocations) much effort has been put into developing algorithms that will reduce the time needed to find an optimal (ex. maximizing production line throughput) or near optimal allocation. Harris and Powell (1999) developed a simplex search algorithm which works by starting with an initial set of buffer allocations (the simplex) and sorting them from best to worst. Then using a combination of the best and worst solutions a new allocation is determined and if it is better than the worst solution, it replaces it in the simplex and the process is repeated. Vouros and Papadopoulos (1998) came up with a knowledge based system in which an algorithm allocates buffer slots to each station then simulation is used to determine line characteristics (e.g. throughput, average waiting time, etc.) or "knowledge". Based on the knowledge the algorithm determines how to rearrange the buffer slots. Kim and Lee (2001) used a non-standard exchange vector algorithm (iterative) with the goal of minimizing WIP. This is done by defining a "neighborhood" around an initial solution and then finding the best solution in the neighborhood. This solution then serves as the starting point for the next neighborhood and a new local optimum is found. This repeats until no better solution can be found. Spinellis et al. (2000) used simulated annealing and a generalized queuing network algorithm to find optimal server, work, and buffer allocations. These papers are just a small portion of the papers in the area of buffer allocation algorithms. For a more thorough review see Vergara (2005).

Design Rules/Guidelines

This section of the literature review begins with a review of the work allocation problem since it is directly related to the buffer allocation problem. The work allocation problem deals with unbalancing the line in order to reduce the negative affects of variability and improve line performance. Following this, the bulk of this section addresses literature directly addressing design rules/guidelines for buffer allocation. Buffer allocation research for both balanced and unbalanced production lines is reviewed, as is buffer allocation research for closed production lines. A balanced line is one in which the mean and the variability of operating times are equal for all workstations.

Work Allocation Problem

Early work by Hillier and Boling (1966) investigated unbalancing production lines by unbalancing the mean operating times of the workstations. They found that by deliberately unbalancing the line in such a way that stations in the middle of the line had lower mean operating times than stations at the ends of the line that the throughput would be increased. They termed this the "bowl phenomenon".

Since then, several other papers have addressed the work allocation problem. Hillier and Boling (1979) looked at the effect of the number of stations in the line, the maximum buffer size between stations, as well as the station operating time variability. They found that as the number of stations in the line increases, the difference in throughput from balanced to optimal also increases (although at a decreasing rate). However, the average amount of unbalance ($[\sum_{j=1}^{N} | w_j - 1 |] / N$, where N is the number of stations, w_j is the mean processing time at workstation j, and 1 is the mean processing time for all stations in the balanced line) in optimal allocation remains approximately the same for all line lengths. They also found that if operating times are highly variable (coefficient of variation greater than 1), increasing the buffer size decreases the average amount of unbalance in the optimal allocation but only slightly decreases the resulting throughput improvement (from balanced to optimal). Finally, they showed that with small buffer capacities, decreasing the operating variability decreases the amount of unbalance and the throughput improvement, but at a very slow rate. However, if the buffer capacity is increased while simultaneously decreasing the variability then the decrease in unbalance and throughput improvement both occur very rapidly.

Hillier and So (1993) investigated the bowl phenomenon by extending the work of Hillier and Boling (1979) to longer lines and lines with larger variability. Their findings also showed that the improvement in throughput associated with the bowl phenomenon increases with line length. However, although Hillier and Boling found that increasing the operating variability results in the increase of both the unbalance and the percentage improvement in throughput, Hillier and So showed that this does not hold true for all levels of variability. As the variability continues to increase, there comes a point where the degree of unbalance and throughput improvement levels off and eventually decreases.

Pike and Martin (1994) investigated the operation time distribution and variance. They found that there was little effect on the optimal configuration. They also found that the amount of imbalance in the line can generally be twice the amount of imbalance in the optimal bowl and still perform better than the balanced line.

Buffer Allocation Problem

The work allocation problem and in particular the bowl phenomenon show that unbalancing production lines can result in an increase in throughput. However, assigning significantly different workloads to different stations can be difficult, so researchers began investigating other ways to unbalance the line. This led to the buffer allocation problem.

Balanced Production Lines

El-Rayah (1979) investigated imbalance of buffer allocation and processing time variability effects on throughput and average number of units in system. He found that for relatively small amounts of buffers (total buffers $\leq 4(N-1)$, where N = the number of workstations in the line) that the throughput is maximized and the amount of WIP is minimized when the buffer allocation is uniformly distributed. However, if imbalance is unavoidable then the extra buffers should be located in the center of the line. El-Rayah also found that imbalances in variability act like imbalances in mean operating times in that the bowl phenomenon (the stations with lower variability should be placed in the center of the line and the stations with larger variability at the ends) results in the optimal throughput.

A landmark paper for general design rules of serial production lines was done by Conway et al. (1988). Among the numerous experiments performed, they looked at balanced lines with no buffers, lines with balanced allocation of buffers, and also lines with unbalanced allocation of buffers. They found that most of the line capacity lost due to interference between stations occurs in the first few stations (75% of capacity is lost in the first five stations), meaning that longer lines are not much worse than shorter ones. The presence of buffers will help overcome this capacity loss, and that these buffers should be proportional to the coefficient of variation (CV) of the workstation operating times. Conway et al. claim that buffer sizes of ten times the CV should be sufficient to recover 80-85% of the capacity lost. In contrast with El-Rayah, Conway et al. found that the optimal buffer allocation pattern should be symmetrical with slightly greater capacity in the center ("inverse bowl phenomenon"). This conclusion was a result of some other observations they made which together have been termed "the decomposition principle". Conway et al. showed that the best location to place a single buffer is where an infinitely large buffer would be most effective and that throughput is a decreasing function of line length. Therefore, the throughput of a line that is split into multiple sections with buffers will be dominated by the throughput of the largest section. Conway et al. also show that when a symmetrical pattern of buffer allocation is not possible, the same throughput is achieved with mirror allocations ("reversibility principle"). This reversibility principle can be seen as a corollary to the reversibility property shown by Muth (1979) in which the workstation processing times were investigated instead of buffers.

Studying reliable lines, Hillier et al. (1993) found that no single pattern of buffer allocations is optimal for all situations. If there are no additional buffer storage spaces available beyond a uniform allocation to each of the storage areas, then for a small number of buffers between each station, a uniform allocation of buffers is optimal. However, for larger number of buffers, the buffers shift away from the ends of the line toward the center. If there is one addition to a buffer storage space, it should be placed in the center position (or the downstream one when there are two center buffers). If there are multiple extra buffer spaces, they should be spread out across all the interior buffers. If the number of buffers is also a decision variable and they are assumed to have a certain cost, then the optimal pattern will commonly be in the form of $(n, n + 1, n + 1, \ldots, n + 1, n)$ where n is the greatest integer $\leq Q/(N-1)$ and Q is the total number of buffers and N is the total number of stations in the line.

Papadopoulos and Vidalis (2001) looked at shorter reliable lines and the effect of buffer allocations on the average work-in-process (WIP). They found that WIP is an increasing function of ordered buffer allocations. The throughput of these allocations forms the shape of an inverse bowl, which agrees with Conway et al.'s findings. However, Papadopoulos and Vidalis further show that within the bowl smaller inverse bowls are formed which are further divided into smaller bowls and so on ("self-similarity phenomenon").

Hillier and So (1991a) investigated the effect of the coefficient of variation on buffer allocation in unreliable serial production lines. They found further evidence of the inverse bowl phenomenon and that the bowl phenomenon is more pronounced with higher values of variability. Hillier and So (1991b) found that for lines with stations having CV values below 1.5 that percentage increases in throughput are the same for incremental increases in buffer capacity regardless of the probability of failure and the mean downtime.

Hillier (2000) found that the inverse bowl pattern becomes more pronounced as the number of buffers increases. He refined the pattern (n, n + 1, n + 1, …, n +1, n) described by Hillier et al. (1993) to be (n, n +0.03Nn, …, n +0.03Nn, n). Also, conflicting with Hillier and So (1991a) he found that the CV had very little impact on the pattern of buffer space allocation. However, he did find that the optimal number of buffer spaces was proportional to the CV.

Enginarlar et al. (2002) studied unreliable production lines with exponential, erlang, and rayleigh distributions for both uptime and downtime. They found that longer lines require a larger level of buffering between stations to reach a minimum throughput and that this increase is exponentially decreasing as a function of the total number of stations. They also found that the buffering necessary for lines with ten stations is sufficient to accommodate downtime in lines with greater than ten stations. Additionally, they showed that larger machine efficiencies requires less buffering while smaller variability of uptime and downtime distributions leads to smaller level of buffering. Finally, they develop some simple equations based on machine efficiencies that can be used to determine the amount of buffers to allocate in order to achieve a minimum throughput.

Unbalanced Production Lines

One of the earlier works on unbalanced production lines was done by Freeman (1964). Several design rules were determined for automated production lines. The first is to avoid extreme buffer allocations. The second is that the greater the difference between a bad station (bottleneck) and a good station the more buffer capacity should be allocated to it. Thirdly, more capacity should be assigned between a bad and a mediocre station (or bad and bad) than a bad and good station. Finally, the end of the line is more critical than the front; therefore, if a bottleneck station is at the end of the line, even more buffer capacity should be allocated to it.

Conway et al. (1988) investigated unbalanced reliable production lines. They claim that buffers are more essential in balanced lines than in unbalanced ones because stations adjacent to the bottleneck (largest mean processing time) in a sense act as buffers. In unbalanced lines, the bottleneck station pulls the buffer capacity toward itself while the fastest station pushes it away.

Powell (1994) looked at unbalanced three-station production lines with regards to means, standard deviations, or both. He found that the station with a higher standard deviation or mean draws the first buffer toward it then the remaining buffers are placed alternatively between the buffer slots unless imbalance is extreme (termed the "Alternation Rule"). Powell also found that if there is both an imbalance in mean and standard deviation that the imbalance in mean has a greater effect on the optimal buffer placement.

Powell and Pyke (1996) studied the effects of bottlenecks on slightly longer production lines. Using lognormal distribution for the processing times and lines with no breakdowns, several conclusions were reached. First, relatively large imbalances (10-30 percent) in means between a single bottleneck and rest of the line are required to shift the optimal allocation from a balanced distribution. As the bottleneck increases the optimal buffer pattern gradually shifts toward the bottleneck, with the best buffer to move being the one furthest away. Second, for small number of buffers, the imbalance required to cause a shift in the optimal pattern is less the closer the bottleneck is to the center of the line. Third, as the number of stations in the line increase, the imbalance required to move all the buffers next to the bottleneck increases while the imbalance required to shift successive buffers decreases. Finally, preliminary results suggest that the CV of the line has little impact on when buffer allocation shifts occur, and that buffer allocation is less sensitive to changes in variability than to the mean (of a bottleneck station).

Closed Production Lines

Virtually all the research involving general design rules have dealt with open production lines; however, one paper that dealt with closed production lines was done by Dallery and Towsley (1991). They claimed that the throughput of a line with population N (can be thought of as the number of carriers) is the same as that with population C-N, where C is the total buffer capacity. They also claim that the reverse network (with the same population) of a line will have an equivalent throughput. However, these results are for systems run using blocking before service which is not typical of most production lines and not what this thesis is concerned about.

Summary

The literature dealing with production lines is quite large and varied. The research in this thesis is specifically addressing general properties for the distribution of buffers in closed production systems and their effect on line productivity (the buffer allocation problem). The buffer allocation problem is a well studied one, and the papers in this area can generally be divided into those that develop algorithms to solve for an optimal solution and those that develop general design rules. This research falls into the latter category, and so the literature reviewed is focused on papers that address the development of general design rules.

The type of line being studied is another way papers related to the buffer allocation problem can be differentiated from one another. To look at literature dealing with all types of lines would be overwhelming. For that reason, certain line types were mostly ignored. With regards to blocking, the vast majority of production lines use blocking after service, and the production lines investigated in this research are no different. Therefore, little attention was paid to production lines with other types of blocking in the literature review. Papers dealing with asynchronous part transfers and synchronous part transfers are both common, as is literature on both push and pull lines. However, since this research is limited to asynchronous serial production lines that are run in push mode, all the papers detailed in this review are also limited to this type. Research dealing with synchronous part transfer and control mechanisms other than "push" mode is only briefly mentioned.

Another production line aspect that often differentiates papers is workstation type. Both reliable and unreliable workstations are prevalent in the real world and in the literature. Although much of the research in this paper deals with reliable workstations, unreliable workstations are addressed as well because of the possibility that the two types don't behave similarly. The literature review covers both types as well.

Over the years, there have been two main approaches to the buffer allocation problem; studying balanced lines, and studying unbalanced lines. Balanced lines have been studied because they are simpler and easier to analyze than unbalanced ones. Also, many real life production lines are designed to be balanced. However, realistically there will always be some imbalance in a line so research has examined unbalanced lines to see what kind of effect it has. The focus of this research is on unbalanced lines, specifically the effect of bottlenecks, but balanced lines are also addressed. Therefore prior research on both types of lines was studied.

Virtually all research on the buffer allocation problem has been done for open production lines. That is the main difference of this research from the research described in this section. The previous research developed rules for open lines; this research examines rules for closed lines. Due to the lack of literature on closed production lines, the prior research on open lines is used as a starting point for experiments and rules to test.

METHODOLOGY

This section describes the details of tools and experiments that were used to explore buffer allocation design rules for closed serial production lines. It consists of two parts. The first part describes the simulation model that was used to calculate the line output (throughput) for the various experiments performed. Specific operating assumptions modeled are explained. The second part describes the individual experiments in more detail. Buffer allocations rules/ guidelines existing for open serial production lines were the starting point for the experiments in this paper. The first step was to duplicate previous experiments used to validate a buffer allocation design rule for an open line, using a closed line. Based on the results, more questions were raised and new experiments performed. This process was repeated as new questions/ideas were formulated.

The Simulation Model

Model Description and Assumptions

The simulation program used to run the experiments was a custom program written using the C programming language. The reason for using a custom program was that commercial general purpose simulation software is too slow to handle the considerable number of tests conducted with the precision desired.

Specific operating assumptions modeled are that workstation processing times are independent, transit times between buffers are zero, blocking follows the block-after-service scheme, and that each job/part requires a carrier for movement and processing in the line. The buffer transit time is assumed to be zero because relative to the processing time it is quite small. Blocking-after-service is assumed because that is how most production lines are run. Requiring a carrier for every job makes the line a closed production line. The output generated from the model was line throughput. For unreliable production lines, processing times were assumed constant, and failures were assumed to be operation-dependent. That is, failures could only occur during the processing of a job. The time between failures and repair times were both assumed to be exponentially distributed. While looking at data from actual automated production lines, Inman (1999) showed that these assumptions are reasonable. The downtime of a station was lumped into the overall processing time of a given job.

An important aspect of the model is the number of carriers in the production line. Closed production lines may have as few as one carrier and as many as the number of workstations plus the total buffer capacity. The number of carriers has an effect on the throughput so a standard rule was followed throughout this research for consistency. According to Kim et al (2002) the upper bound on the number of carriers in a three-station production line is equivalent to the number of stations in the line plus half the total buffer capacity. Any value above this number will have a negative impact on the throughput. It is conjectured that this rule holds for closed production lines with more than three workstations. To be consistent as well as to maximize throughput this rule was adopted for all closed production lines tested.

Running the Simulation (Simulation Parameters)

Before running the simulation, several factors had to be considered. For each production line setup (number of stations, buffer allocation, processing time distributions), the number of jobs to simulate (simulation length), the number of individual runs per replication, and the number of replications had to be decided upon. The main tradeoffs for each of these parameters were precision and run time. By increasing any of these parameters the precision of the model increases, but at the cost of taking more time. Five throughput values (replications) were calculated. Five replications were chosen to get as many points to analyze as possible without taking too much computer time. Each throughput value was determined by taking an average of 30 individual runs of 50,000 jobs. By taking the average of 30 runs there was more confidence in the data analysis because the averages come from a normal population (central limit theorem). The run time was set at 50,000 jobs, a value that was large enough that the system had time to settle into a steady state and any startup time became negligible.

Also, for each unique buffer allocation pattern, the same sequence of random numbers was used for determining workstation processing times. This was done to cut down on variability between buffer allocations and to allow pair-wise comparisons. The intent is to reduce as much variability from the throughput differences (between different buffer allocations) as possible. The use of random numbers was completely synchronized for all open lines and for closed production lines with a least one buffer after the last workstation. Therefore, for two different buffer allocations, the difference in throughput would be solely due to the buffer allocation scheme and not the variability of the processing times. However, it should be noted that complete synchronization of random numbers was not obtained for closed production lines with no buffer after the last workstation. This will result in slightly higher variability when comparing two buffer allocations.

Validation of Simulation Model

A critical component to having an accurate model is the uniform random variables used in generating the workstation processing times. To verify that these random variables were indeed random and possessed the necessary characteristics of uniformity, independence, and correlation, the random number generator was put through a series of tests and passed them all. See Appendix A for the detailed results. These tests were suggested by Law and Kelton (1991).

To verify the simulation model's accuracy, production line setups from other papers were run and compared to the results in those papers. Results for both open and closed serial production lines as well as lines with lognormal and exponential processing times and unreliable workstations were compared. The largest difference between throughputs of this paper and published results was 0.3% with the majority less than 0.05%. See Appendix B for complete results.

The Experiments

There were several general questions about buffer allocation in closed serial production lines examined in this research. The five basic questions investigated were:

1. In a balanced closed serial production line, is the optimal buffer allocation evenly distributed across the line?

2. What is the effect of the presence of a bottleneck workstation? Is buffer allocation less sensitive to bottlenecks in closed production lines (vs. open)?
3. What is the difference in throughput between the optimal buffer allocation and an even allocation for higher bottleneck levels?
4. Do automated lines behave the same as reliable?
5. Does the behavior seen in shorter lines hold for longer lines?

A series of experiments were designed to help shed some light onto these questions. This section of the paper will go into more detail on how these experiments were run.

Experimental Line Setups

Although the types of production lines being studied in this research operate under specific assumptions, there are still several factors involved that may cause different behaviors. Several different line setups were used in each experiment to show the effect (or lack of effect) of various line variables. These variables and the respective levels used are summarized in Table 1. Every possible combination of these variables (for the exponential distribution, only a CV equal to 1 could be used) was used in each of the first three experiments.

Table 1: Line Setup Summary

Variable	Levels
Distribution Type	Lognormal, Exponential
CV	0.5, 1, 2
Line Length	4, 8

Both exponential and lognormal processing time distributions were used in the various tests performed. The exponential distribution was chosen because it is easy to work with and has often been used in previous research. The lognormal distribution was used because the mean and CV can be easily changed (independently of each other) and according to Powell and Pyke (1996) (citing Knot and Sury (1987) and Buzacott and Shantikumar (1993)) it has a positively skewed distribution which is more consistent with real world production lines.

The lognormal processing times used had coefficient of variations (CV) of 0.5, 1, and 2. A CV of 1 was chosen to match the exponential so that a comparison between distributions could be made, a CV of .5 was examined because it is in the center of the range of realistic values (according to Powell and Pyke (1996) citing Knott and Sury (1987)), and a CV of 2 was included as a worst case scenario (worse than typically found in industry).

As line length increases the number of possible buffer allocations also increases, but at an exponentially greater rate. Therefore, only line lengths of four and eight stations are examined for most of the experiments. Longer lines are addressed in experiment 5.

Experiment 1

As shown previously in Conway et al.(1988), in open production lines the best location to place a single buffer is where an infinitely large buffer would be most effective. An infinite buffer effectively creates two independent lines and Conway also showed that throughput is a decreasing function of line length. Therefore, the throughput of a line that is split into multiple sections with buffers will be dominated by the throughput of the largest section (decomposition principle). In a balanced closed production line, this means that the optimal solution for a given number of buffers should be one in which those buffers are as evenly placed as possible.

The purpose of the first experiment was to show that the optimal buffer allocation in a balanced closed production line is in fact one in which the buffers are spread as evenly across the line as possible. This was accomplished by finding the throughput for the different possible buffer allocations (for each setup) and determining which allocation resulted in the highest throughput (optimal solution).

Total line buffer capacities of 1 through 4 were tested using each possible setup determined from Table 1. To save computation time only the exponential distribution was used to test buffer capacities greater than 4. Also, in the eight-station lines, not all possible buffer allocations were considered. Each buffer had a maximum capacity set at 2. This may not include the optimal allocation, but if a buffer required a capacity greater than 2 to optimize the line, then it would stand to reason that having a capacity of 2 would be a better solution than a capacity of 1 (no buffer capacity would exceed 1 in an evenly distributed case) . Therefore, a maximum capacity of 2 should be enough to determine if an even distribution is not optimal.

Experiment 2

The second experiment is a continuation of the work done in Powell and Pyke (1996) who examined bottlenecks in open production lines and how the magnitude of the bottleneck affects optimal buffer allocation. This research took the same basic experimental design of Powell and Pike and performed it on both open and closed production lines.

Again, all line setups possible from Table 1 were tested for both open and closed production lines. For each case the starting condition was such that the workstations were balanced so the optimal buffer allocation for the closed production lines was an evenly distributed one. It was decided to have one buffer between each station; therefore, the total buffer capacity depended on the line length. Closed production lines always have one more buffer than their open equivalents. For example, a closed production line with four stations would have a total buffer capacity of four, while an open production line with four stations would only have a total buffer capacity of three (one buffer space between station).

Starting with a balanced system, a single bottleneck was created by increasing only the mean processing time or increasing both the mean processing time and the processing time variation of one station. This bottleneck was made progressively worse until a shift of the optimal buffer allocation (highest throughput) occurred. Starting with workstations with processing times equal to 1 (arbitrary time unit), the increase in mean processing times was done in increments of 0.01 time units. This increment was chosen because it was small enough to offer a fairly high precision level without being so small that computer time needed to find the optimal buffer allocation was exceptionally long. The bottleneck for all closed lines was chosen to be the first station. The first station was arbitrarily chosen because the behavior of the line should be the same wherever the bottleneck occurs. However, in an open line, the behavior of the line may differ depending on where the bottleneck occurs. Therefore in open lines, multiple runs of each setup were performed with a different station chosen as the bottleneck for each run.

With experimental runs involving the exponential distribution only one parameter could be changed, which effectively increased both the mean processing time as well as the processing time variability. With the lognormal distribution, the mean and variability can be increased separately or at once. Two different approaches to increasing the bottleneck in the lines with lognormal processing times were employed. The first approach was to increase just the mean and the second was to increase both the mean and the variability. Increasing only the mean is simpler and more likely to happen in a real life situation because mean processing times are easier to control than variability. However, the exponential distribution has a constant CV and increasing the mean only in a lognormal distribution reduces the CV. Therefore, the second approach was to increase both the mean and variability such that the CV remained constant so that a more direct comparison between the two distributions could be made.

Experiment 3

Experiment 2 examined bottleneck severity such that a statistically significant difference in throughput between allocations (optimal and even) was detected.

However, a statistical difference does not necessarily equate to a practical difference. Due to the long run lengths and common random number threads which are to reduce variability and shrink confidence interval sizes, a statistically significant difference can be detected with an absolute difference of less than 0.05% in throughput. The purpose of experiment 3 is to see what happens to the absolute difference in throughput between the optimal and even buffer allocations as the bottleneck continues to increase in severity past the point of a new optimal buffer allocation.

The same setups (distribution, CV, buffer capacity, line length) used in experiment 2 were also used in this experiment. Line throughputs were determined at several set bottleneck levels. The bottleneck workstation's mean processing time was incremented by 0.25 time units from 1.5 to 3 and from 1.5 to 2.75 for lines with four stations and eight stations respectively. The non-bottleneck workstation's mean processing time was 1 time unit. The percent below optimal of the balanced buffer allocation (*1-(even allocation throughput/optimal allocation throughput*)) was recorded. A heuristic algorithm program developed by Vergara (2005) was used in order to get a near optimal solution for eight-station lines because the time necessary for an exhaustive search of all possible solutions was too great. This algorithm is a genetic algorithm that uses the number of times each workstation (on a critical path) is starved to determine where to place the buffers. Also to save time, the bottleneck in the open eight-station lines was only considered to be one of the first 4 stations because results from the four-station line and experiment 2, show that the behavior is symmetric around the center of the line (i.e. the first and last station behave the same, as does the second and second to last, etc.). This appears to be a result of the reversibility property of open lines addressed by Muth (1979) and Conway et al. (1988) among others. Graphs of this data were plotted to see if any trends could be determined (e.g. maximum throughput % difference, bottleneck level when the difference reaches some critical point, etc.)

Experiment 4

The first three experiments all examined reliable production lines; however, unreliable lines are also of interest. Experiment 4 examined these lines. The first thing investigated with unreliable lines was to see if an evenly distributed buffer allocation pattern is optimal for balanced closed production lines. Like all the other experiments, production lines of length 4 and 8 were used. However, unlike reliable lines, the processing times are constant and the variability comes from the failure rates. Therefore, a variety of different MTTR (Mean Time to Repair) and MTBF (Mean Time between Failures) were used in the testing instead of different distributions and CVs. The values of MTTR and MTBF were chosen such that they fell within the range of a real world situation representing an automobile body shop assembly system (MTTR = 1, 5, and 10; MTBF = 50, 100, 250, and 500).

The real world data was supplied by an automotive manufacturer through personal communication (see Appendix C).

The next part of this experiment examined the difference between open production lines and closed production lines with respect to how sensitive the optimal buffer allocation pattern is to the severity of a bottleneck. An initial balanced line consisting of stations with the various MTTR and MTBF values had a bottleneck created by increasing the MTTR of a single station. The MTTR was increased such that the t_e (effective mean processing time) increased by increments of 0.05. Effective mean processing time is a term defined in Hopp and Spearman (2001) and is equivalent to the mean processing time after accounting for the time a station is unavailable due to machine breakdowns. The t_e is found by dividing the workstation's mean processing time by its availability. The MTTR was increased until there was a shift in the buffer allocation pattern or until the t_e of the bottleneck station was equal to two time units (approximately twice the t_e of the non-bottleneck stations). Once a change was detected, the bottleneck was looked at closer by evaluating the buffer allocations with MTTR values in increments of one just prior to the change. This was done to get the precision of the MTTR level creating a shift down to one time unit and be consistent for all setups. Only lines of length 4 were used in this comparison. Also, because of the reversibility property the bottleneck l was only considered to be located in one of the first two stations. Eight-station lines were not tested due to time constraints.

Unreliable lines were then compared to reliable lines. Using the same MTTR and MTBF values that fell within a range of a real production line, the unreliable line was compared to its equivalent reliable line (in terms of t_e and CV). The reliable lines had lognormal distributions. Again due to time constraints, only four-station closed lines were tested. The optimal, second best, and worst solutions were examined to see if the allocations were the same. The throughput values were recorded as well.

The next experiment comparing unreliable lines and equivalent reliable lines examined the magnitude of throughput differences with various buffer allocations and bottleneck severity levels. The optimal, 2nd best, and worst buffer allocations were determined at multiple bottleneck levels (te = 1.05, 1.1, 1.25, 1.5, 1.75, 2). The throughput for these allocations was determined and compared to the throughput for an evenly distributed buffer allocation pattern. Four-station lines were investigated where the CV = 0.5 and the CV = 1 to see if the lines behaved the same for different CVs and line lengths. The CV was kept constant to compare unreliable lines more directly with reliable lines. Increasing the MTTR only in an unreliable line to increase the t_e would result in an increase of the bottleneck station's CV, while increasing the mean only in a reliable station to increase t_e would decrease the CV.

Experiment 5

The prior experiments dealt with lines that consisted of four or eight stations. However, in industry, lines can be much longer and these longer lines need to be investigated as well. They were not included in the previous experiments because of the massive computer time needed to run through all the possible buffer allocations. Only a small number of tests were performed on longer lines for this reason.

To investigate longer lines, a line length of 20 stations was used. The first thing done was to check the hypothesis that an evenly distributed buffer allocation is best for a balanced closed serial production line. Stations with an exponential processing time distributions and the lognormal distributions with CV = 0.5 and 1, as well as unreliable stations with CV = 1 were included. The reliable lines had workstations with t_e's = 1, whereas the unreliable line used a t_e's = 1.01. A t_e of 1.01 was used in the unreliable case instead of t_e = 1 because a t_e of 1 is only possible if there are no failures (when the processing time equals 1 time unit). To save more time, the lognormal distribution with a CV = 2 was omitted because a CV value that high is less likely to be encountered in industry. Also, due to computer time constraints, only a fraction of the possible buffer allocations were examined. Included allocations were an evenly distributed allocation (all buffers of capacity 1) and allocations in which one buffer had a capacity of 2, one a capacity of 0, and the rest with a capacity of 1. This can be thought of as a shift of one buffer. Since buffers attempt to reduce the impact of variability to improve throughput, if one section of a line is more variable than the rest, then shifting a single buffer toward it should improve the overall throughput of the line. If shifting a single buffer does not improve the throughput of the line, it would stand to reason that the even distribution is optimal. Also, to further verify that an evenly distributed distribution is best, the optimal allocation was determined with the heuristic algorithm developed by Vergara(2005) and checked for agreement with the above results.

Next the hypothesis that closed production lines are less sensitive to bottlenecks than open lines was tested. Using the same technique as in experiment 2, the t_e at one station was increased until the optimal buffer allocation pattern changes. The CV was kept constant for all setups for this experiment. Exponential and lognormal (CV = 0.5, 1) processing time distributions were used. The lognormal distributions were tested only for the closed production lines to see the effect of distribution and CV on bottleneck sensitivity in a closed production line. For open lines, only the exponential distribution was used. From prior experiments the behavior of exponential and lognormal distributions was the same, so the lognormal distribution was not used to save time. For open production lines, the bottleneck was located at the 1st, 5th, and 10th station. The behavior had been symmetric in the other experiments so the bottleneck was not included in the second half of the line. These three bottleneck locations were chosen to represent the beginning of the line, the center of the line, and one station somewhere in between.

ANALYSIS AND RESULTS

The analysis and results are divided into six subsections with the first five addressing each experiment and the last a summary of the results. The organization of each sub-section consists of data analysis followed by a discussion of the results. Some experiments consisted of multiple sub-experiments. In these cases each sub-experiment is addressed separately describing the analysis and results of one sub-experiment before describing the next.

Experiment 1

For experiment 1, the throughput of production lines with multiple buffer allocations were compared to each other to find the optimal (i.e., highest average throughput as indicated by simulation) allocation(s). Of interest was whether or not the mean throughputs for each allocation differed from each other, and if so, does the even allocation result in the greatest mean throughput. Since several buffer allocations were to be compared against each other, the appropriate statistical tests applied were multiple sample comparison tests. From the several methods available for use, the LSD (least significant difference), Scheffe, and Bonferroni methods were all used. These are standard statistical tests that are described in many statistical texts such as the one by Ramsey and Shafer (2002). The LSD method generates the narrowest confidence intervals, the Scheffe method generates the largest confidence intervals, and the Bonferroni method generates an intermediate confidence interval size. A 95% confidence level was used for all comparisons. Using the statistical software package *Statgraphics*, the allocations that had statistically significant differences in throughput means were determined and graphed (See Appendix G for the raw data). Unless there was a difference in conclusions among the different statistical tests applied, only the results of the LSD method are presented.

Four-station line setups were investigated first. Starting with the placement of a single buffer and testing the null hypothesis that $\mu_1 = \mu_2 = \mu_3 = \mu_4$ with the alternative hypothesis being one $\mu \neq$ at least one other μ, there was found to be no significant difference in throughput between any of the allocations. This holds for all processing time distributions. This was expected because no matter where the buffer is placed it will break up the line exactly the same since all stations are identical.

Table 2 shows the groupings of the buffer allocations using the multiple comparison methods (each column of X's represent one grouping of allocations in which the mean throughputs are statistically indistinguishable from each other). Note that the notation $(b_1, b_2, \ldots)$ for the buffer allocation represents the buffer capacity after workstation 1, 2, etc. Also, the allocations are listed from top to bottom in ascending order of mean throughput.

Table 2: LSD 95% Multiple Comparison of Throughput for Exponential Four-Station Line with Single Buffer

```
Method: 95.0 percent LSD
allocation     Count     Mean              Homogeneous Groups
--------------------------------------------------------------------------------
(1,0,0,0)      5         0.513284          X
(0,0,0,1)      5         0.513288          X
(0,1,0,0)      5         0.513308          X
(0,0,1,0)      5         0.513328          X
```

Looking at the exponential distribution, the largest difference in means was only 0.0000442 while the half-widths of the 95% confidence intervals (CI) for throughput difference were 0.000405708, 0.000596584, and 0.000575736 for the LSD, Scheffe, and Bonferroni methods respectively. Table 3 summarizes the largest difference in means and the half-widths of the 95% CIs for throughput difference for the four different distributions as well as the F-test p-values. All p-values were well above 0.05; therefore, the null hypothesis that $\mu_1 = \mu_2 = \mu_3 = \mu_4$ cannot be rejected.

Table 3: Summary of Multiple Comparison Half-Widths for 95% CI's for Throughput - Four-Station Lines with Single Buffer

Distribution	Largest difference	LSD Half-Interval	Scheffe Half-Interval	Bonferroni Half-Interval	P-value
Exponential	0.000044	0.00040571	0.00059658	0.00057574	0.9952
Lognormal (1,.25)	0.000039	0.00023515	0.00034578	0.00033369	0.9842
Lognormal (1,1)	0.000045	0.00030518	0.00044876	0.00043308	0.9918
Lognormal (1,4)	0.000043	0.00025605	0.00037651	0.00036335	0.9861

When a second buffer was introduced into the system, the number of possible allocations increased, but the results for the optimal allocation stayed consistent with the hypothesis that evenly distributed buffers in a balanced line is optimal. Using the same multiple comparison methods as before, the allocations were separated into groups within which no statistical difference in mean throughput existed. These groupings are presented in Table 4.

Table 4: LSD 95% Multiple Comparison of Throughput for Exponential Four-Station Line with Two Buffers

Method: 95.0 percent LSD

allocation	Count	Mean	Homogeneous Groups
(2,0,0,0)	5	0.513284	X
(0,0,0,2)	5	0.513288	X
(0,2,0,0)	5	0.513308	X
(0,0,2,0)	5	0.513328	X
(1,1,0,0)	5	0.53434	X
(1,0,0,1)	5	0.534343	X
(0,0,1,1)	5	0.534359	X
(0,1,1,0)	5	0.534378	X
(1,0,1,0)	5	0.541813	X
(0,1,0,1)	5	0.541814	X

A couple of things stood out when looking at Table 4. First, the optimal allocations were the two in which the buffers were as evenly distributed throughout the line as possible ((1,0,1,0) and (0,1,0,1)). Second, all the results stayed consistent with the decomposition principle. There were clearly three groupings among the allocations which were also shown in Figure 2. Figure 2 presents the means and the LSD intervals for the average throughput of each buffer allocation.

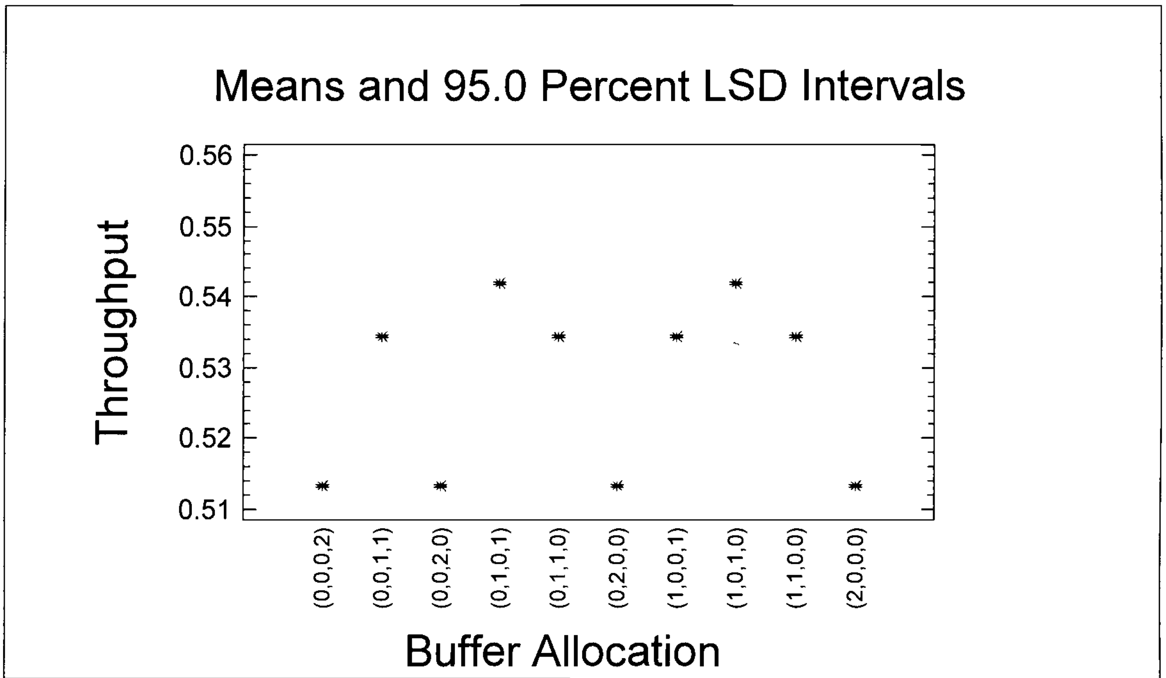

Figure 2: Means Plot for Exponential Four-Station Line with Two Buffers

The best were the two in which the buffers separated the lines into two smaller lines consisting of two stations each, the next best were the lines which were broken up into two lines consisting of one and three stations, and the worst were the lines that were not separated at all and still one line of four stations. See Figures 3-5 for graphical representations of these lines. Finally, the allocations (2,0,0,0), (0,2,0,0), (0,0,2,0), and (0,0,0,2) had the exact same throughput as

(1,0,0,0), (0,1,0,0), (0,0,1,0), and (0,0,0,1) respectively. Placing a second buffer in the same spot as the first had no effect on the throughput at all.

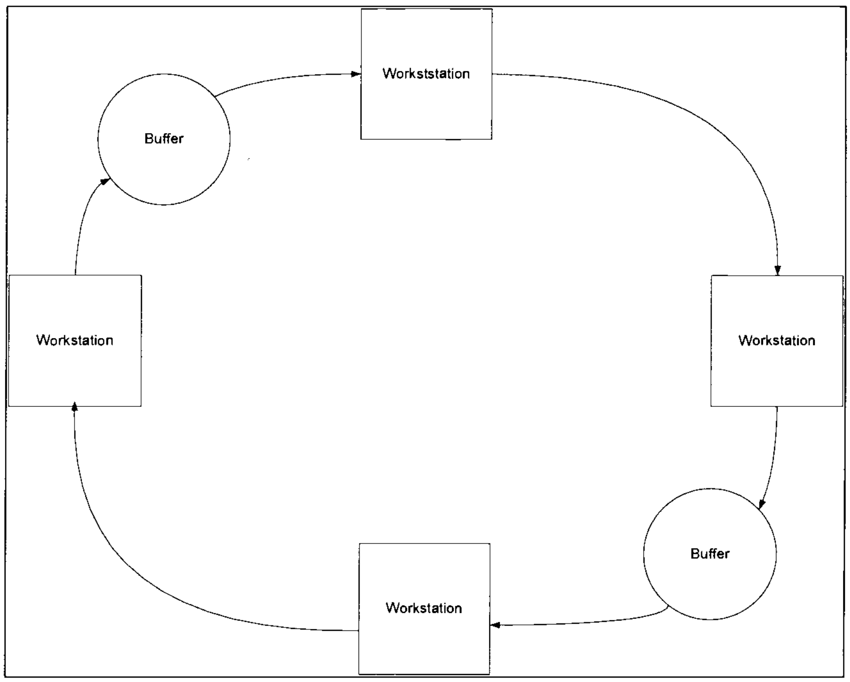

Figure 3: Buffer Configuration Resulting in Two Sub-lines of Length 2

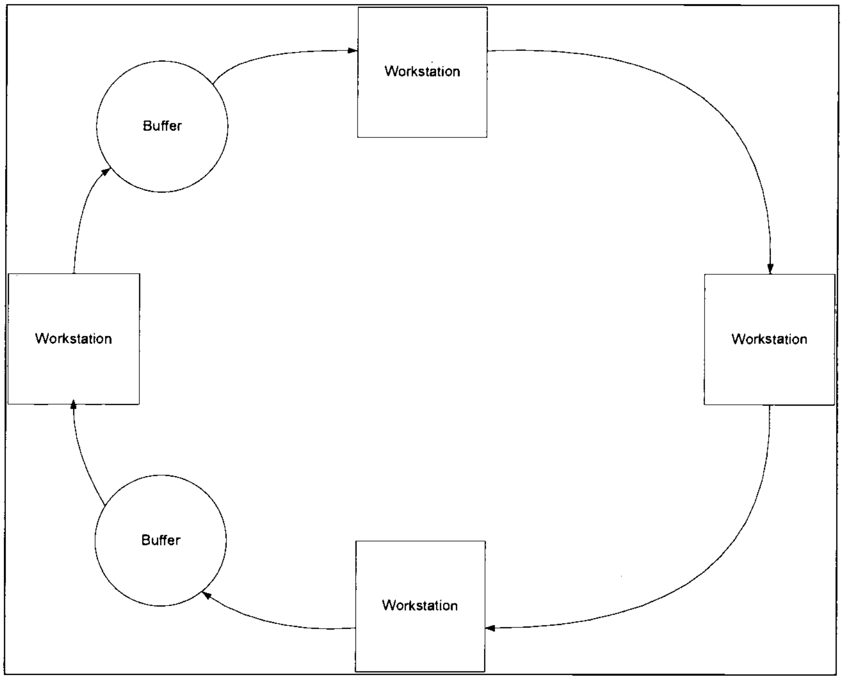

Figure 4: Buffer Configuration Resulting in Sub-lines of Length 1 and 3

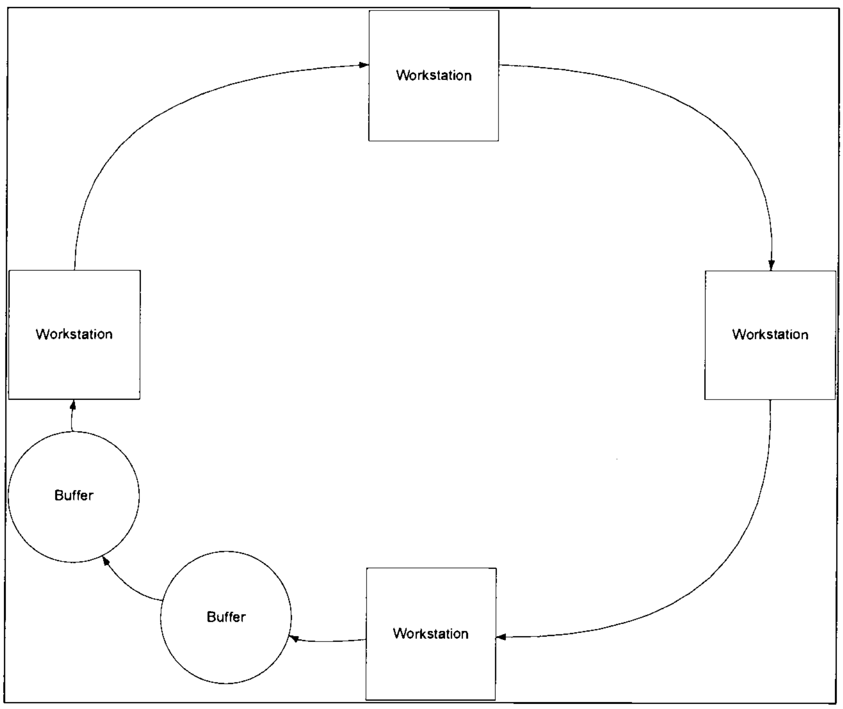

Figure 5: Buffer Configuration Resulting in Sub-line of Length 4

As more buffers were added to the system, the patterns seen in the case of two buffers continued to hold. Tables 5 and 6 show the results for three and four buffers in four-station lines respectively. Table 7 presents the half-widths of 95% CIs for line throughput difference for the setups represented by Tables 4-6. The best solutions (resulting in the highest throughput) continued to be the allocations that separated the line into the smallest length subsections which were the allocations in which the buffers were as evenly distributed across the line as possible. Also, the worst solutions (lowest throughput) were the allocations that did not break up the line at all. Finally, intermediate allocations resulted in increased throughput as the length of the largest subsection (stations not separated by buffers) decreased. Lines with five to eight buffers were also examined with no new findings from what was already observed. These results can be obtained from the data in Appendix G.

Table 5: LSD 95 % Multiple Comparison of Throughput for Exponential Four-Station Line with Three Buffers

Method: 95.0 percent LSD

allocation	Count	Mean	Homogeneous Groups
(3,0,0,0)	5	0.514914	X
(0,0,0,3)	5	0.514918	X
(0,3,0,0)	5	0.51493	X
(0,0,3,0)	5	0.514955	X
(0,0,2,1)	5	0.544078	X
(2,1,0,0)	5	0.544084	X
(2,0,0,1)	5	0.544085	X
(1,0,0,2)	5	0.544089	X
(1,2,0,0)	5	0.5441	X
(0,2,1,0)	5	0.544117	X
(0,0,1,2)	5	0.544118	X
(0,1,2,0)	5	0.544181	X
(0,2,0,1)	5	0.558165	X
(2,0,1,0)	5	0.558181	X
(0,1,0,2)	5	0.558185	X
(1,0,2,0)	5	0.558228	X
(0,1,1,1)	5	0.571825	X
(1,1,0,1)	5	0.571838	X
(1,0,1,1)	5	0.571845	X
(1,1,1,0)	5	0.571849	X

Table 6: LSD 95% Multiple Comparison of Throughput for Exponential Four-Station Line with Four Buffers

```
Method: 95.0 percent LSD
allocation     Count     Mean              Homogeneous Groups
--------------------------------------------------------------------
(4,0,0,0)      5         0.514914          X
(0,0,0,4)      5         0.514918          X
(0,4,0,0)      5         0.51493           X
(0,0,4,0)      5         0.514955          X
(0,0,3,1)      5         0.544078           X
(3,1,0,0)      5         0.544084           X
(3,0,0,1)      5         0.544085           X
(1,0,0,3)      5         0.544089           X
(1,3,0,0)      5         0.5441             X
(0,3,1,0)      5         0.544117           X
(0,0,1,3)      5         0.544118           X
(0,1,3,0)      5         0.544181           X
(2,2,0,0)      5         0.552021            X
(2,0,0,2)      5         0.552025            X
(0,0,2,2)      5         0.552033            X
(0,2,2,0)      5         0.55207             X
(0,3,0,1)      5         0.558165             X
(3,0,1,0)      5         0.558181             X
(0,1,0,3)      5         0.558185             X
(1,0,3,0)      5         0.558228             X
(2,0,2,0)      5         0.573348              X
(0,2,0,2)      5         0.573349              X
(0,1,2,1)      5         0.579512               X
(2,1,0,1)      5         0.57952                X
(1,2,1,0)      5         0.579561               X
(1,0,1,2)      5         0.579566               X
(0,2,1,1)      5         0.583629                X
(2,0,1,1)      5         0.583653                X
(1,0,2,1)      5         0.583666                X
(0,1,1,2)      5         0.583676                X
(2,1,1,0)      5         0.583681                X
(1,1,0,2)      5         0.583683                X
(1,2,0,1)      5         0.583685                X
(1,1,2,0)      5         0.583718                X
(1,1,1,1)      5         0.607215                 X
```

Table 7: Multiple Comparison Half-Widths for 95% CI's for Throughput – Exponential Four-Station Lines

Number of Buffers	LSD Half-Interval	Scheffe Half-Interval	Bonferroni Half-Interval
2	0.000405708	0.000596584	0.000575736
3	0.000360817	0.001035880	0.000692440
4	0.000356838	0.001294690	0.000731220

Eight -stations lines were also examined in experiment 1. The same tests performed for the four-station lines were performed for the eight -station lines and the results (see Appendix G for raw data) also demonstrated applicability of the decomposition principle and that the even buffer allocation pattern was optimal. Table 8 presents the multiple comparisons for a single buffer and shows that there was no difference in throughput means between the different allocations. Also, Table 9 presents the partial results of the multiple comparisons with four buffers. It shows the lowest throughput grouping of allocations and the highest throughput grouping. Again, the optimal solution was an evenly distributed buffer allocation scheme and the worst scheme was when all buffers were located at a single station.

Table 8: LSD 95% Multiple Comparison of Throughput for Exponential Eight - Station Line with Single Buffer

Method: 95.0 percent LSD

allocation	Count	Mean	Homogeneous Groups
(00100000)	5	0.426503	X
(00010000)	5	0.426507	X
(00001000)	5	0.426509	X
(01000000)	5	0.426518	X
(00000100)	5	0.42652	X
(00000010)	5	0.426527	X
(10000000)	5	0.426542	X
(00000001)	5	0.426543	X

Table 9: Partial LSD 95% Multiple Comparison of Throughput for Exponential Eight-Station Line with Four Buffers

Method: 95.0 percent LSD

allocation	Count	Mean	Homogeneous Groups	
(04000000)	5	0.313326	X	
(00000400)	5	0.313329	X	
(00004000)	5	0.313333	X	
(00000040)	5	0.313338	X	
(00040000)	5	0.313347	X	
(00000004)	5	0.313349	X	
(00400000)	5	0.313354	X	
(10101010)	5	0.331384		X
(01010101)	5	0.331393		X

Experiment 2

The purpose of the second experiment was to determine if a difference in the sensitivity of the optimal buffer allocation (to the severity of a single bottleneck) exists between open and closed production lines. The bottleneck severity level that created a change in the optimal buffer allocation was determined for a variety of different setups. A single workstation had its mean processing time increased from a mean equal to one, until a buffer allocation change showed a better throughput. This change was determined to have occurred once there was a statistically significant difference between the buffer allocation that had the highest simulated throughput (investigated all cases where there was a shift of a single buffer from one spot to another) and the buffer allocation pattern optimal for the balanced line. The optimal allocation for all balanced lines, with the total number of buffers set to have one buffer between each station, was one buffer between each station (except the eight-station open production lines with lognormal(1, 4)[1] distributed processing times). Paired t-tests were used to determine when there was a significant (at a 95% confidence level) difference between the two allocations. A paired t-test was used because the simulation of production lines (to determine throughput) used synchronized random number streams. The null hypothesis of the paired t-test was that the mean throughput difference between lines with different buffer allocations was zero and the alternative hypothesis was that the difference was greater than zero. The reason for using a one-sided test was because the question of interest was whether the new allocation was better than the even allocation, not just whether the two were different. The final results for the four-station and eight-station lines are shown in Table 10 and Table 11 respectively. They show the bottleneck severity level (the mean processing time at the bottleneck) required to cause a change in the optimal (balanced) buffer allocation in closed production lines and open production lines (for each position of the bottleneck workstation). The starting mean processing times for all stations was one.

---

[1] if X~lognormal(a,b), then E[X] = a, Var[X] = b

Table 10: Required Mean Processing Time at the Bottleneck Station to Cause a Change in the Optimal Buffer Allocation for Four-Station Lines

	Four Stations					
	Processing Time Distribution	Closed	Open: Position of Bottleneck			
			1	2	3	4
	exponential	1.76	1.45	1.44	1.44	1.45
Const. Variance	lognormal(1,.25)	1.49	1.35	1.39	1.39	1.35
	lognormal(1,1)	1.69	1.50	1.44	1.44	1.49
	lognormal(1,4)	1.71	1.50	1.30	1.29	1.50
Const. CV	lognormal(1,.25)	1.51	1.33	1.38	1.38	1.33
	lognormal(1,1)	1.74	1.46	1.44	1.43	1.46
	lognormal(1,4)	1.88	1.51	1.34	1.33	1.51

Table 11: Required Mean Processing Time at the Bottleneck Station to Cause a Change in the Optimal Buffer Allocation for Eight-Station Lines

		Eight Stations								
	Processing Time Distribution	Closed	Open: Position of Bottleneck							
			1	2	3	4	5	6	7	8
	exponential	1.54	1.35	1.26	1.23	1.22	1.21	1.22	1.26	1.35
Const. Variance	lognormal(1,.25)	1.38	1.30	1.27	1.27	1.28	1.29	1.27	1.27	1.30
	lognormal(1,1)	1.64	1.48	1.35	1.28	1.25	1.25	1.28	1.35	1.48
	lognormal(1,4)	1.77	1.09	1.09	1.27	1.14	1.14	1.27	1.09	1.08
Const. CV	lognormal(1,.25)	1.35	1.26	1.24	1.24	1.25	1.25	1.24	1.24	1.26
	lognormal(1,1)	1.57	1.36	1.25	1.20	1.18	1.18	1.20	1.25	1.36
	lognormal(1,4)	1.75	1.08	1.08	1.24	1.12	1.12	1.24	1.08	1.08

Several conclusions were reached by looking at this data. First, the bottleneck severity necessary for a different buffer allocation (other than the balanced allocation) to be the optimal allocation was greater for the closed production line than the open production line with the same number of stations, regardless of the open system bottleneck location. This is also shown graphically in Figure 6 and Figure 7.

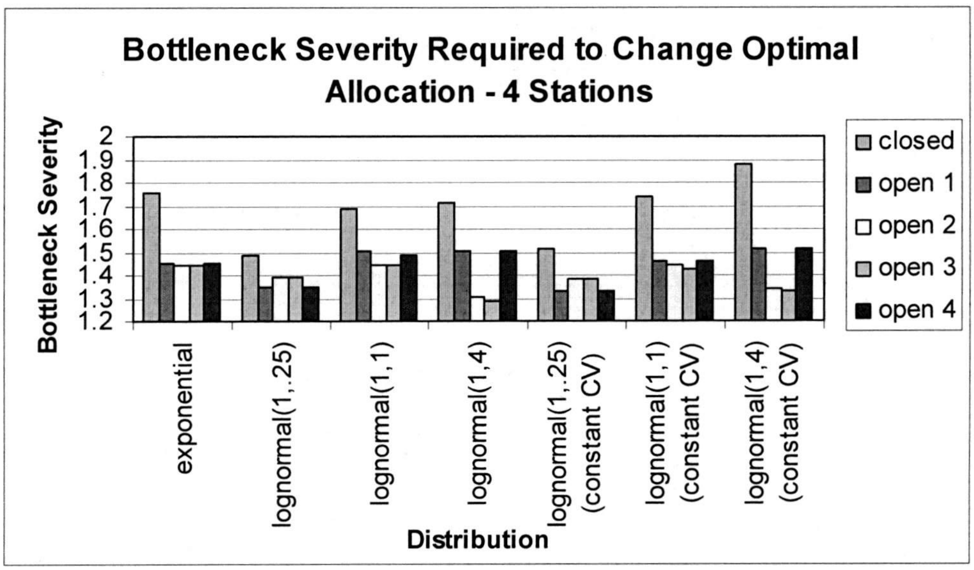

Figure 6: Bottleneck Severity Level Required to Change to a Non-Even Buffer Allocation as Optimal – Four Stations

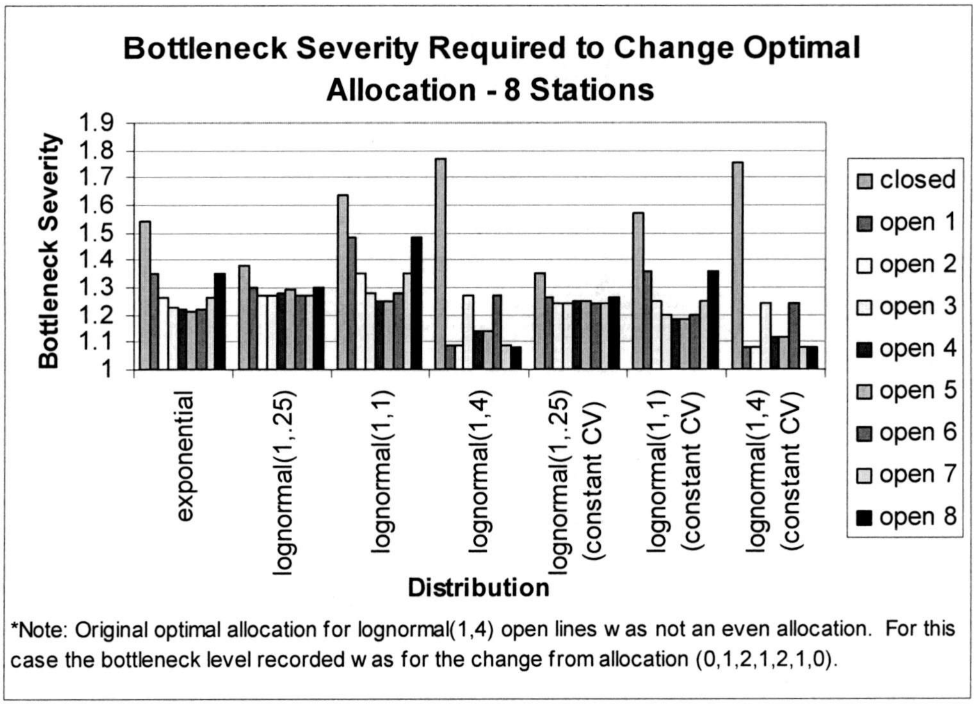

Figure 7: Bottleneck Severity Level Required to Change to a Non-Even Buffer Allocation as Optimal – Eight Stations

Second, the bottleneck severity appeared to be affected by the CV of the workstations. The larger the CV, the larger the bottleneck required to create a change in the optimal buffer allocation for closed production lines (Figure 8). For open productions lines the effect of the CV was less clear. In open lines the bottleneck severity looked to also depend on both the location of the bottleneck and the length of the line.

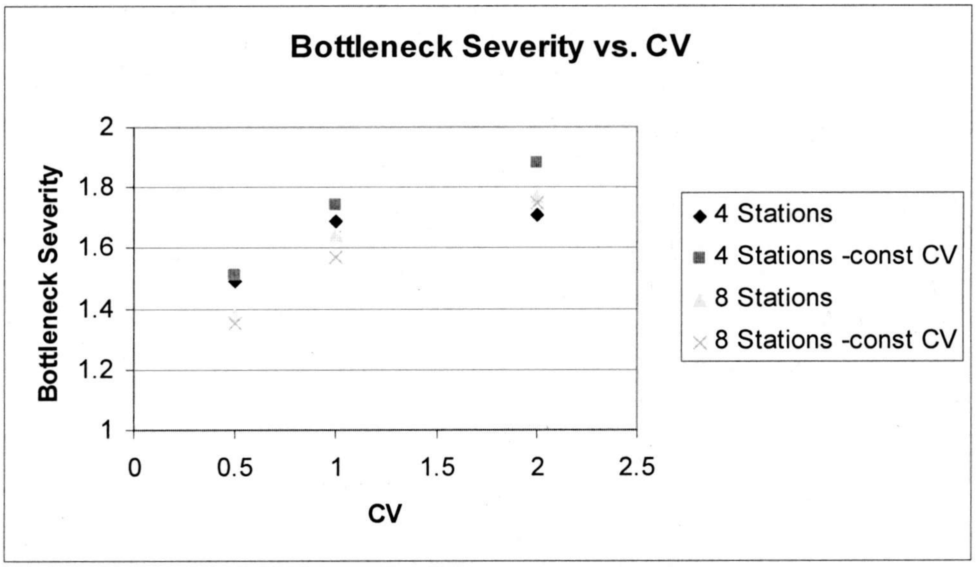

Figure 8: Bottleneck Severity that First Causes a Change to the Optimal Buffer Allocation at Various CV Levels for Lognormal Distribution

With regards to the length of the line, the bottleneck severity level required to create a shift in the optimal buffer allocation appeared to decrease with an increase in line length (Figure 9). There was only one case in which the severity was larger in the eight-station line (closed production line with lognormal(1,4) distributed processing times). Since the bottleneck location in open lines seemed to have an effect, when comparing four-station lines and eight-station lines, the bottleneck needed to be in the same position. Both the first workstation (labeled "open-end" on graph) and one of the inner most stations (second station in four-station line, fourth station in eight-station line labeled "open-center" on graph) were chosen for comparisons in Figure 9.

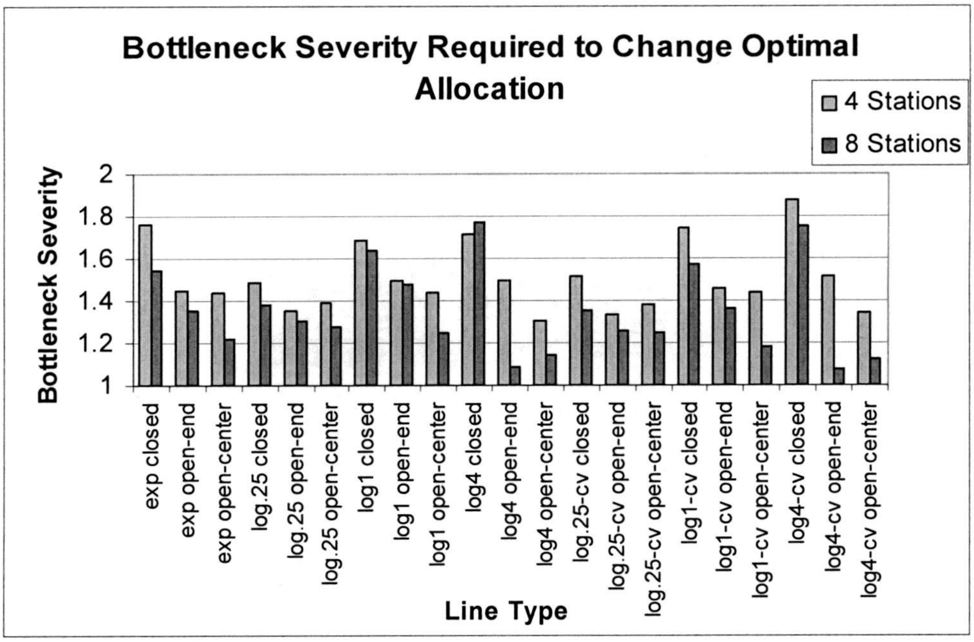

Figure 9: Bottleneck Severity Levels that First Cause a Change to the Optimal Buffer Allocation for Different Line Types

Finally, in open production lines the bottleneck severity level necessary to create a change in optimal allocation depended on the location of the bottleneck. It appeared to be symmetric around the center of the line. That is, the change occurred at the same time when the bottleneck was located at the first or last station, the second or second to last station, the third or third to last station, etc. This result is not surprising because the reversibility property of open lines addressed by Muth (1979) and Conway et al. (1988) among others states that two serial lines, which are mirror images of each other, will have the same throughput.

Experiment 3

As shown in experiment 1, the evenly distributed buffer allocation is optimal for a balanced line and is optimal in the presence of a single bottleneck up to some bottleneck severity level. For many cases this level exceeded a 50% increase in the mean effective processing time relative to other stations in the line (e.g., when the CV ≥ 1). Since lines are not normally designed to have such severe bottlenecks but they do occur, a natural question is: can an even buffer allocation be used as a general design rule for closed and /or open production lines with minimal risk of losing much in terms of throughput? Specifically of interest was how much worse an evenly distributed buffer allocation performs in lines where it is not the optimal buffer allocation (i.e., the allocation resulting in the greatest throughput). This is the question addressed in experiment 3.

The throughput was calculated for both the optimal allocation of the balanced line (one buffer between each station except for lognormal(1,4) open lines) and the optimal (or near optimal) allocation at a given bottleneck level. With four-station lines, the optimal buffer allocation was determined after estimating the throughput for all possible allocations through simulation. The optimal buffer allocation for eight -station lines was determined using the algorithm developed by Vergara (2005). For a complete listing of the optimal buffer allocations see Appendix D. Once the throughput for each allocation was calculated, the percentage of throughput lost by using the even allocation rather than the optimal allocation was determined (*1-(even allocation throughput/optimal allocation throughput*)). These percentages were determined for all the same setups explored in experiment 2 at multiple bottleneck levels. For eight-station open lines the bottleneck was located in the first half of the line only (due to reversibility property). Table 12 shows an example of this data (for complete data see Appendix E). A series of graphs (percentage loss vs. bottleneck severity) were then plotted to determine if anything of interest could be noted.

Table 12: Percentage of Throughput Lost by Using an Even Buffer Allocation when it is not Optimal (Bottleneck Mean = 2.00)

	4 stations, Bottleneck mean = 2.00					
	Processing Time Distribution	**Closed**	**Open: Position of Bottleneck**			
			1	**2**	**3**	**4**
	exponential	0.4751%	2.2611%	1.1542%	1.1478%	2.2588%
const. Var.	lognormal(1,.25)	0.0988%	0.0568%	0.0512%	0.0508%	0.0570%
	lognormal(1,1)	0.7581%	1.3896%	0.9104%	0.9160%	1.3923%
	lognormal(1,4)	0.5621%	1.6431%	1.2656%	1.2771%	1.6505%
const. CV	lognormal(1,.25)	0.3262%	0.2474%	0.1882%	0.1878%	0.2477%
	lognormal(1,1)	0.4654%	1.7920%	1.0202%	1.0290%	1.7971%
	lognormal(1,4)	0.1806%	1.3628%	1.0032%	1.0131%	1.3670%

The first thing that stood out when examining the data from Figures 10-13 was that the throughput difference between the optimal and the even allocations initially increased as the bottleneck increased. The increase occurred quickly at first, but then gradually lessened until some point was reached at which the difference started to decrease (note: this was true for all but the lognormal(1,4) distribution because the maximum was not reached in every case). This pattern makes sense because there will be a point at which a bottleneck is so severe that it dominates the line and any buffer allocation scheme will have little effect on the throughput.

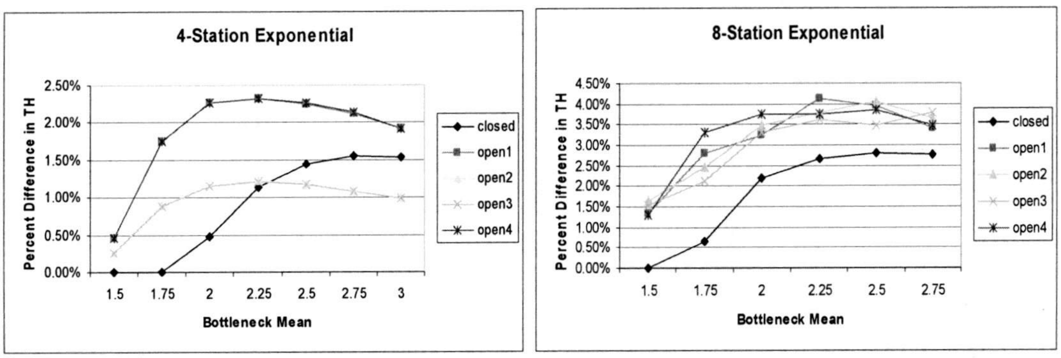

Figure 10: Percentage below Optimum Throughput of Even Buffer Allocation for Exponential Lines

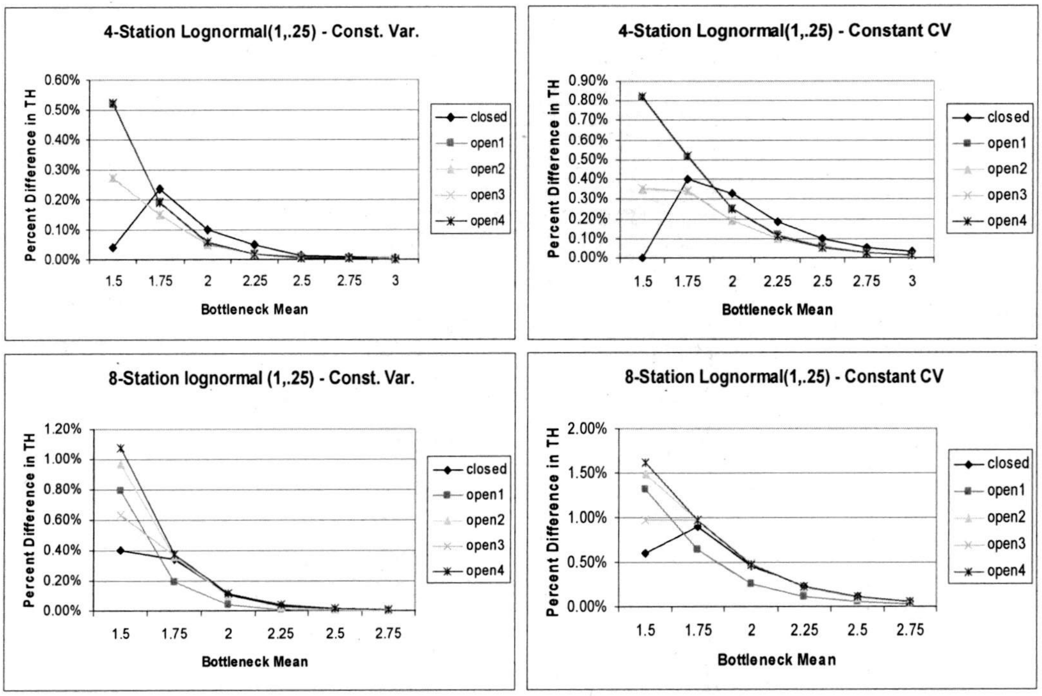

Figure 11: Percentage below Optimum Throughput of Even Buffer Allocation for Lognormal(1,.25) Lines

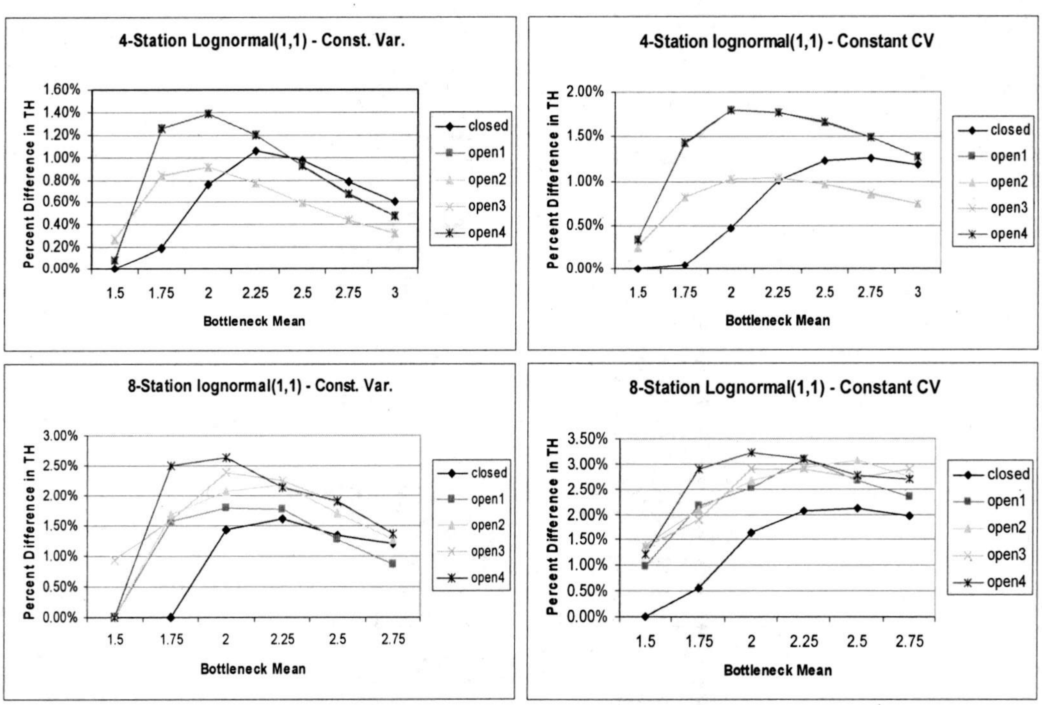

Figure 12: Percentage below Optimum Throughput of Even Buffer Allocation for Lognormal(1,1) Lines

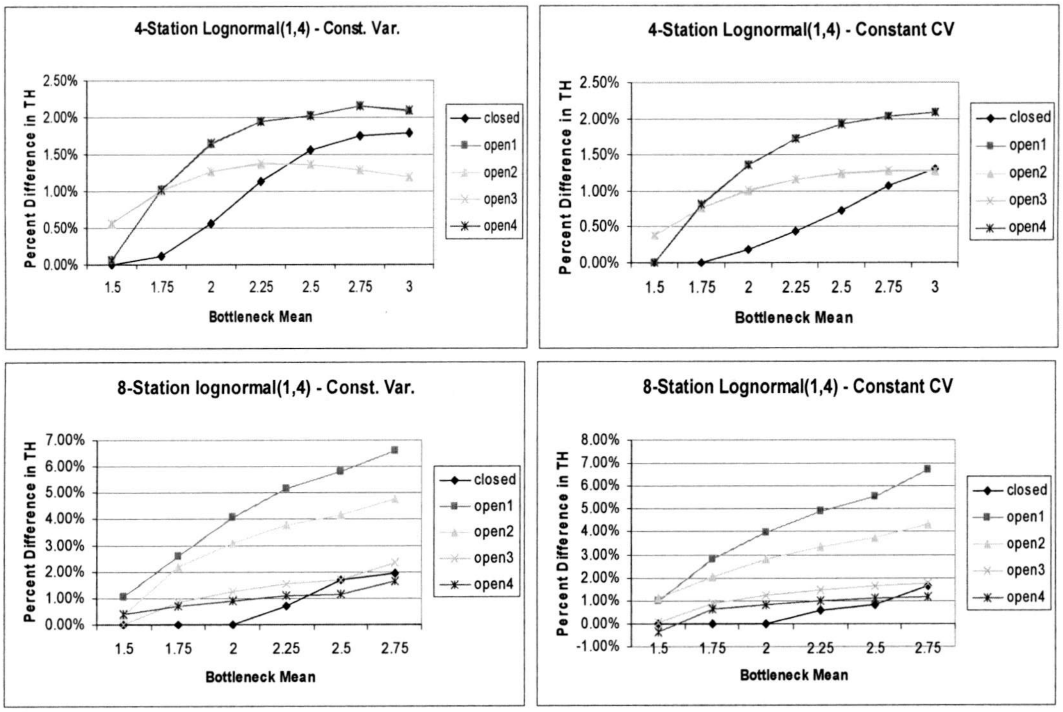

Figure 13: Percentage below Optimum Throughput of Even Buffer Allocation for Lognormal(1,4) Lines

Although all the different lines tested followed the same basic pattern several differences were seen based on line length, CV of the non-bottleneck stations, CV of just the bottleneck station, and type of line (open or closed). First, the most obvious effect seen was that by which the line length affected the maximum throughput difference seen. In every case (in which a maximum could be determined) the maximum throughput difference between allocations was greater in the line with eight stations than the one with four stations. Table 13 shows the data for the closed production lines tested. Also, when looking at the data for closed lines, the maximum difference appeared to occur sooner for longer lines.

Table 13: Maximum Throughput Difference between Optimal and Even Buffer Allocation for Closed Production Line Setups

	4 Stations		8 Stations	
Processing Time Distribution	**Maximum Difference Observed**	**Bottleneck severity at Maximum**	**Maximum Difference Observed**	**Bottleneck severity at Maximum**
Exponential	1.5538%	2.75	2.7862%	2.5
Lognormal (1,.25)	0.2347%	1.75	0.3978%	1.50
Lognormal (1,1)	1.0542%	2.25	1.6062%	2.25
Lognormal (1,4)	1.7826%	3.00*	1.9377%	2.75*
Lognormal (1,.25) - Const. CV	0.4014%	1.75	0.9033%	1.75
Lognormal (1,1) - Const. CV	1.2494%	2.75	2.1136%	2.50
Lognormal (1,4) - Const. CV	1.2956%	3.00*	1.6479%	2.75*

*maximum bottleneck level tested was 3.00 and 2.75 for four and eight stations respectively

The next observation is the effect of the CV of the non-bottleneck stations. For the smallest value of CV (CV = 0.5) tested, the maximum throughput difference occurred quicker and at the lower bottleneck severity levels than when compared to the larger CV values. The maximum difference always occurred by a bottleneck severity level of 1.75 and often prior to a 1.50 severity level. See

Figure 11 for the graphs of all lognormal(1,.25) data. In addition, the degree of this difference was relatively small. In the four-station closed lines, the maximum difference seen was only 0.4% and in the eight-station closed lines the difference was only 0.9%. This suggests that for small values of CV, that an even buffer allocation could be a decent option no matter what kind of bottleneck is present. For the largest CV (CV = 2), the maximum occurred at much larger bottleneck severity levels, often occurring at bottleneck severity levels in excess of 2.75 (See Figure 13). Also, for closed lines the maximum throughput difference observed appeared to increase with CV. This observation does not hold for open production lines.

Not only did the CV of the non-bottleneck workstations appear to have an effect in closed production lines, but so did the CV of the bottleneck station. When the mean only was increased to create a bottleneck (constant variance) the overall CV of the bottleneck decreased; therefore the CV of the bottleneck station was smaller than in the constant CV case. When examining these two cases (shown best in Figure 12) the effect of the bottleneck's CV appeared to be similar to that of the CV of the non-bottleneck stations. The maximum percentage difference in throughput was greater for the constant CV (larger bottleneck CV) case, and the bottleneck severity level when this occurred was equal to or greater than the constant variance (smaller bottleneck CV) case. Unlike the effect of CV of the bottleneck station, the CV of the non-bottleneck station appears to behave the same in the open production lines tested as the closed production lines.

When looking at the behavior of open lines vs. the closed lines, a few things were seen. First, although the first change from an even allocation being optimal was later (in terms of bottleneck severity level) for closed lines, the maximum throughput difference at higher bottlenecks sometimes exceeded that of open lines. For example, looking at Figure 10 in the four-station line, the closed line reached a maximum throughput difference of just more than 1.50%; where as, the open line with the bottleneck located at the second or third workstation reached an approximate 1.20% difference. However, when there were eight stations, the closed production lines never had a maximum difference that was more than the open lines. It seems that for longer lines the even allocation will perform as well or better (with regards to optimal buffer allocation) in closed lines than open lines no matter the severity of the bottleneck. Also, in closed lines the maximum difference occurred sooner (with respect to the bottleneck severity level) as the line length increased. However, the behavior of open lines was not consistent. For example, when looking at open lines with the bottleneck located in the center of the line (second and fourth stations in the four-station and eight-station lines respectively), the maximum difference occurred in the four-station line sooner than the eight-station line with the exponential distribution, but occurred later with the lognormal(1,1) distribution with constant CV.

Experiment 4

Experiment 4 examined buffer allocation in automated/unreliable lines. Unreliable lines are ones in which the workstation processing time variability comes from random breakdowns. The first part of experiment 4 was to determine if the evenly allocated buffer pattern is optimal for closed automated/unreliable production lines. Lines of four stations and eight stations were tested with buffer capacities of four and eight respectively. Like experiment 1, differences in simulated throughput for lines with different buffer allocations were determined using multiple comparison tests. The LSD, Bonferroni, and Scheffe methods were applied. Not all test results are shown here, but they can be verified from the data in Appendix G. Various MTTR and MTBF combinations were tested which represented a wide range of CV values for the effective processing times. The CV values resulting from the MTTR/MTBF combinations are shown in Table 14 (assuming a constant processing time of one).

Table 14: CVs (Mean Processing Time = 1) for Various MTTR and MTBF Combinations

CV	MTBF	MTTR
0.1961	50	1
0.9091	50	5
1.6667	50	10
0.1400	100	1
0.6734	100	5
1.2856	100	10
0.0891	250	1
0.4384	250	5
0.8600	250	10
0.0631	500	1
0.3131	500	5
0.6201	500	10
0.0447	1000	1
0.2225	1000	5
0.4428	1000	10

Test results showed that the even allocation of buffers has the highest average throughput for all buffer allocations tested (see Table 15). However, the results of the multiple comparison tests were not the same as in the reliable lines tested in experiment 1. For reliable closed lines, the even buffer allocation always had a statistically significantly higher throughput than any other buffer allocation, and the different homogenous groups of buffer allocations could be predicted based on the decomposition principle. However, in the unreliable case, the results were much more variable. Sometimes the even allocation was significantly higher than all other allocations (see Table 16), but sometimes there were multiple buffer allocations that resulted in the same throughput (see Table 17). Often the result would depend on the statistical test being employed. This suggests that there may be less throughput difference between the different buffer allocations in unreliable lines than in reliable ones. While the even allocation may not be the only optimal solution, it is always among the optimal solution(s) for unreliable closed lines. More extensive simulations may be able to detect significant differences that were not found in this experimentation.

Table 15: Buffer Allocation Resulting in the Greatest Throughput

MTBF	MTTR	Greatest TH	Allocation
50	1	0.948203	(1-1-1-1)
50	5	0.738164	(1-1-1-1)
50	10	0.57472	(1-1-1-1)
100	1	0.97309	(1-1-1-1)
100	5	0.847019	(1-1-1-1)
100	10	0.726101	(1-1-1-1)
250	1	0.989001	(1-1-1-1)
250	5	0.932045	(1-1-1-1)
250	10	0.867779	(1-1-1-1)
500	1	0.994403	(1-1-1-1)
500	5	0.964445	(1-1-1-1)
500	10	0.928457	(1-1-1-1)
1000	1	0.997178	(1-1-1-1)
1000	5	0.981846	(1-1-1-1)
1000	10	0.9628	(1-1-1-1)

Table 16: LSD Multiple Comparison of Throughput for Unreliable (MTTR=1, MTBF=50) Four-Station Closed Line (95% Confidence Level)

Method: 95.0 percent LSD

allocation	Count	Mean	Homogeneous Groups						
(0,0,4,0)	5	0.927471	X						
(0,4,0,0)	5	0.927478	X						
(4,0,0,0)	5	0.927485	X						
(0,0,0,4)	5	0.927486	X						
(1,3,0,0)	5	0.935654		X					
(3,1,0,0)	5	0.935656		X					
(1,0,0,3)	5	0.935656		X					
(3,0,0,1)	5	0.935663		X					
(0,0,1,3)	5	0.935677		X					
(0,3,1,0)	5	0.935681		X					
(0,1,3,0)	5	0.935682		X					
(0,0,3,1)	5	0.935685		X					
(1,0,3,0)	5	0.938588			X				
(0,1,0,3)	5	0.938596			X				
(3,0,1,0)	5	0.938596			X				
(0,3,0,1)	5	0.938605			X				
(2,2,0,0)	5	0.939347				X			
(2,0,0,2)	5	0.939347				X			
(0,0,2,2)	5	0.939386				X			
(0,2,2,0)	5	0.939397				X			
(0,1,2,1)	5	0.943255					X		
(1,2,1,0)	5	0.943266					X		
(1,0,1,2)	5	0.943271					X		
(2,1,0,1)	5	0.943277					X		
(0,2,0,2)	5	0.94411						X	
(2,0,2,0)	5	0.944111						X	
(0,2,1,1)	5	0.944593							X
(1,1,2,0)	5	0.944601							X
(2,1,1,0)	5	0.944602							X
(1,1,0,2)	5	0.944602							X
(0,1,1,2)	5	0.944609							X
(1,2,0,1)	5	0.944614							X
(1,0,2,1)	5	0.944639							X
(2,0,1,1)	5	0.944644							X
(1,1,1,1)	5	0.948203								X

Table 17: Scheffe Multiple Comparison of Throughput for Unreliable (MTTR=1, MTBF=1000) Four-Station Closed Line (95% Confidence Level)

Method: 95.0 percent Scheffe

allocation	Count	Mean	Homogeneous Groups
(0,4,0,0)	5	0.995943	X
(0,0,0,4)	5	0.995943	X
(4,0,0,0)	5	0.995943	X
(0,0,4,0)	5	0.995943	X
(1,0,0,3)	5	0.996433	X
(3,1,0,0)	5	0.996433	X
(1,3,0,0)	5	0.996433	X
(3,0,0,1)	5	0.996433	X
(0,1,3,0)	5	0.996439	X
(0,3,1,0)	5	0.996439	X
(0,0,3,1)	5	0.996448	X
(0,0,1,3)	5	0.996448	X
(1,0,3,0)	5	0.996608	XX
(0,3,0,1)	5	0.996608	XX
(0,1,0,3)	5	0.996609	XX
(3,0,1,0)	5	0.996609	XX
(2,0,0,2)	5	0.996658	XXX
(2,2,0,0)	5	0.996658	XXX
(0,2,2,0)	5	0.996667	XXXX
(0,0,2,2)	5	0.996692	XXXXX
(2,1,0,1)	5	0.996885	XXXXX
(1,2,1,0)	5	0.996885	XXXXX
(1,0,1,2)	5	0.996894	XXXXX
(0,1,2,1)	5	0.996896	XXXXX
(2,0,2,0)	5	0.996943	XXXX
(0,2,0,2)	5	0.996943	XXXX
(1,2,0,1)	5	0.996964	XXX
(1,1,2,0)	5	0.996964	XXX
(2,1,1,0)	5	0.996965	XXX
(1,1,0,2)	5	0.996965	XXX
(0,2,1,1)	5	0.996967	XXX
(2,0,1,1)	5	0.996969	XXX
(0,1,1,2)	5	0.996979	XX
(1,0,2,1)	5	0.996981	XX
(1,1,1,1)	5	0.997178	X

The second part of experiment 4 examined the differences in sensitivity of the optimal buffer allocation to the severity of a single bottleneck in closed lines and open lines, as was done in experiment 2 for reliable lines. In this experiment the MTTR of a single workstation was increased in order to create a bottleneck. A bottleneck severity level up to $t_e = 2$ time units was investigated. Looking at the results for four-station lines, a few observations can be made. First, the even allocation pattern remained optimal at the largest bottleneck level for all MTTR/MTBF combinations (Table 18). Since the MTTR/MTBF combinations had similar CVs to the reliable lines tested but exceeded the bottleneck levels that created a shift in the optimal solution, it can be concluded that closed unreliable lines are less sensitive to bottlenecks than closed reliable lines.

Table 18: Buffer Allocation Resulting in the Greatest Throughput with $t_e = 2$ at Bottleneck Station

MTBF	MTTR	Allocation	Greatest TH
50	1	(1-1-1-1)	0.488339
50	5	(1-1-1-1)	0.426412
50	10	(1-1-1-1)	0.366722
100	1	(1-1-1-1)	0.494334
100	5	(1-1-1-1)	0.459642
100	10	(1-1-1-1)	0.421725
250	1	(1-1-1-1)	0.50001
250	5	(1-1-1-1)	0.484986
250	10	(1-1-1-1)	0.467014
500	1	(1-1-1-1)	0.501666
500	5	(1-1-1-1)	0.493894
500	10	(1-1-1-1)	0.484284
1000	1	(1-1-1-1)	0.505376
1000	5	(1-1-1-1)	0.50138
1000	10	(1-1-1-1)	0.496352

Next, the results for open lines were examined. Again, the results differed from the reliable cases examined in experiment 2. When determining a starting optimal solution, an even allocation was only best when the MTTR equaled one time unit, and depending on the comparison method was not always the only optimal solution. In cases where the MTTR equaled five or ten time units, the best solutions were where all the buffers were placed in the center location, but again there were usually many optimal solutions and in the extreme case with MTTR = 10 and MTBF =1000, there was no difference between any of the allocations (Table 19). These results also indicate that buffer allocation is less important in unreliable open lines. Finally, although the results are not as straight forward as in experiment 2, the data does indicate that the closed lines are less affected by the presence of a bottleneck than the open line equivalents. For many of the open line setups and all of the closed lines examined, there was no change in the optimal buffer allocation for the bottleneck levels tested. A change in the optimal buffer allocation was seen at some point in all the open lines with an MTTR equal to one time unit, and in two cases where the MTTR equaled five time units. When a shift occurred, it usually happened at a very low bottleneck severity. The lines where a change in the optimal allocation occurred are summarized in Table 20 (optimal solution designated to be allocation with the highest average throughput when multiple allocations determined optimal).

Table 19: LSD Multiple Comparison of Throughput for Unreliable (MTTR=10, MTBF=1000) Four-Station Open Line (95% Confidence Level)

Method: 95.0 percent Scheffe

allocation	Count	Mean	Homogeneous Groups
(3,0,0,6)	5	0.962878	X
(0,0,3,6)	5	0.962913	X
(2,0,1,6)	5	0.962985	X
(1,0,2,6)	5	0.962994	X
(2,1,0,6)	5	0.963162	X
(0,1,2,6)	5	0.963186	X
(1,1,1,6)	5	0.963223	X
(1,2,0,6)	5	0.963371	X
(0,2,1,6)	5	0.963387	X
(0,3,0,6)	5	0.963506	X

Table 20: Four-Station Open Lines where an Increase in Bottleneck Severity Caused a Change in the Optimal Buffer Allocation

Original MTTR	MTBF	bottleneck station	MTTR value causing shift
1	50	1	2
1	50	2	3
1	100	1	3
1	100	2	3
1	250	1	3
1	250	2	3
1	500	1	3
1	500	2	4
1	1000	1	3
1	1000	2	7
5	50	1	8
5	100	1	62

After performing the first two parts of experiment 4, it appeared that the unreliable lines were behaving differently than the reliable lines, so a more direct comparison between unreliable and reliable lines was made. For each unreliable line setup tested, a reliable equivalent (same t_e and CV) was also tested. Only closed lines were investigated. Looking at the multiple comparison tests on throughput for the equivalent unreliable and reliable lines, it can be seen that there is much lower throughput differences between different buffer allocations in the unreliable lines. Tables 19 and 20 show results from applying the Scheffe multiple comparison method for throughput from equivalent reliable and unreliable lines with $t_e = 1.01$ and CV = 0.313.

Table 21: Scheffe Multiple Comparison of Throughput for Reliable (with Lognormal (1.01, 0.1) Processing Times) Four-Station Closed Line (95% Confidence Level)

Method: 95.0 percent Scheffe

allocation	Count	Mean	Homogeneous Groups
(4,0,0,0)	5	0.764229	X
(0,0,0,4)	5	0.764232	X
(0,4,0,0)	5	0.76425	X
(0,0,4,0)	5	0.764265	X
(1,3,0,0)	5	0.790606	X
(3,1,0,0)	5	0.790609	X
(0,0,1,3)	5	0.790611	X
(1,0,0,3)	5	0.790611	X
(0,0,3,1)	5	0.790612	X
(3,0,0,1)	5	0.790615	X
(0,3,1,0)	5	0.79062	X
(0,1,3,0)	5	0.790624	X
(0,2,2,0)	5	0.792	X
(2,2,0,0)	5	0.792002	X
(2,0,0,2)	5	0.792005	X
(0,0,2,2)	5	0.792017	X
(3,0,1,0)	5	0.811381	X
(0,1,0,3)	5	0.811382	X
(1,0,3,0)	5	0.811384	X
(0,3,0,1)	5	0.811391	X
(2,0,2,0)	5	0.822406	X
(0,2,0,2)	5	0.822406	X
(1,0,1,2)	5	0.833706	X
(1,2,1,0)	5	0.833709	X
(2,1,0,1)	5	0.833709	X
(0,1,2,1)	5	0.833749	X
(2,0,1,1)	5	0.836303	X
(2,1,1,0)	5	0.836316	X
(1,1,0,2)	5	0.836317	X
(1,2,0,1)	5	0.836323	X
(1,1,2,0)	5	0.836325	X
(1,0,2,1)	5	0.836329	X
(0,1,1,2)	5	0.836356	X
(0,2,1,1)	5	0.836363	X
(1,1,1,1)	5	0.886243	X

Table 22: Scheffe Multiple Comparison of Throughput for Unreliable (MTTR=5, MTBF=500) Four-Station Closed Line (95% Confidence Level)

Method: 95.0 percent Scheffe

allocation	Count	Mean	Homogeneous Groups
(0,4,0,0)	5	0.961104	X
(0,0,4,0)	5	0.961104	X
(4,0,0,0)	5	0.961109	X
(0,0,0,4)	5	0.96111	X
(1,3,0,0)	5	0.962375	XX
(3,1,0,0)	5	0.962376	XX
(1,0,0,3)	5	0.962377	XX
(3,0,0,1)	5	0.962378	XX
(0,1,3,0)	5	0.962382	XX
(0,3,1,0)	5	0.962382	XX
(0,0,1,3)	5	0.962397	X
(0,0,3,1)	5	0.962398	X
(1,0,3,0)	5	0.962811	XX
(0,3,0,1)	5	0.962815	XX
(3,0,1,0)	5	0.962816	XX
(0,1,0,3)	5	0.962817	XX
(2,2,0,0)	5	0.963422	XXX
(2,0,0,2)	5	0.963422	XXX
(0,2,2,0)	5	0.963434	XXX
(0,0,2,2)	5	0.963466	XXX
(1,2,1,0)	5	0.963618	XXX
(2,1,0,1)	5	0.96362	XXX
(1,0,1,2)	5	0.96363	XXX
(0,1,2,1)	5	0.963642	XXX
(2,1,1,0)	5	0.963968	XX
(1,1,0,2)	5	0.963969	XX
(1,1,2,0)	5	0.963976	XX
(2,0,1,1)	5	0.963976	XX
(1,2,0,1)	5	0.963981	XX
(0,2,1,1)	5	0.963984	XX
(0,1,1,2)	5	0.963993	XX
(1,0,2,1)	5	0.963993	XX
(2,0,2,0)	5	0.964229	X
(0,2,0,2)	5	0.964229	X
(1,1,1,1)	5	0.964445	X

Reliable and unreliable lines were then compared by looking at the throughput differences between the optimal buffer allocation, the second best allocation, and the worst allocation (in terms of throughput). Table 23 shows an example for one line (results for other lines are in Appendix F). One observation is that the throughput values for different buffer allocations vary much more in the reliable lines than the unreliable ones. The difference between the optimal throughput and the second best allocation's throughput (Figure 14) and the difference between the optimal and worst throughput (Figure 15) were plotted for each setup tested. The unreliable lines had little throughput difference between the various allocations at all CV values. The reliable lines on the other hand, had much larger differences that depended on the CV. Peaking at CV values between 0.1 and 0.2, the throughput differences rapidly increase at first and then slowly decrease.

Table 23: Reliable and Unreliable Line Comparison of Throughput for the Optimal, Second Best, and Worst Buffer Allocation

Setup	Optimal Allocation	TH	2^{nd} Best Allocation	TH	Worst Allocation	TH
MTTR=1, MTBF=50	(1-1-1-1)	0.948203	(2-0-1-1)	0.944644	(0-0-4-0)	0.927471
log(1.02,.04)	(1-1-1-1)	0.931590	(0-1-1-2)	0.882390	(4-0-0-0)	0.828330

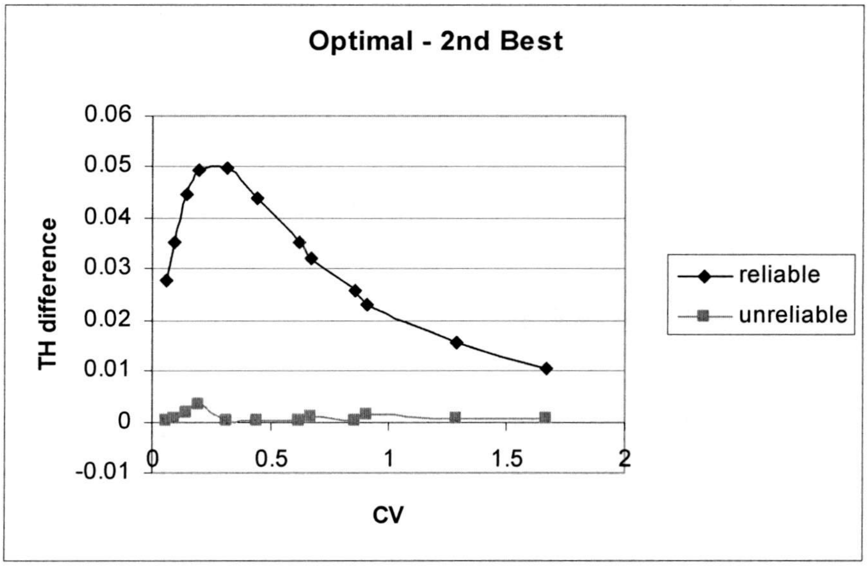

Figure 14: Throughput Differences between the Best and Second Best Buffer Allocations vs. CV

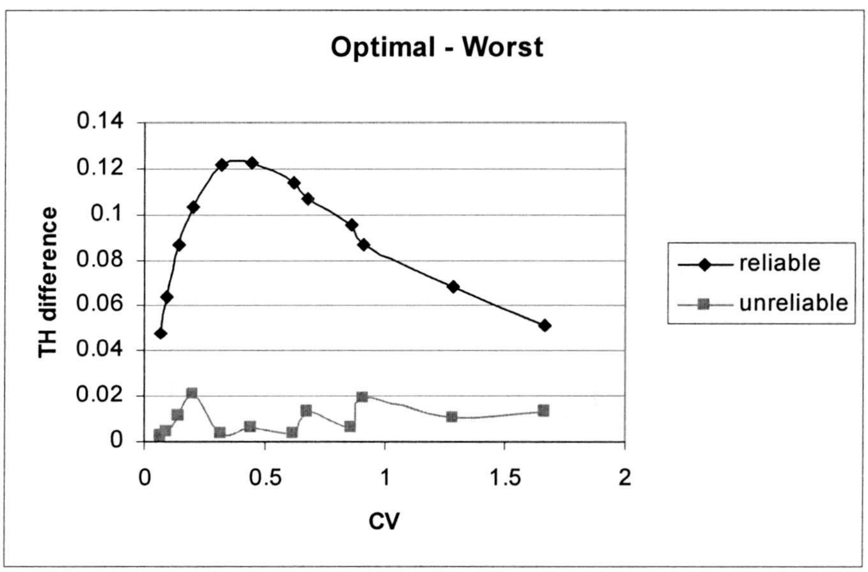

Figure 15: Throughput Differences between the Best and Worst Buffer Allocations vs. CV

Also, by plotting the optimal throughput rates against CV (Figure 16) it can be seen, as expected, that the throughput of the unreliable lines decrease with an increasing CV just like the reliable line. However, it does not occur at a smooth consistent rate like the reliable line, so the unreliable lines were separated into three groups based on the MTTR and replotted (Figure 17). This graph shows that for an MTTR = 1, the unreliable lines behave very much like the reliable lines, but as the MTTR increases, the throughput differences between the reliable and unreliable lines increases.

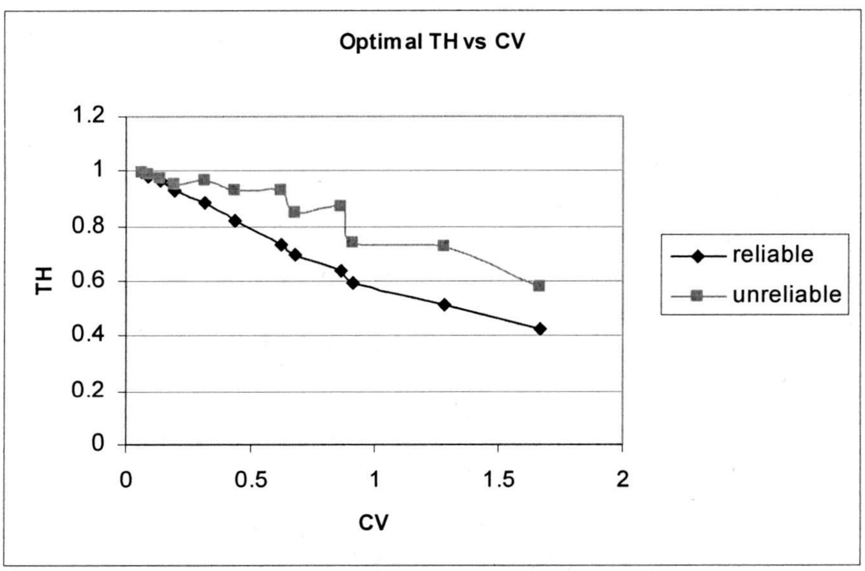

Figure 16: Optimal Throughput for Lines with Various CVs

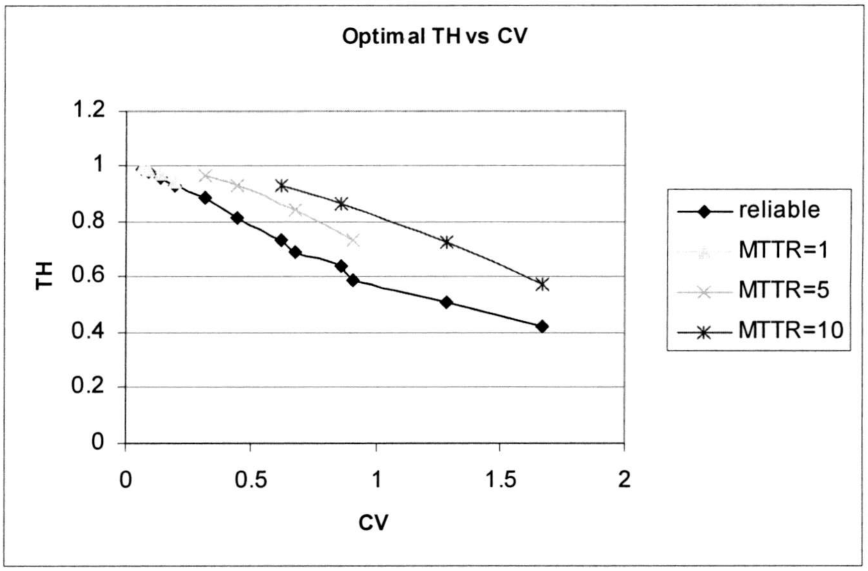

Figure 17: Optimal Throughput for Lines with Various CVs (Unreliable Lines Separated by MTTR)

Finally, reliable and unreliable lines were compared when there was a bottleneck present. Looking at four-station lines with a CV of 1 and 0.5, the optimal buffer allocation was found at different bottleneck severity levels. Then the throughputs for the optimal, even, and worst buffer allocations were recorded (see Appendix F). Using the LSD multiple comparison test (95% confidence level) on the unreliable lines, no significant differences in throughputs could be determined for any of the buffer allocations when CV = 1. Only the most extreme allocations (all buffers assigned to the same location) could be seen as significantly different when CV = 0.5. However, the even allocation had the highest average throughput when CV = .05 and the third highest when CV = 1, so it was chosen to be the optimal solution for the balanced line, and was used in the comparisons. Several observations were made. First, the reliable lines behaved consistently with the observations made in experiment 2, but the unreliable lines tested did not behave the same as the earlier ones tested. The optimal allocation did not change until the bottleneck severity was 1.75 and 2 time units (for line with CV = 0.5 and 1 respectively) in the reliable line, but the optimal solution changed almost immediately (bottleneck severity = 1.05 time units) in the unreliable lines. The difference in results for the unreliable lines could be due to the fact that both the MTTR and MTBF were changed to keep a constant CV and both had larger starting values than previously tested. Although the optimal solution changed almost immediately, the actual difference in throughputs between the new optimal solution and the even allocation was quite small. These differences are shown in Table 22.

Table 24: Percent Loss of Throughput of the Even Buffer Allocation from Optimal Allocation in Unreliable Lines

Bottleneck Severity te	**TH loss of even from optimal allocation**	
	CV=1	**CV=.5**
1.05	0.034%	0.103%
1.10	0.039%	0.123%
1.25	0.040%	0.127%
1.50	0.041%	0.124%
1.75	0.042%	0.121%
2.00	0.040%	0.116%

The second observation was made when comparing the optimal allocation throughput at the different bottleneck levels as seen in Figure 18. The difference in throughput between the reliable and unreliable is quite large at small bottleneck levels, but difference becomes smaller as the bottleneck severity increases. Figure 18 also shows that the throughput values are nearly identical between the two unreliable lines, suggesting that CV does not affect the throughput as much as in reliable lines.

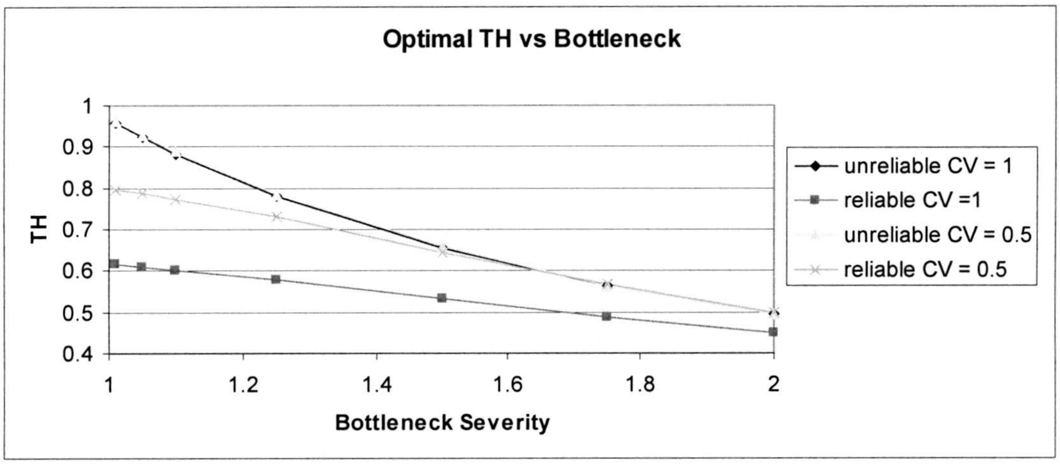

Figure 18: Throughput of Optimal Buffer Allocation at Various Bottleneck Severity Levels

The last observation was made by looking at Figure 19 in which the difference in throughput between the best allocation and worst allocation was plotted for the various bottleneck levels. For the reliable lines there was quite a large difference showing the importance of buffer allocation. The difference decreased as the bottleneck worsened, but at differing rates depending on the CV. The buffer allocation appeared to be more important at low bottleneck levels when the CV was smaller, but less important at higher bottleneck levels. For unreliable lines, the difference was small for all bottleneck levels at both CV levels. In Table 23 the maximum percentage lost due to a non-optimal allocation was determined for the various bottleneck levels by taking (*1-(worst allocation throughput / optimal allocation throughput*)). This table suggests that the potential lost throughput is greater when the CV is smaller, but still very minimal. If the absolute worst buffer allocation was chosen, the maximum throughput loss would only be about 0.4%, further supporting the conclusion that buffer allocation is rather unimportant when dealing with unreliable lines.

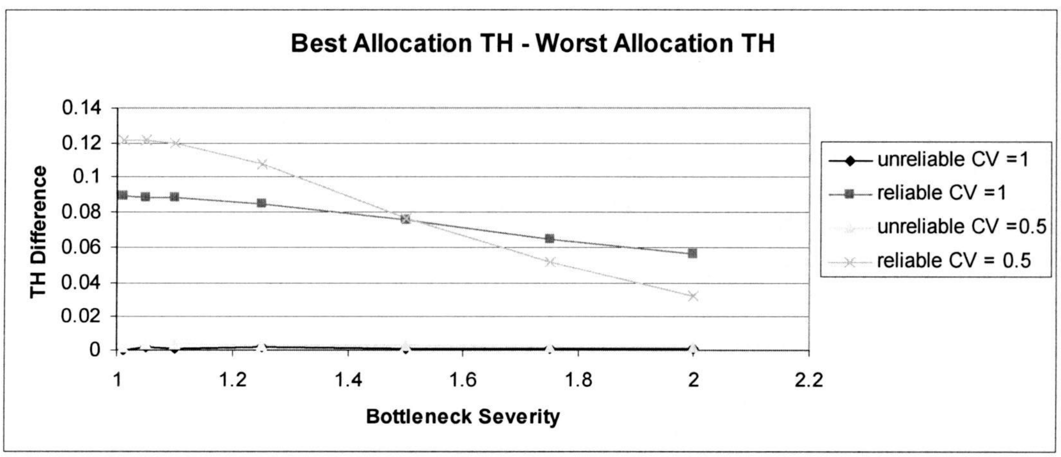

Figure 19: Throughput Difference between Optimal Buffer Allocation and Worst Buffer Allocation at Various Bottleneck Severity Levels

Table 25: Percent Loss of Throughput of the Worst Buffer Allocation from the Optimal Allocation

Bottleneck Severity te	TH loss of worst from optimal allocation	
	CV=1	CV=.5
1.01	0.043%	0.146%
1.05	0.183%	0.373%
1.10	0.147%	0.408%
1.25	0.198%	0.402%
1.50	0.104%	0.379%
1.75	0.147%	0.332%
2.00	0.161%	0.337%

Experiment 5

Experiment 5 was designed to see if the previous results for four and eight-station lines hold for longer lines. This was done by examining lines with 20 stations. First, the hypothesis that the evenly distributed buffer allocation is optimal for a balanced closed production line was tested. The exponential, lognormal(1,1), and lognormal(1,.25) processing time distributions were used in the examination of reliable lines. To examine unreliable lines, a line where the workstations had t_e = 1.01 time units and CV = 1 was used. The LSD, Bonferroni, and Scheffe multiple comparison tests were used to determine if the even buffer allocation was optimal. The data (Tables 26-28) showed the same results as for the four and eight-station lines. The even distribution was significantly better than all other allocations at a 95% confidence level using the LSD and Bonferroni method for all three reliable line distributions. Only the results of the Scheffe method applied to the lognormal

(1, 1) line concluded the even buffer allocation was not significantly better than all other allocations. The evidence suggests that an even buffer allocation is indeed optimal for a longer closed line with identical stations. These results were further supported by the results from running the algorithm developed by Vergara (2005) to find the optimal/near optimal buffer allocation. Table 26 provides the ten best buffer allocations ranked by average throughput for the reliable line with exponentially distributed processing times and Figure 20 shows the scatterplot of the data for the 10 best allocations. The even allocation results in a much higher throughput. The results for the two lines tested with lognormally distributed processing times are similar to the exponential case and are shown in Tables 25 and 26 and Figures 18 and 19.

Table 26: 10 Best Buffer Allocations Ranked by Average Throughput for Line with Exponentially Distributed Processing Times

Exponential Distribution		
Rank	**Allocation**	**Avg. TH**
1	(1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1)	0.522560
2	(1,1,1,1,0,2,1,1,1,1,1,1,1,1,1,1,1,1,1,1)	0.518266
3	(1,1,1,2,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1)	0.518259
4	(1,1,1,1,1,0,2,1,1,1,1,1,1,1,1,1,1,1,1,1)	0.518248
5	(1,1,1,1,1,1,1,1,1,1,0,2,1,1,1,1,1,1,1,1)	0.518244
6	(1,1,1,1,1,1,0,2,1,1,1,1,1,1,1,1,1,1,1,1)	0.518244
7	(1,1,1,1,1,1,2,0,1,1,1,1,1,1,1,1,1,1,1,1)	0.518242
8	(1,1,2,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1)	0.518241
9	(1,1,1,1,1,1,1,1,0,2,1,1,1,1,1,1,1,1,1,1)	0.518234
10	(2,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1)	0.518230

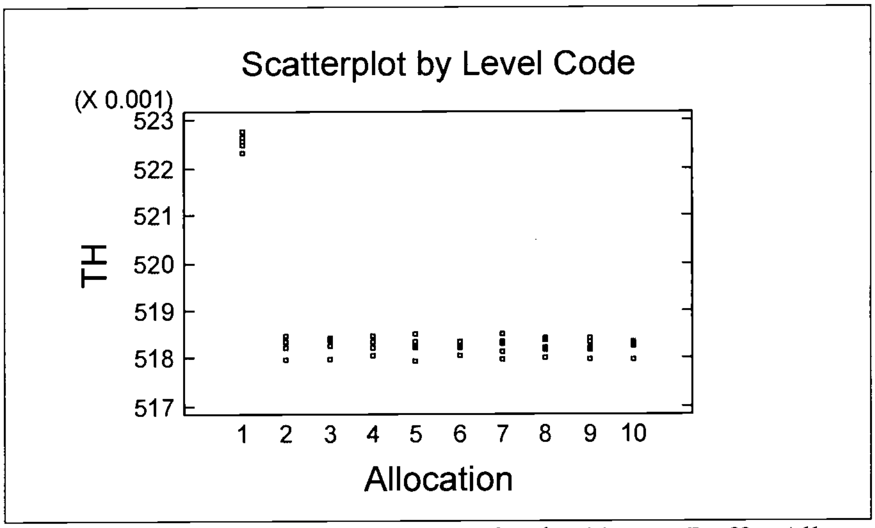

Figure 20: Scatterplot of Throughput Results for the 10 Best Buffer Allocations for Line with Exponentially Distributed Processing Times

Table 27: 10 Best Buffer Allocations Ranked by Average Throughput for Line with Lognormally(1,1) Distributed Processing Times

Lognormal(1,1) Distribution		
Rank	**Allocation**	**Avg. TH**
1	(1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1)	0.506652
2	(1,1,1,1,1,1,1,1,1,1,1,1,0,2,1,1,1,1,1,1)	0.503494
3	(1,1,1,1,1,1,1,1,1,1,1,2,0,1,1,1,1,1,1,1)	0.503481
4	(1,1,1,1,1,1,1,1,1,1,1,1,1,2,0,1,1,1,1,1)	0.50348
5	(1,1,1,1,1,1,1,1,1,1,1,1,1,0,2,1,1,1,1,1)	0.503479
6	(1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,2,1,1,1,1)	0.503473
7	(1,1,1,2,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1)	0.503473
8	(1,1,1,1,1,1,0,2,1,1,1,1,1,1,1,1,1,1,1,1)	0.503469
9	(1,1,1,1,1,1,1,1,1,1,1,1,2,0,1,1,1,1,1,1)	0.503469
10	(1,1,1,1,1,1,1,1,1,1,2,0,1,1,1,1,1,1,1,1)	0.503466

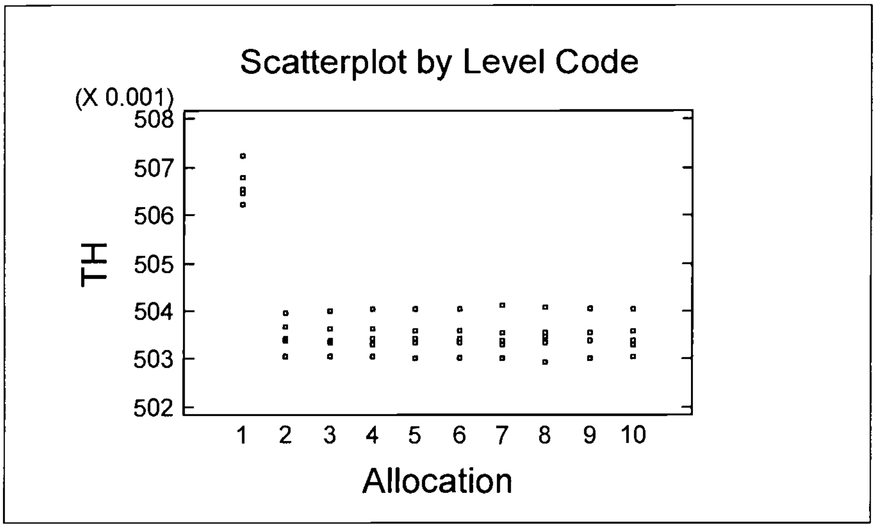

Figure 21: Scatterplot of Throughput Results for the 10 Best Buffer Allocations for Line with Lognormally(1,1) Distributed Processing Times

Table 28: 10 Best Buffer Allocations Ranked by Average Throughput for Line with Lognormally(1,.25) Distributed Processing Times

Lognormal(1,.25) Distribution		
Rank	**Allocation**	**Avg. TH**
1	(1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1)	0.7437072
2	(1,1,1,1,1,1,1,1,1,1,1,1,2,0,1,1,1,1,1)	0.7329688
3	(1,1,1,1,1,1,1,1,1,1,1,1,1,0,2,1,1,1,1)	0.7329524
4	(1,1,1,1,1,1,1,1,1,1,1,1,1,2,0,1,1,1,1)	0.7329506
5	(1,1,1,1,1,1,1,1,1,1,1,2,0,1,1,1,1,1,1)	0.732948
6	(1,1,1,1,1,1,1,1,1,1,1,1,0,2,1,1,1,1,1)	0.7329436
7	(1,1,1,2,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1)	0.7329422
8	(1,1,1,1,1,0,2,1,1,1,1,1,1,1,1,1,1,1,1)	0.7329392
9	(1,1,1,1,1,1,1,1,1,1,1,1,0,2,1,1,1,1,1)	0.732939
10	(1,1,1,1,0,2,1,1,1,1,1,1,1,1,1,1,1,1,1)	0.7329382

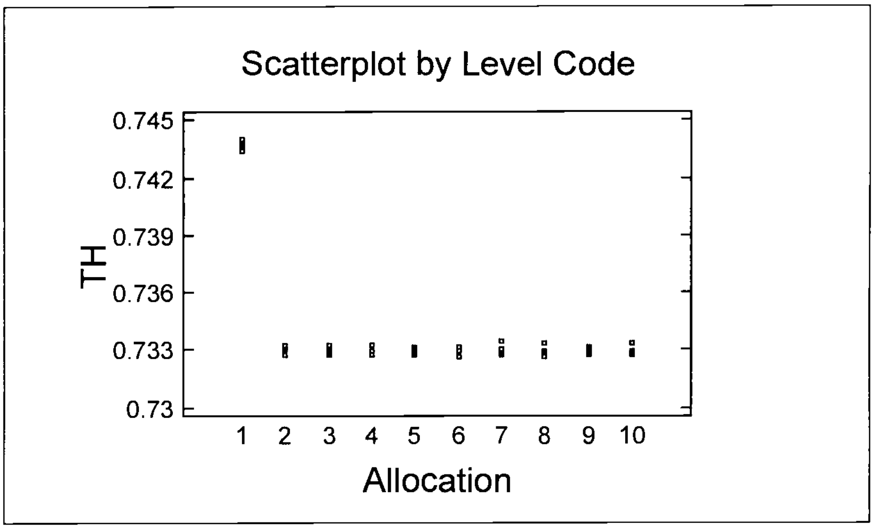

Figure 22: Scatterplot of Throughput Results for the 10 Best Buffer Allocations for Line with Lognormally(1,.25) Distributed Processing Times

For longer lines with unreliable stations, the results are different from reliable lines but consistent with the results seen in experiment 4. With the unreliable line, the buffer allocation giving the highest throughput was not the even allocation. The even allocation was the fifth best among the allocations tested (see Table 29). Yet the multiple comparison tests did not detect a significant difference between the throughputs for any of the different allocations. A scatterplot of throughput for the 10 best allocations (Figure 23) shows no discernable differences. Only a small fraction of the possible allocations were tested, so there is a chance that an allocation that was not tested could be shown to be significantly better than the even allocation. The algorithm developed by Vergara (2005) found that the allocation (1,0,0,0,0,1,0,0,0,9,0,0,0,1,0,0,0,0,8,0) was optimal/near optimal. However, when the even allocation was compared to the optimal/near optimal solution determined by the algorithm, the even allocation had a significantly higher throughput using a one-sided paired t-test. Under the null hypothesis that *the mean difference* (*the even allocation's throughput – the algorithm allocation's throughput*) $= 0$ and alternate hypothesis of *the mean difference* ≥ 0, a p-value $<$ 0.05 was calculated and thus the null hypothesis can be rejected with a confidence of 95%. This result suggests that the even allocation can be considered an optimal or near optimal buffer allocation.

Table 29: 10 Best Buffer Allocations Ranked by Average Throughput for an Unreliable Line (Mean = 1.0, t_e = 1.01, CV =1.0)

Unreliable Line (CV = 1)		
Rank	**Allocation**	**Avg. TH**
1	(2,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0)	0.82828
2	(1,2,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0)	0.8282678
3	(1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,1,2)	0.8282674
4	(1,1,1,1,1,1,2,0,1,1,1,1,1,1,1,1,1,1,1,1)	0.828264
5	(1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1)	0.8282636
6	(1,2,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1)	0.8282626
7	(1,1,1,1,1,1,2,1,0,1,1,1,1,1,1,1,1,1,1,1)	0.8282604
8	(1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,2,1)	0.8282604
9	(1,1,1,1,1,1,1,1,1,1,2,0,1,1,1,1,1,1,1,1)	0.8282598
10	(1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,2,1,1,1)	0.8282594

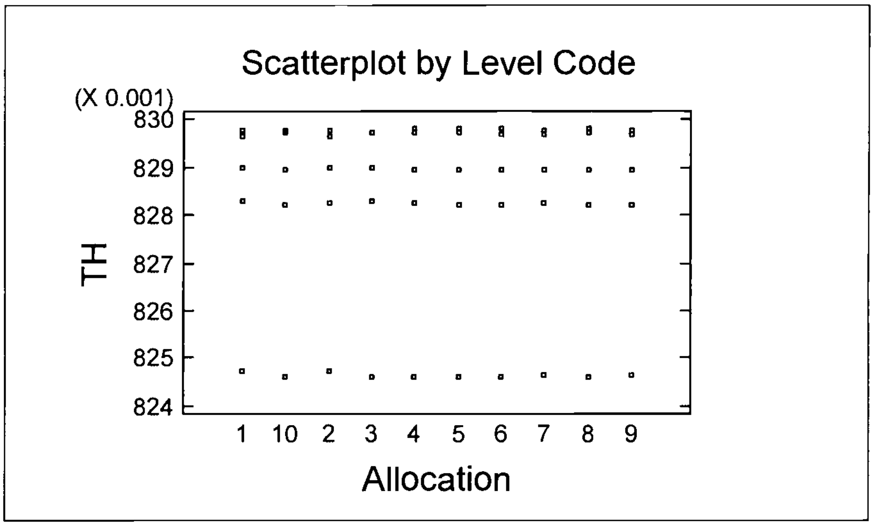

Figure 23: Scatterplot of Throughput for the 10 Best Buffer Allocations Ranked by Average Throughput for an Unreliable Line (Mean = 1.0, t_c = 1.01, CV = 1.0)

Next, the sensitivity of the best buffer allocation to the severity level of a bottleneck (as in experiment 2) was investigated for longer lines. This was done by plotting the bottleneck severity level causing the optimal allocation to shift from the original (for the balanced line) optimal allocation for various line setups. In Figure 24, a closed reliable line was compared to an open line using lines in which the stations had exponentially distributed processing times. When comparing the buffer allocation sensitivity to bottleneck severity in closed lines and in open lines (with the bottleneck station located at the first, fifth, and tenth station), it is clear that the closed line still requires a larger bottleneck severity level to create a change in the optimal allocation. These confirm the results obtained for lines with four and eight stations. A plot of the results from the four and eight-station lines along with the 20-station line (Figure 25) shows the behavior. In all three cases, the closed line is the least sensitive to bottleneck severity, and an open line with the bottleneck located at the end of the line is less sensitive than an open line with the bottleneck located in the center of the line.

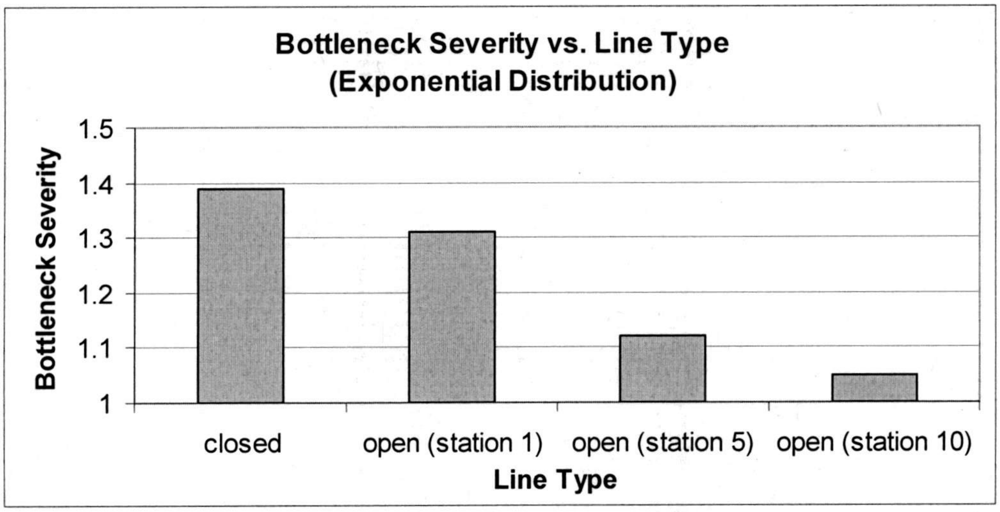

Figure 24: Bottleneck Level Required to Cause a Shift in Optimal Allocation for Various Exponential Lines

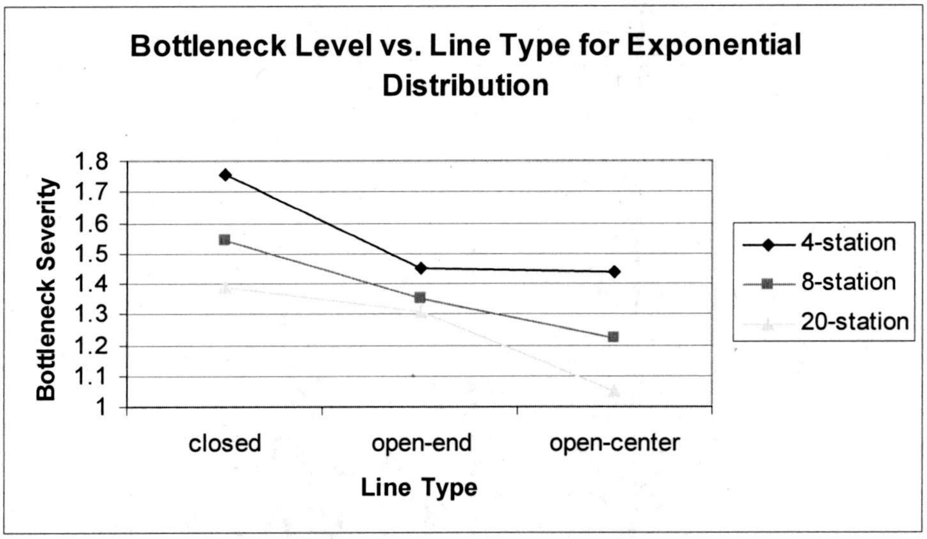

Figure 25: Bottleneck Level Required to Shift the Optimal Buffer Allocation in Exponentially Distributed Lines

Also, in experiment 2 it was shown that the CV had an effect on the sensitivity of a buffer allocation to bottleneck severity. Figure 26 shows that the 20-station line behaves the same as the shorter lines in this regard. The bottleneck required to cause the shift in optimal allocation was plotted against different distributions (processing times of the individual workstations). For the smaller CV (lognormal(1,.25)) the required bottleneck severity level causing a change in the best buffer allocation is lower than for the larger CV (lognormal(1,1) and exponential) There is not much difference between the two distributions with the larger CVs (both have CV =1).

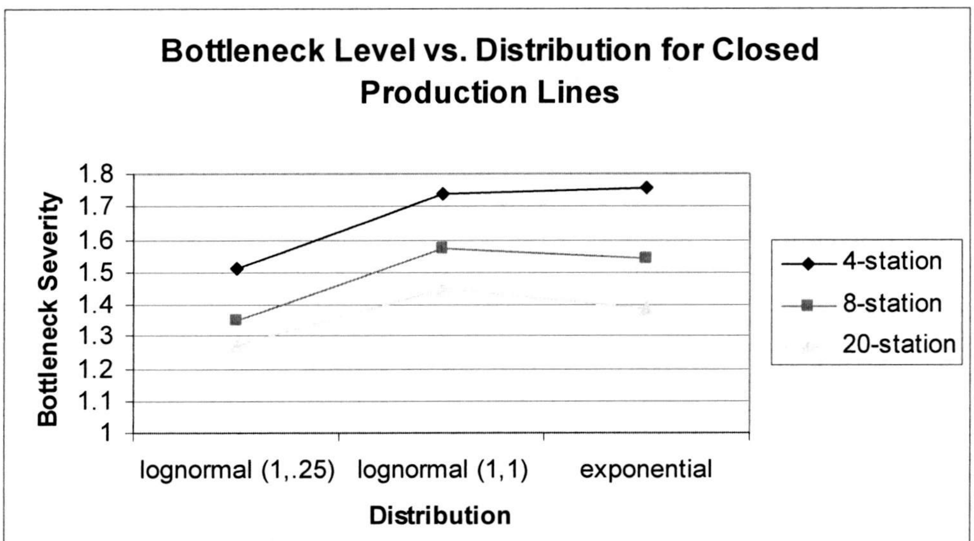

Figure 26: Bottleneck Level Required to Shift Optimal Buffer Allocation in Closed Lines with Various Workstation Distributions

Finally, the effect of line length is clearly seen from Figures 22 and 23. The 20-station case stays consistent with the results from experiment 2. As the line length increases, the necessary bottleneck level to create a shift in optimal allocation decreases.

CONCLUSIONS

Based on the results from the five experiments several conclusions were reached regarding closed serial production lines. Since a difference was observed between the behavior of reliable and unreliable lines the conclusions have been separated into those for reliable production lines and those for unreliable production lines.

Reliable Production Lines

- The optimal/near optimal buffer allocation for a balanced closed production line will be the allocation in which the buffers are as evenly distributed across the line as possible.
- The optimal buffer allocation in a closed line is less likely to change when a bottleneck is present than in an open line.
- In open lines, if a bottleneck is located at the end of the line the optimal allocation is less affected than if the bottleneck is located in the center of the line.
- The bottleneck level required to create a change in optimal buffer allocation increases as the CV's of stations increase.
- The bottleneck level required to create a change in optimal buffer allocation decreases as line length increases.

- The maximum potential throughput lost from using the even buffer allocation rather than the optimal allocation increases as line length, CV of the bottleneck station, and CV of the non-bottleneck stations increases.
- The maximum potential throughput loss is less for closed lines than open lines for longer line lengths.

Unreliable Production Lines

- The even buffer allocation is not always the best solution (highest throughput) for a balanced unreliable closed production line, but no significant difference between it and the best buffer allocation can be detected.
- The buffer allocation pattern is much less important in unreliable lines. There is less throughput difference between different allocations (compared to reliable lines).
- An even allocation appears to result in a throughput that is near optimal for all cases and all bottleneck levels.
- The throughput difference between an even buffer allocation and the best (highest throughput) buffer allocation decreases as the CV of the workstations increase.
- If MTTR is equal to 1, the optimal throughput of the unreliable line is similar to the equivalent reliable line, but as MTTR increases, the difference between the two increases with the unreliable line performing better.

FUTURE RESEARCH

The potential for additional experiments and investigations from this research is virtually limitless. This section suggests some of the possible future research that can be performed. It is separated into two parts describing two different types of research. The first part describes some ways that this research can be changed and improved, while the second part discusses ways to extend or continue this research.

Limitations/Improvements

There were some limitations to this research that may be improved upon in future research. Some of these limitations have been identified and suggestions for improving them have been made. They are as follows:

1. All buffer allocations in longer lines were not tested.

In order to determine optimal allocations the average throughputs of multiple buffer allocations were determined and then compared to each other. This method was very time consuming. The time needed to calculate the throughput for each allocation increased as the number of workstations in the line increased, as did the number of possible allocations. As a result, the time requirements became greater as the line length increased. To compensate, the number of buffer allocations considered was reduce, which in effect reduced the reliability of the results. This research could be improved if a better more efficient way of determining optimal buffer allocations is used. This would make it easier to test longer lines, and generate more data.

2. Precision of the model and results is arbitrary

The precision of the model and results could also be improved. Although the parameters of the simulation model chosen seemed reasonable, some could be changed in order to give more precision. For example, more jobs could be simulated or additional replications could be performed. The only downside to doing this is that the simulation run time will increase. Within the experiments themselves, additional precision could be added by decreasing the increment size when increasing bottlenecks. The increment size used in this research was arbitrarily chosen and could easily be reduced if higher precision is desired. In particular, Experiments 3 and 4 could be improved by using smaller increments.

3. Inferences do not extend beyond the lines tested

The use of fixed factors in the experimental design used limits the inference space so that any conclusions do not apply to lines outside of what was tested. Conclusions regarding behavior outside of what was tested are speculative (not statistically supported), but can lead to more thorough investigations. For example, one conclusion reached was that the necessary bottleneck severity to change the optimal buffer allocation decreases as the line length increases. However, this conclusion was only based on results from lines of length 4, 8, and 20. To draw inferences to all line lengths an experiment would need to be designed with line length as a random factor.

Extensions/Continuations

There was not time to investigate everything in this research so there are numerous ways to extend it. The following is a small list of suggestions:

1. Different line types

This research only investigated a very small portion of the many different production line types. One potential area for further research is to examine some of the other types of lines. What happens if the line uses a pull production control mechanism instead of a push mechanism, part transfer is synchronous instead of asynchronous, or uses a blocking-before-service instead of blocking-after-service blocking scheme? Any one of these factors could be studied.

2. Different processing time distributions

The only two processing time distributions used in this research were the exponential and lognormal distributions. Many other types of distributions can and have been used to model reliable production lines. Therefore, one logical direction to take this research is to look at some of these other distributions and whether or not they behave the same as the exponential and lognormal.

3. Different unreliable parameters and assumptions

Like using different processing time distributions in reliable lines, there are also other ways to model unreliable lines. The unreliable lines in this research were assumed to have constant processing times, but this doesn't have to be so. Also, the exponential distribution was used to model time to failures and repair times, but other distributions could be used instead. Also, failures could be modeled as time-dependent instead of operation-dependent. Varying any, or all, of the factors would be a natural extension to this research.

4. Different parameters (different line lengths, buffers, CVs, etc.)

Another easy extension of this research would be to use more levels of the various line factors. For many of the experiments only two different levels were used for the different factors. Using two factors would allow for the detection of linear relationships, but there is no guarantee that the relationships are linear. The more factor levels used the better the understanding of the relationships will be.

5. Create bottlenecks by increasing variance or MTBF only

In reliable lines, bottlenecks can be created by a workstation with a higher mean processing time than the other stations or a higher processing time variance. In unreliable lines, bottlenecks are created when a station has a higher MTTR or a lower MTBF than the other stations. In this research the method to create bottlenecks was to increase the mean in reliable lines and the MTTR in unreliable.

Although MTBFs were changed in some cases to keep constant variances, they were never looked at alone for causing bottlenecks. Examining if the cause of the bottleneck has an effect on the observed behaviors would be an interesting continuation of this research.

6. Multiple bottlenecks

Another interesting extension would be to look at multiple bottlenecks. How would having two or more bottlenecks present in the line have effected the results? Also, what effect does the position of those bottlenecks have? It seems likely that larger lines could have multiple bottlenecks so this would be a worthwhile area of research to pursue.

7. Observation from investigation of unreliable lines

One of the more interesting observations made during this research was that for many unreliable line setups, there was no statistically significant difference between any buffer allocation for any bottleneck level. This means that the buffers could be placed anywhere and it wouldn't affect the throughput. More research into this area would be useful to see under what circumstances this observation holds true. If true a lot of time can be saved when deciding on buffer placement.

BIBLIOGRAPHY

Atwater, J.B. and Chakravorty, S.S. (1996) Impact of restricting the flow of inventory in serial production systems. *International Journal of Production Research,* 34, 2657-2669.

Blackstone Jr., J.H. (2004) On the shape of protective capacity in a simple line. *International Journal of Production Research,* 42, 629-637.

Buzacott, J.A. (1967) Automatic transfer lines with buffer stocks. *International Journal of Production Research*, 5, 183-200.

Buzacott, J.A., and Shantikumar, J.G. (1993) *Stochastic models of manufacturing systems.* New York: Prentice-Hall.

Conway, R., Maxwell, W., McClain, J.O., and Thomas, L.J. (1988) The role or work in-process inventory in serial production lines. *Operations Research,* 36, 229-241.

Dallery, Y., and Towsley, D. (1991) Symmetry property of the throughput in closed tandem queueing networks with finite buffers. *Operations Research Letters*, 10, 541-547.

El-Rayah, T. (1979) The effect of inequality of interstage buffer capacities and operation time variability on the efficiency of production line systems. *International Journal of Production Research,* 17, 77-89.

Enginarlar, E., Li, J., Meerkov, S.M., and Zhang, R.Q. (2002) Buffer capacity for accommodating machine downtime in serial production lines. *International Journal of Production Research*, 40, 601-624.

Freeman, M. (1964) The effects of breakdowns and interstage storage on production line capacity. *The Journal of Industrial Engineering*, 15, 194-200.

Han, M., and Park, D. (2002) Performance analysis and optimization of cyclic production lines. *IIE Transactions*, 34, 411-422.

Harris, J., and Powell, S. (1999) An algorithm for optimal buffer placement in reliable serial lines. *IIE Transactions*, 31, 287-302.

Hillier, F., and Boling, R. (1966) The effect of some design factors on the efficiency of production lines with variable operation times. *The Journal of Industrial Engineering*, 17, 651-658.

Hillier, F., and Boling, R. (1979) On the optimal allocation of work in symmetrically unbalanced production line systems with variable operation times. *Management Science*, 25, 721-728.

Hillier, F., and So, K. (1991a) The effect of the coefficient of variation of operation times on the allocation of storage space in production line systems. *IIE Transactions*, 23, 198-206.

Hillier, F., and So, K. (1991b) The effect of machine breakdowns and interstage storage on the performance of production line systems. *International Journal of Production Research*, 29, 2043-2055.

Hillier, F., and So, K. (1993) Some data for applying the bowl phenomenon to large production line systems. *International Journal of Production Research*, 31, 811-822.

Hillier, F., So, K., and Boling, R. (1993) Notes: toward characterizing the optimal allocation of storage space in production line systems with variable processing times. *Management Science*, 39, 126-132.

Hillier, M. (2000) Characterizing the optimal allocation of storage space in production line systems with variable processing times. *IIE Transactions*, 32, 1-8.

Hopp, W., and Spearman, M. (1991) *Factory physics (2nd ed.)*. New York : McGraw-Hill.

Inman, R. (1999) Empirical evauluation of exponential and independence assumptions in queueing models of manufacturing systems. *Production and Operations Management*, 8, 409-432.

Kadipasaoglu, S., Xiang, W., Hurley, S., and Khumawala, B. (2000) A study on the effect of the extent and location of protective capacity in flow systems. *International Journal of Production Economics*, 63, 217-228.

Kim D.S., F. Lin and D.M. Kulkarni (2002) An upper bound for the carrier number in a closed serial production system operating under production blocking. *IEEE Transactions on Automatic Control*, 47, 1134-1138.

Kim, S., and Lee, H. (2001) Allocation of buffer capacity to minimize average work-in-process. *Production Planning & Control*, 12, 706-716.

Kirkavak, N., and Dincer, C. (1999) The general behavior of pull production systems: the allocation problems. *European Journal of Operational Research*, 119, 479-494.

Law, A., and Kelton, W. (2001) *Simulation modeling and analysis (2nd ed.)*. New York : McGraw-Hill.

Liu, X., Zhuang, L., and Buzacott, J. (1992) A decomposition method for throughput analysis of cyclic queues with production blocking. *Proc. 2nd Int. Workshop Queueing Networks With Finite Capacity,* Research Triangle Park, NC, May 28-29, 445-475.

Muth, E. (1979) The reversibility property of production lines. *Management Science*, 25, 152-158.

Papadopoulos, H.T., and Vidalis, M.I. (2001) Minimizing WIP inventory in reliable production lines. *International Journal of Production Economics*, 70, 185-197.

Pike, R., and Martin, G. (1994) The bowl phenomenon in unpaced lines. *International Journal of Production Research*, 32, 483-499.

Powell, S. (1994) Buffer allocation in unbalanced three-station serial lines. *International Journal of Production Research*, 32, 2201-2217.

Powell, S., and Pyke, D. (1996) Allocation of buffers to serial production lines with bottlenecks. *IIE Transactions*, 28, 18-29.

Ramsey, F., and Shafer, D. (2002) *The statistical sleuth: A course in methods of data analysis (2nd ed.)*. Australia ; Pacific Grove, CA : Duxbury/Thomson Learning.

Sheskin, T. (1976) Allocation of interstage storage along an automatic production line. *AIIE Transactions*, 8, 146-152.

Spinellis, D., Papadopoulos, C., and MacGregor Smith, J. (2000) Large production line optimization using simulated annealing. *International Journal of Production Research*, 38, 509-541.

Vergara Arteaga, Hector A. (2005) Efficient heuristics for buffer allocation in closed serial production lines. MS Thesis. Oregon State University.

Vouros, G., and Papadopoulos, H. (1998) Buffer allocation in unreliable production lines using a knowledge based system. *Computers & Operations Research*, 25, 1055-1067.

Yamashina, H., and Okamura, K. (1983) Analysis of in-process buffers for multi-stage transfer line systems. *International Journal of Production Research*, 21, 183-195.

APPENDICES

APPENDIX A: RANDOM NUMBER TESTS

The following tests were performed to check various characteristics of the uniform(0, 1) random variables which are used to generate the exponential and lognormal random variables used in the simulation. For further explanation see Law and Kelton (1991). The results of these tests can be found in Appendix G.

Runs-up Test

The runs test is a direct test of independence. In the runs-up test the sequence of the uniform (0, 1) random variables (U_i's) was checked for subsequences in which the U_i's monotonically increased, which is referred to as a "run up". The length of the runs up were counted and added to the appropriate r_i. Where,

$$r_i = \begin{cases} \text{number of runs up of length i} & \text{for i} = 1, 2, \ldots, 5 \\ \text{number of runs up of length} \geq 6 & \text{for i} = 6 \end{cases}$$

The test statistic $R = \frac{1}{n}\sum_{i=1}^{6}\sum_{j=1}^{6} a_{ij}(r_i - nb_i)(r_j - nb_j)$ was then compared to a chi-square distribution with 6 df (degrees of freedom) using the null hypothesis that the U_i's are IID random variables.

When this test was performed R was calculated as 7.4728. The chi-square distribution with 6 df and an α of 0.95 is 12.59159. Therefore, the null hypothesis fails to be rejected and independence can be assumed.

Note that aij is the *(i,j)*th element of the matrix shown below.

Table 30: Runs-up Test Matrix used to Calculate Test Statistic

4,529.4	9,044.9	13,568	18,091	22,615	27,892
9,044.9	18,097	27,139	36,187	45,234	55,789
13,568	27,139	40,721	54,281	67,852	83,685
18,091	36,187	54,281	72,414	90,470	111,580
22,615	45,234	67,852	90,470	113,262	139,476
27,892	55,789	83,685	111,580	139,476	172,860

and

$$(b_1, b_2, b_3, b_4, b_5, b_6) = \left(\frac{1}{6}, \frac{5}{24}, \frac{11}{120}, \frac{19}{720}, \frac{29}{5040}, \frac{1}{840}\right)$$

Chi-Square Test

The chi-square test was performed in order to check for uniformity along the interval from 0 to 1. To do so the interval from 0 to 1 was divided into k equal length subintervals and n uniform random variables were generated (U_1, U_2, ..., U_n). Then the number of U_i's that fell within the j_{th} subinterval (j = 1, 2, ..., k) were totaled (f_j).

Letting

$$\chi^2 = \frac{k}{n}\sum_{j=1}^{k}\left(f_j - \frac{n}{k}\right)^2,$$

χ^2 has a chi-square distribution with k-1 df for large values of n. Then with the null hypothesis being that the U_i's are IID U(0,1) random variables, the null hypothesis is rejected if $\chi^2 > \chi^2_{k-1,1-\alpha}$, where $\chi^2_{k-1,1-\alpha}$ is the upper 1-α critical point of the chi-squared distribution with k-1 df.

Using k = 100 and n = 100,000, χ^2 was calculated to be 945.18 which was less than $\chi^2_{k-1,1-\alpha}$ which equaled 1073.643 for α = 0.95. Therefore, the null hypothesis fails to be rejected and it can be assumed that the random variables are indeed uniformly distributed.

The values of k and n were arbitrarily chosen as 100 and 100,000 respectively, but complied with Kelton and Law (1991) who suggested that k should be at least 100 and n/k at least 5.

Serial Tests

The serial tests provide indirect tests on the independence of the random variables by checking uniformity in higher dimensions. This again was done using a chi-square test. The interval [0,1] was divided into k equal size subintervals and arrays of uniform (0, 1) random variables ($V_1 = (U_1, U_2, \ldots, U_d)$, $V_2 = (U_{d+1}, U_{d+2}, \ldots, U_{2d})$, … , V_n) were generated. Next the number of U_i's were totaled ($f_{j_j j_2 \cdots j_d}$) that had the first component in subinterval j_1 and second component in subinterval j_2, etc.

Letting

$$\chi^2(d) = \frac{k^d}{n} \sum_{j_1=1}^{k} \sum_{j_2=1}^{k} \cdots \sum_{j_d=1}^{k} \left(f_{j_1 j_2 \cdots j_d} - \frac{n}{k^d} \right)^2,$$

$\chi^2(d)$ has a chi-square distribution with k^d -1 df for large values of n. Then with the null hypothesis being that the V_i's are IID random variables, the null hypothesis is rejected if $\chi^2(d) > \chi^2_{k^d-1,1-\alpha}$, where $\chi^2_{k^d-1,1-\alpha}$ is the upper 1-α critical point of the chi-squared distribution with k^d-1 df.

Both d = 2 and d = 3 were tested. With d = 2, k=100, and n=100,000, $\chi^2(d)$ was calculated to be 9,843.80 and $\chi^2_{k^d-1,1-\alpha}$ = 10,232.70 (for α = 0.95).

Since $\chi^2(d) < \chi^2_{k^d-1,1-\alpha}$ the null hypothesis cannot be rejected and it is assumed that the random variables are uniform in the 2nd dimension. With d = 3, k=25, and n=100,000, $\chi^2(d)$ was calculated to be 15,784.38 and $\chi^2_{k^d-1,1-\alpha}$ = 15,915.90 (for α = .95). Since $\chi^2(d) < \chi^2_{k^d-1,1-\alpha}$ the null hypothesis fails to be rejected and it is assumed that the random variables are uniform in the 3rd dimension.

Correlation Tests

The last tests performed on the uniform (0, 1) random variables verified that no correlation existed. This was done by testing that several levels of correlation lag were zero. Correlation lag is correlation between two series where one of the series has a lag with reference to the other. A correlation lag of *j* is defined as $p_j = C_j/C_0$, where $C_j = \mathrm{Cov}(X_i, X_{i+j})$, and a good estimator of correlation lag is the following:

$$\hat{p}_j = \frac{12}{n+1}\sum_{k=0}^{h}\left[U_{1+k_j}U_{1+(k+1)j} - 3\right] \quad \text{where } h = \lfloor (n-1)/j \rfloor - 1$$

Since previous tests showed that the U_i's can be assumed independent,

$$Var(\hat{p}_j) = \frac{13h+7}{(h+1)^2}$$

To test the null hypothesis that the correlation lag = 0, the test statistic $A_j = \hat{p}_j \big/ \sqrt{Var(\hat{p}_j)}$ was used which has an approximate standard normal distribution. The null hypothesis is rejected if $\left|A_j\right| > Z_{1-\alpha/2}$.

Correlations with lags of 1 through 10 were tested with the following test statistics calculated.

Table 31: Calculated Test Statistics for Lags of 1 to 10

Lag	Test Stat.
1	1.2379
2	0.8081
3	0.7359
4	0.0646
5	0.3884
6	0.6837
7	0.1725
8	0.5705
9	1.0417
10	0.2600

Using $\alpha = .95$, $Z_{1-\alpha/2} = 1.96$. All of the calculated test statistics are less than 1.96 so the null hypothesis fails to be rejected and it can be assumed that there is no discernable correlation among the U_i's.

APPENDIX B: SIMULATION MODEL VALIDATION

To help validate the throughput results obtained by the simulation model used in this research, various line setups were run that matched various ones in previous publications and the resulting throughputs compared to each other.

Closed Serial Production Lines

To check the validity of the simulation of closed production lines, results were compared to ones in Lui et al. (1992). Several setups using three, four, and seven workstations were examined as shown in Table 30, Table 33, and Table 34 respectively. In these tables, M is equal to the number of workstations in the line, μ represents the exponential rate of the service time of each workstation, b is the buffer capacity at each workstation (including space at the workstation), and c is the number of carriers in the line. The difference is calculated as *(the throughput from the paper – throughput from the simulation model) / the throughput from the paper.*

Table 32: Results of Closed Production Line Comparisons with 3 Workstations

Setup (M=3)	Throughput		Difference
	Paper	Simulation	
μ=(2,1,2), b=(6,2,4), c=7	0.928	0.928	0.02%
μ=(1,1,3), b=(2,6,2), c=6	0.812	0.812	0.02%
μ=(1,4,4), b=(3,6,6), c=9	0.997	0.997	0.03%
μ=(1,4,3), b=(6,2,2), c=6	0.984	0.984	-0.01%
μ=(4,1,1), b=(3,2,2), c=5	0.749	0.749	0.06%
μ=(1,4,1), b=(3,3,6), c=7	0.800	0.799	0.11%
μ=(1,1,2), b=(6,2,6), c=8	0.750	0.750	0.05%
μ=(3,4,3), b=(2,4,2), c=5	2.144	2.142	0.09%
μ=(1,1,1), b=(6,3,6), c=9	0.770	0.770	0.03%
μ=(1,4,2), b=(2,6,3), c=7	0.933	0.933	0.04%

Table 33: Results of Closed Production Line Comparisons with 4 Workstations

Setup (M=4)	Throughput		Difference
	Paper	Simulation	
μ=(1,2,2,1), b=(4,2,6,2), c=9	0.805	0.805	0.05%
μ=(3,4,4,1), b=(6,2,4,2), c=9	0.988	0.987	0.12%
μ=(1,4,3,2), b=(3,2,6,2), c=8	0.959	0.958	0.09%
μ=(1,1,1,3), b=(2,6,6,6), c=12	0.821	0.821	0.06%
μ=(2,4,2,1), b=(3,2,3,3), c=7	0.946	0.945	0.11%
μ=(3,4,4,1), b=(5,6,2,4), c=10	0.998	0.997	0.09%
μ=(3,2,2,3), b=(2,3,3,2), c=7	1.492	1.493	-0.03%
μ=(2,3,3,1), b=(6,4,6,6), c=13	0.996	0.995	0.13%
μ=(4,4,4,1), b=(6,4,6,6), c=13	1	0.999	0.13%
μ=(4,3,3,2), b=(6,2,6,6), c=12	1.919	1.917	0.08%

Table 34: Results of Closed Production Line Comparisons with 7 Workstations

Setup (M=7)	Throughput		Difference
	Paper	Simulation	
μ=(3,2,4,5,1,2,3) b=(2,2,2,2,2,2,2), c=9	0.925	0.925	0.05%
μ=(3,2,4,5,1,2,3) b=(2,2,2,2,2,2,2), c=11	0.918	0.918	0.01%

From these three tables, the largest difference is only 0.13% meaning the simulation model can be considered fairly accurate in determining the throughput of closed production lines of different lengths and setups.

Open Serial Production lines

Further validation was done by comparing various open production line setups. First, some more lines with exponentially distributed processing times (mean = 1) was examined by using Hillier et al.(1993) for comparison. These results are shown in Table 35. Again, b represents the buffer capacities at each workstation (not including space at the workstation). These results once again show very little difference between the two sources.

Table 35: Results of Open Production Line Comparisons from Hillier et al. (1993)

	Throughput		
Buffer Setup	**Paper**	**Simulation**	**Difference**
b = (1,1,1)	0.6312	0.6312	0.00%
b = (7,7,7)	0.8445	0.8444	0.01%
b = (7,8,9)	0.8580	0.8578	0.02%
b = (12,14,13)	0.9013	0.9012	0.01%
b = (1,1,1,1)	0.6076	0.6074	0.03%
b = (4,4,4,4)	0.7659	0.7662	-0.04%
b = (4,6,5,5)	0.7934	0.7936	-0.02%
b = (0,1,0)	0.5589	0.5589	0.00%
b = (5,6,5)	0.8154	0.8153	0.02%
b = (0,0,1,0)	0.5191	0.5189	0.04%
b = (4,4,5,4)	0.7736	0.7739	-0.04%

Next, some lines with lognormal processing times were examined. Table 36 shows the results of some comparisons made to balanced lines with lognormal(1,.25) distributed processing times and various numbers of buffers (b) from Powell (1994). The purpose of including these comparisons was to check the validity of the simulated lognormal distribution. The results show only a small difference between the two sources.

Table 36: Results of Open Production Line Comparisons from Powell (1994)

Buffer Setup	Throughput		Difference
	Paper	Simulation	
b1=0, b2=0	0.7202	0.7181	0.30%
b1=1, b2=0	0.7734	0.7715	0.25%
b1=1, b2=1	0.8423	0.8413	0.12%
b1=2, b2=1	0.8649	0.8640	0.11%
b1=2, b2=2	0.8911	0.8900	0.13%
b1=3, b2=2	0.9037	0.9023	0.15%

Finally, some automated lines were compared for validity. From Vouros and Papadoupolos (1998) several setups were investigated and these results are shown in Table 35 and Table 38. In these lines, exponential service rates (μ), repair rates (r), and time between failure rates (β) were used. The results show very little difference in results validating the unreliable simulation model used.

Table 37: Results of Automated Open Production Line Comparisons from Vouros and Papadoupolos (1998) with $\mu = (1,1,1,1)$, $r = (.5,.5,.5,.5)$, $\beta = (.05,.02,.01,.001)$

Buffer Setup	Throughput		Difference
	Paper	Simulation	
b = (0-1-0)	0.5250	0.5251	0.02%
b = (1-1-0)	0.5577	0.5578	0.03%
b = (1-1-1)	0.5916	0.5918	0.04%
b = (2-1-1)	0.6128	0.6130	0.04%
b = (2-1-2)	0.6307	0.6309	0.03%
b = (2-2-2)	0.6563	0.6565	0.04%

Table 38: Results of Automated Open Production Line Comparisons from Vouros and Papadoupolos (1998) with $\mu = (1.6,1.4,1.2,1)$, $r = (.5,.5,.5,.5)$, $\beta = (.05,.02,.01,.005)$

Buffer Setup	Throughput		Difference
	Paper	Simulation	
b = (0-1-0)	0.6476	0.6478	0.03%
b = (1-1-0)	0.6652	0.6654	0.03%
b = (1-1-1)	0.7289	0.7291	0.02%
b = (2-1-1)	0.7394	0.7396	0.02%
b = (1-3-1)	0.7670	0.7672	0.03%
b = (1-1-4)	0.8013	0.8016	0.03%

APPENDIX C: REAL WORLD DATA

In order to get an idea of what some realistic values are for MTBF and MTTR in a production environment, actual data was provided via personal communication by an automotive manufacturer. This information is presented in Table 39 below.

Table 39: Production Line Data for a Motor Compartment

Workstation	Speed(JPH)	Speed(JPM)	MTBF (min)	MTTR (min)
1	63	1.050	609.52	3
2	62	1.033	58.06	2
3	63	1.050	419.05	1.7
4	64	1.067	28.13	1.7
5	63	1.050	95.24	1.3
6	63	1.050	114.29	1.7
7	65	1.083	18.46	1.7
8	64	1.067	140.63	2
9	63	1.050	219.05	1.7
10	59	0.983	1016.95	0
11	63	1.050	95.24	1.3
12	63	1.050	47.62	2.3
13	60	1.000	420.00	5.3
14	64	1.067	93.75	2
15	63	1.050	609.52	3
16	73	1.217	24.66	2.3

APPENDIX D: OPTIMAL BUFFER ALLOCATIONS

In experiment 3, an even buffer allocation was compared to the optimal/near optimal buffer allocation of a given line setup. The actual buffer allocations used as the optimal allocation for the comparisons are shown in Tables 38 - 51. Each table gives the buffer allocation for lines with the same processing time distribution at various bottleneck levels. The buffer allocations are given for both four and eight-station production lines.

Four-Station Lines

Table 40: Optimal/Near Optimal Buffer Allocations for Exponentially Distributed Four-Station Reliable Production Lines at Various Bottleneck Levels

Exponential Distribution					
Bottleneck Mean	**Closed**	**Open: Position of Bottleneck**			
		1	**2**	**3**	**4**
1.50	1-1-1-1	2-1-0	1-2-0	0-2-1	0-1-2
1.75	1-1-0-2	2-1-0	1-2-0	0-2-1	0-1-2
2.00	1-1-0-2	2-1-0	1-2-0	0-2-1	0-1-2
2.25	2-0-0-2	2-1-0	1-2-0	0-2-1	0-1-2
2.50	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3
2.75	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3
3.00	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3

Table 41: Optimal/Near Optimal Buffer Allocations for Lognormal(1,.25) Distributed Four-Station Reliable Production Lines at Various Bottleneck Levels

Lognormal(1,.25) Distribution					
Bottleneck		Open: Position of Bottleneck			
Mean	Closed	1	2	3	4
1.50	1-1-0-2	2-1-0	1-2-0	0-2-1	0-1-2
1.75	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3
2.00	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3
2.25	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3
2.50	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3
2.75	2-0-0-2	3-0-0	1-2-0	0-1-2	0-0-3
3.00	2-0-0-2	3-0-0	1-2-0	0-1-2	0-0-3

Table 42: Optimal/Near Optimal Buffer Allocations for Lognormal(1,1) Distributed Four-Station Reliable Production Lines at Various Bottleneck Levels

Lognormal(1,1) Distribution					
Bottleneck		Open: Position of Bottleneck			
Mean	Closed	1	2	3	4
1.50	1-1-1-1	2-1-0	1-2-0	0-2-1	0-1-2
1.75	2-0-1-1	2-1-0	1-2-0	0-2-1	0-1-2
2.00	2-0-0-2	2-1-0	1-2-0	0-2-1	0-1-2
2.25	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3
2.50	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3
2.75	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3
3.00	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3

Table 43: Optimal/Near Optimal Buffer Allocations for Lognormal(1,4) Distributed Four-Station Reliable Production Lines at Various Bottleneck Levels

Logormal(1,4) Distribution					
Bottleneck		Open: Position of Bottleneck			
Mean	Closed	1	2	3	4
1.50	1-1-1-1	2-1-0	1-2-0	0-2-1	0-1-2
1.75	2-0-1-1	2-1-0	1-2-0	0-2-1	0-1-2
2.00	2-0-1-1	2-1-0	1-2-0	0-2-1	0-1-2
2.25	2-0-0-2	2-1-0	1-2-0	0-2-1	0-1-2
2.50	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3
2.75	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3
3.00	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3

Table 44: Optimal/Near Optimal Buffer Allocations for Lognormal(1,.25) with Constant CV Distributed Four-Station Reliable Production Lines at Various Bottleneck Levels

Lognormal(1,.25) - Const. CV Distribution					
Bottleneck Mean/Var.	**Closed**	**Open: Position of Bottleneck**			
		1	**2**	**3**	**4**
1.50/.5625	1-1-1-1	2-1-0	1-2-0	0-2-1	0-1-2
1.75/.765625	2-0-0-2	2-1-0	1-2-0	0-2-1	0-1-2
2.00/1.00	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3
2.25/1.265625	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3
2.50/1.5625	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3
2.75/1.890625	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3
3.00/2.25	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3

Table 45: Optimal/Near Optimal Buffer Allocations for Lognormal(1,1) with Constant CV Distributed Four-Station Reliable Production Lines at Various Bottleneck Levels

Lognormal(1,1) - Const. CV Distribution					
Bottleneck Mean/Var.	**Closed**	**Open: Position of Bottleneck**			
		1	**2**	**3**	**4**
1.50/2.25	1-1-1-1	2-1-0	1-2-0	0-2-1	0-1-2
1.75/3.0625	2-0-1-1	2-1-0	1-2-0	0-2-1	0-1-2
2.00/4.00	2-0-1-1	2-1-0	1-2-0	0-2-1	0-1-2
2.25/5.0625	2-0-0-2	2-1-0	1-2-0	0-2-1	0-1-2
2.50/6.25	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3
2.75/7.5625	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3
3.00/9.00	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3

Table 46: Optimal/Near Optimal Buffer Allocations for Lognormal(1,4) with Constant CV Distributed Four-Station Reliable Production Lines at Various Bottleneck Levels

Logormal(1,4) - Const. CV Distribution					
Bottleneck		**Open: Position of Bottleneck**			
Mean/Var.	**Closed**	**1**	**2**	**3**	**4**
1.50/9.00	1-1-1-1	1-1-1	1-2-0	0-2-1	1-1-1
1.75/12.25	1-1-1-1	2-1-0	1-2-0	0-2-1	0-1-2
2.00/16.00	2-0-1-1	2-1-0	1-2-0	0-2-1	0-1-2
2.25/20.25	2-0-1-1	2-1-0	1-2-0	0-2-1	0-1-2
2.50/25.00	2-0-0-2	2-1-0	1-2-0	0-2-1	0-1-2
2.75/30.25	2-0-0-2	2-1-0	1-2-0	0-2-1	0-1-2
3.00/36.00	2-0-0-2	3-0-0	1-2-0	0-2-1	0-0-3

8-Station Lines

Table 47: Optimal/Near Optimal Buffer Allocations for Exponentially Distributed Eight-Station Reliable Production Lines at Various Bottleneck Levels

Exponential Distribution					
Bottleneck		**Open: Postion of Bottleneck**			
Mean	**Closed**	**1**	**2**	**3**	**4**
1.50	1-1-1-1-1-1-1	2-1-1-1-1-1-0	1-2-1-1-1-1-0	1-1-2-1-1-1-0	1-1-1-2-1-1-0
1.75	1-1-1-0-1-1-2	2-1-1-1-1-1-0	1-2-1-1-1-1-0	1-1-2-1-1-1-0	0-1-2-2-1-1-0
2.00	2-1-0-1-1-0-1-2	2-1-1-1-1-1-0	2-2-1-1-1-0-0	1-2-2-1-1-0-0	0-1-2-2-1-1-0
2.25	2-1-0-1-1-0-1-2	3-1-1-1-1-0-0	2-2-1-1-1-0-0	1-2-2-1-1-0-0	0-1-2-2-1-1-0
2.50	2-1-1-0-1-0-1-2	4-1-1-1-0-0-0	2-3-1-0-1-0-0	1-2-2-1-1-0-0	0-1-2-3-1-0-0
2.75	2-1-0-0-1-0-1-3	4-1-1-1-0-0-0	3-3-1-0-0-0-0	0-3-3-1-0-0-0	0-1-2-3-1-0-0

Table 48: Optimal/Near Optimal Buffer Allocations for Lognormal(1,.25) Distributed Eight-Station Reliable Production Lines at Various Bottleneck Levels

Lognormal(1,.25) Distribution						
Bottleneck Mean	Closed	Open: Postion of Bottleneck			
		1	2	3	4
1.50	2-1-1-0-1-1-1-1	2-1-1-1-1-1-0	2-2-1-1-0-1-0	1-1-2-1-1-1-0	0-1-2-2-1-1-0
1.75	2-1-0-1-1-0-1-2	3-1-1-1-0-1-0	2-3-1-1-0-0-0	1-2-2-1-1-0-0	0-1-2-2-1-1-0
2.00	3-1-0-0-1-0-0-3	3-1-1-0-1-0-1	3-3-1-0-0-0-0	0-3-3-1-0-0-0	0-1-3-2-1-0-0
2.25	3-0-1-0-0-0-1-3	3-1-0-0-1-0-2	2-3-2-0-0-0-0	0-3-3-0-0-1-0	0-0-3-3-0-0-1
2.50	2-0-0-1-0-1-1-3	3-0-0-0-0-0-4	2-3-0-0-0-0-2	0-2-2-1-0-0-2	0-0-2-3-0-0-2
2.75	3-0-0-0-0-0-1-4	3-0-0-0-0-0-4	2-2-0-0-0-0-3	0-2-2-0-0-0-3	0-0-2-2-0-0-3

Table 49: Optimal/Near Optimal Buffer Allocations for Lognormal(1,1) Distributed Eight-Station Reliable Production Lines at Various Bottleneck Levels

Lognormal(1,1) Distribution						
Bottleneck Mean	Closed	Open: Postion of Bottleneck			
		1	2	3	4
1.50	1-1-1-1-1-1-1-1	1-1-1-1-1-1-1	1-1-1-1-1-1-1	1-1-2-1-1-1-0	1-1-1-1-1-1-1
1.75	1-1-1-1-1-1-1-1	2-1-1-1-1-1-0	1-2-1-1-1-1-0	1-1-2-1-1-1-0	0-1-2-2-1-1-0
2.00	2-1-0-1-1-0-1-2	2-1-1-1-1-1-0	2-2-1-1-0-1-0	0-2-2-1-1-1-0	0-1-2-2-1-1-0
2.25	2-1-0-1-0-1-1-2	3-1-1-1-1-0-0	2-3-1-0-1-0-0	0-3-2-1-1-0-0	0-1-2-2-1-1-0
2.50	2-1-0-1-0-1-1-2	4-1-1-0-1-0-0	2-3-1-1-0-0-0	0-3-3-1-0-0-0	0-0-3-3-1-0-0
2.75	3-0-1-0-0-1-0-3	5-1-1-0-0-0-0	3-3-1-0-0-0-0	0-3-3-1-0-0-0	0-0-3-3-1-0-0

Table 50: Optimal/Near Optimal Buffer Allocations for Lognormal(1,4) Distributed Eight-Station Reliable Production Lines at Various Bottleneck Levels

Logormal(1,4) Distribution						
Bottleneck Mean	Closed	Open: Postion of Bottleneck			
		1	2	3	4
1.50	1-1-1-1-1-1-1-1	1-1-1-1-1-1-1	1-1-1-1-1-1-1	1-1-2-1-1-1-0	0-1-2-2-1-1-0
1.75	1-1-1-1-1-1-1-1	2-1-1-1-1-1-0	1-2-1-1-1-1-0	0-2-2-1-1-1-0	0-1-2-2-1-1-0
2.00	1-1-1-1-1-1-1-1	2-1-1-1-1-1-0	1-2-1-1-1-1-0	0-2-2-1-1-1-0	0-1-2-2-1-1-0
2.25	2-1-1-0-1-1-1-1	2-1-1-1-1-1-0	1-2-1-1-1-1-0	0-2-2-1-1-1-0	0-1-2-2-1-1-0
2.50	2-1-0-1-1-0-1-2	2-1-1-1-1-1-0	1-2-1-1-1-1-0	0-2-2-1-1-1-0	0-1-2-2-1-1-0
2.75	2-1-1-0-1-0-1-2	3-1-1-1-1-0-0	2-2-1-1-1-0-0	0-2-3-1-1-0-0	0-1-2-3-1-0-0

Table 51: Optimal/Near Optimal Buffer Allocations for Lognormal(1,.25) with Constant CV Distributed Eight-Station Reliable Production Lines at Various Bottleneck Levels

Lognormal(1,.25) - Const. CV Distribution						
Bottleneck Mean/Var.	**Closed**	**Open: Postion of Bottleneck**				
		1	**2**	**3**	**4**
1.50/.5625	2-1-1-0-1-1-1-1	2-1-1-1-1-1-0	2-2-1-1-0-1-0	1-1-2-1-1-1-0	0-1-2-2-1-1-0
1.75/.765625	2-1-1-0-1-0-1-2	3-1-1-1-1-0-0	2-2-1-1-1-0-0	1-2-2-1-1-0-0	0-1-2-2-1-1-0
2.00/1.00	3-1-0-1-0-0-1-2	3-1-1-0-1-1-0	3-3-1-0-0-0-0	0-3-3-1-0-0-0	0-1-2-3-1-0-0
2.25/1.265625	3-0-1-0-0-0-1-3	4-1-0-0-1-0-1	3-3-0-1-0-0-0	0-3-3-1-0-0-0	0-0-3-3-1-0-0
2.50/1.5625	3-0-0-0-1-0-1-3	4-0-0-0-1-1-1	3-3-0-0-0-0-1	0-3-3-1-0-0-0	0-0-3-3-1-0-0
2.75/1.890625	3-0-0-1-0-0-1-3	3-1-1-0-0-0-2	3-3-0-0-0-1-0	0-3-3-1-0-0-0	0-0-3-3-1-0-0

Table 52: Optimal/Near Optimal Buffer Allocations for Lognormal(1,1) with Constant CV Distributed Eight-Station Reliable Production Lines at Various Bottleneck Levels

Lognormal(1,1) - Const. CV Distribution					
Bottleneck Mean/Var.	**Closed**	**Open: Postion of Bottleneck**			
		1	**2**	**3**	**4**
1.50/2.25	1-1-1-1-1-1-1-1	2-1-1-1-1-1-0	1-2-1-1-1-1-0	1-1-2-1-1-1-0	1-1-1-2-1-1-0
1.75/3.0625	2-1-1-0-1-1-1-1	2-1-1-1-1-1-0	1-2-1-1-1-1-0	1-1-2-1-1-1-0	0-1-2-2-1-1-0
2.00/4.00	2-1-1-0-1-0-1-2	2-1-1-1-1-1-0	2-2-1-1-1-0-0	0-2-2-1-1-1-0	0-1-2-2-1-1-0
2.25/5.0625	2-1-0-1-0-1-1-2	3-1-1-1-1-0-0	2-2-1-1-1-0-0	1-2-2-1-1-0-0	0-1-2-2-1-1-0
2.50/6.25	2-1-0-1-0-1-1-2	3-1-1-1-1-0-0	2-3-1-0-1-0-0	1-2-2-1-1-0-0	0-1-2-2-1-1-0
2.75/7.5625	2-1-1-0-1-0-1-2	4-1-1-1-0-0-0	2-3-1-1-0-0-0	0-3-3-1-0-0-0	0-1-2-3-1-0-0

Table 53: Optimal/Near Optimal Buffer Allocations for Lognormal(1,4) with Constant CV Distributed Eight-Station Reliable Production Lines at Various Bottleneck Levels

Logormal(1,4) - Const. CV Distribution						
Bottleneck Mean/Var.	**Closed**	**Open: Postion of Bottleneck**				
		1	**2**	**3**	**4**	
1.50/9.00	1-1-1-1-1-1-1-1	1-1-1-1-1-1-1	1-2-1-1-1-1-0	1-1-2-1-1-1-0	1-1-1-2-1-1-0	
1.75/12.25	1-1-1-1-1-1-1-1	2-1-1-1-1-1-0	1-2-1-1-1-1-0	0-2-2-1-1-1-0	0-1-2-2-1-1-0	
2.00/16.00	1-1-1-1-1-1-1-1	2-1-1-1-1-1-0	1-2-1-1-1-1-0	0-2-2-1-1-1-0	0-1-2-2-1-1-0	
2.25/20.25	1-1-1-1-0-1-1-2	2-1-1-1-1-1-0	1-2-1-1-1-1-0	0-2-2-1-1-1-0	0-1-2-2-1-1-0	
2.50/25.00	1-1-1-1-0-1-1-2	2-1-1-1-1-1-0	1-2-1-1-1-1-0	0-2-2-1-1-1-0	0-1-2-2-1-1-0	
2.75/30.25	2-1-1-0-1-0-1-2	3-1-1-1-1-0-0	2-2-1-1-1-0-0	0-2-2-1-1-1-0	0-1-2-2-1-1-0	

APPENDIX E: EXPERIMENT 3 DATA

This Appendix contains the data used to create the graphs used in the analysis of experiment 3. Tables 52 – 64 contain the values of the percentage of throughput lost due to using an even allocation instead of the optimal/near optimal solution (*1-(even allocation throughput/optimal allocation throughput)*).

Four-Station Lines

Table 54: Percentage of Throughput Lost by Using an Even Buffer Allocation when it is not Optimal for Four-Station Line with Bottleneck Mean = 1.50

	4 Stations, Bottleneck Mean = 1.50					
	Processing Time Distribution	**Closed**	**Open: Position of Bottleneck**			
			1	**2**	**3**	**4**
	exponential	0.0000%	0.4639%	0.2557%	0.2492%	0.4556%
const. Var.	lognormal(1,.25)	0.0413%	0.5201%	0.2733%	0.2746%	0.5241%
	lognormal(1,1)	0.0000%	0.0684%	0.2584%	0.2674%	0.0770%
	lognormal(1,4)	0.0000%	0.0511%	0.5490%	0.5602%	0.0550%
const. CV	lognormal(1,.25)	0.0000%	0.8156%	0.3506%	0.3552%	0.8229%
	lognormal(1,1)	0.0000%	0.3326%	0.2484%	0.2582%	0.3412%
	lognormal(1,4)	0.0000%	0.0000%	0.3753%	0.3857%	0.0000%

Table 55: Percentage of Throughput Lost by Using an Even Buffer Allocation when it is not Optimal for Four-Station Line with Bottleneck Mean = 1.75

	4 Stations, Bottleneck Mean = 1.75					
	Processing Time Distribution	Closed	Open: Position of Bottleneck 1	2	3	4
	exponential	0.0054%	1.7526%	0.8861%	0.8800%	1.7492%
const. Var.	lognormal(1,.25)	0.2347%	0.1887%	0.1512%	0.1505%	0.1897%
	lognormal(1,1)	0.1947%	1.2540%	0.8335%	0.8426%	1.2620%
	lognormal(1,4)	0.1193%	1.0175%	1.0057%	1.0180%	1.0247%
const. CV	lognormal(1,.25)	0.4014%	0.5152%	0.3389%	0.3389%	0.5162%
	lognormal(1,1)	0.0414%	1.4226%	0.8130%	0.8188%	1.4293%
	lognormal(1,4)	0.0000%	0.8045%	0.7550%	0.7660%	0.8081%

Table 56: Percentage of Throughput Lost by Using an Even Buffer Allocation when it is not Optimal for Four-Station Line with Bottleneck Mean = 2.00

	4 Stations, Bottleneck Mean = 2.00					
	Processing Time Distribution	Closed	Open: Position of Bottleneck 1	2	3	4
	exponential	0.4751%	2.2611%	1.1542%	1.1478%	2.2588%
const. Var.	lognormal(1,.25)	0.0988%	0.0568%	0.0512%	0.0508%	0.0570%
	lognormal(1,1)	0.7581%	1.3896%	0.9104%	0.9160%	1.3923%
	lognormal(1,4)	0.5621%	1.6431%	1.2656%	1.2771%	1.6505%
const. CV	lognormal(1,.25)	0.3262%	0.2474%	0.1882%	0.1878%	0.2477%
	lognormal(1,1)	0.4654%	1.7920%	1.0202%	1.0290%	1.7971%
	lognormal(1,4)	0.1806%	1.3628%	1.0032%	1.0131%	1.3670%

Table 57: Percentage of Throughput Lost by Using an Even Buffer Allocation when it is not Optimal for Four-Station Line with Bottleneck Mean = 2.25

	4 Stations, Bottleneck Mean = 2.25					
	Processing Time Distribution	Closed	Open: Position of Bottleneck 1	2	3	4
	exponential	1.1260%	2.3133%	1.2143%	1.2128%	2.3150%
const. Var.	lognormal(1,.25)	0.0515%	0.0189%	0.0176%	0.0178%	0.0185%
	lognormal(1,1)	1.0542%	1.1928%	0.7719%	0.7738%	1.1962%
	lognormal(1,4)	1.1287%	1.9444%	1.3656%	1.3766%	1.9493%
const. CV	lognormal(1,.25)	0.1846%	0.1150%	0.0969%	0.0965%	0.1148%
	lognormal(1,1)	1.0066%	1.7593%	1.0359%	1.0433%	1.7614%
	lognormal(1,4)	0.4414%	1.7204%	1.1542%	1.1634%	1.7214%

Table 58: Percentage of Throughput Lost by Using an Even Buffer Allocation when it is not Optimal for Four-Station Line with Bottleneck Mean = 2.50

	4 Stations, Bottleneck Mean = 2.50					
	Processing Time Distribution	**Closed**	**Open: Position of Bottleneck 1**	**2**	**3**	**4**
	exponential	1.4483%	2.2465%	1.1705%	1.1731%	2.2544%
const. Var.	lognormal(1,.25)	0.0150%	0.0070%	0.0063%	0.0068%	0.0068%
const. Var.	lognormal(1,1)	0.9700%	0.9269%	0.5935%	0.5946%	0.9301%
const. Var.	lognormal(1,4)	1.5564%	2.0191%	1.3593%	1.3684%	2.0256%
const. CV	lognormal(1,.25)	0.0995%	0.0560%	0.0498%	0.0501%	0.0558%
const. CV	lognormal(1,1)	1.2237%	1.6524%	0.9619%	0.9671%	1.6602%
const. CV	lognormal(1,4)	0.7260%	1.9268%	1.2354%	1.2448%	1.9243%

Table 59: Percentage of Throughput Lost by Using an Even Buffer Allocation when it is not Optimal for Four-Station Line with Bottleneck Mean = 2.75

	4 Stations, Bottleneck Mean = 2.75					
	Processing Time Distribution	**Closed**	**Open: Position of Bottleneck 1**	**2**	**3**	**4**
	exponential	1.5538%	2.1252%	1.0816%	1.0847%	2.1342%
const. Var.	lognormal(1,.25)	0.0071%	0.0030%	0.0025%	0.0041%	0.0025%
const. Var.	lognormal(1,1)	0.7846%	0.6678%	0.4421%	0.4397%	0.6713%
const. Var.	lognormal(1,4)	1.7472%	2.1476%	1.2903%	1.2970%	2.1526%
const. CV	lognormal(1,.25)	0.0542%	0.0289%	0.0261%	0.0267%	0.0289%
const. CV	lognormal(1,1)	1.2494%	1.4822%	0.8539%	0.8574%	1.4879%
const. CV	lognormal(1,4)	1.0645%	2.0238%	1.2690%	1.2790%	2.0231%

Table 60: Percentage of Throughput Lost by Using an Even Buffer Allocation when it is not Optimal for Four-Station Line with Bottleneck Mean = 3.00

	4 Stations, Bottleneck Mean = 3.00					
	Processing Time Distribution	**Closed**	**Open: Position of Bottleneck 1**	**2**	**3**	**4**
	exponential	1.5387%	1.9110%	0.9764%	0.9794%	1.9198%
const. Var.	lognormal(1,.25)	0.0042%	0.0015%	0.0012%	0.0030%	0.0012%
const. Var.	lognormal(1,1)	0.6035%	0.4746%	0.3258%	0.3231%	0.4783%
const. Var.	lognormal(1,4)	1.7826%	2.0855%	1.1896%	1.1937%	2.0908%
const. CV	lognormal(1,.25)	0.0309%	0.0156%	0.0141%	0.0147%	0.0156%
const. CV	lognormal(1,1)	1.1785%	1.2650%	0.7406%	0.7425%	1.2691%
const. CV	lognormal(1,4)	1.2956%	2.0858%	1.2700%	1.2803%	2.0916%

Eight-Station Lines

Table 61: Percentage of Throughput Lost by Using an Even Buffer Allocation when it is not Optimal for Eight-Station Line with Bottleneck Mean = 1.50

	8 Stations, Bottleneck Mean = 1.50					
	Processing Time Distribution	Closed	Open: Position of Bottleneck 1	2	3	4
	exponential	0.0000%	1.3358%	1.6520%	1.5361%	1.2998%
const. Var.	lognormal(1,.25)	0.3978%	0.7892%	0.9690%	0.6295%	1.0722%
	lognormal(1,1)	0.0000%	0.0000%	0.0000%	0.9431%	0.0000%
	lognormal(1,4)	0.0000%	1.0497%	0.3374%	0.0000%	0.4122%
const. CV	lognormal(1,.25)	0.5956%	1.3195%	1.4810%	0.9695%	1.6082%
	lognormal(1,1)	0.0000%	0.9850%	1.3830%	1.3628%	1.2023%
	lognormal(1,4)	0.0000%	1.0152%	1.1226%	0.0605%	-0.3769%

Table 62: Percentage of Throughput Lost by Using an Even Buffer Allocation when it is not Optimal for Eight-Station Line with Bottleneck Mean = 1.75

	8 Stations, Bottleneck Mean = 1.75					
	Processing Time Distribution	Closed	Open: Position of Bottleneck 1	2	3	4
	exponential	0.6590%	2.7934%	2.4641%	2.1244%	3.3007%
const. Var.	lognormal(1,.25)	0.3423%	0.1904%	0.3596%	0.3674%	0.3711%
	lognormal(1,1)	0.0000%	1.5581%	1.6757%	1.5959%	2.5050%
	lognormal(1,4)	0.0000%	2.5829%	2.1640%	0.8651%	0.6838%
const. CV	lognormal(1,.25)	0.9033%	0.6493%	0.9686%	0.9694%	0.9697%
	lognormal(1,1)	0.5419%	2.1685%	2.0898%	1.8907%	2.9004%
	lognormal(1,4)	0.0000%	2.8000%	2.0249%	0.8788%	0.6539%

Table 63: Percentage of Throughput Lost by Using an Even Buffer Allocation when it is not Optimal for Eight-Station Line with Bottleneck Mean = 2.00

	8 Stations, Bottleneck Mean = 2.00					
	Processing Time Distribution	Closed	Open: Position of Bottleneck			
			1	2	3	4
	exponential	2.1959%	3.2527%	3.4749%	3.2919%	3.7633%
const. Var.	lognormal(1,.25)	0.1058%	0.0458%	0.1010%	0.1172%	0.1152%
	lognormal(1,1)	1.4345%	1.7962%	2.0613%	2.3902%	2.6328%
	lognormal(1,4)	0.0000%	4.0506%	3.1011%	1.2491%	0.9152%
const. CV	lognormal(1,.25)	0.4577%	0.2615%	0.4697%	0.4847%	0.4701%
	lognormal(1,1)	1.6327%	2.5223%	2.6663%	2.8846%	3.2172%
	lognormal(1,4)	0.0000%	3.9841%	2.7698%	1.2022%	0.8381%

Table 64: Percentage of Throughput Lost by Using an Even Buffer Allocation when it is not Optimal for Eight-Station Line with Bottleneck Mean = 2.25

	8 Stations, Bottleneck Mean = 2.25					
	Processing Time Distribution	Closed	Open: Position of Bottleneck			
			1	2	3	4
	exponential	2.6613%	4.1271%	3.7980%	3.6117%	3.7361%
const. Var.	lognormal(1,.25)	0.0317%	0.0090%	0.0286%	0.0409%	0.0430%
	lognormal(1,1)	1.6062%	1.7637%	2.1891%	2.2495%	2.1459%
	lognormal(1,4)	0.6748%	5.1683%	3.7830%	1.5250%	1.0799%
const. CV	lognormal(1,.25)	0.2279%	0.1165%	0.2159%	0.2237%	0.2254%
	lognormal(1,1)	2.0587%	3.0859%	2.9308%	2.9025%	3.1012%
	lognormal(1,4)	0.5999%	4.8957%	3.3379%	1.4493%	0.9775%

Table 65: Percentage of Throughput Lost by Using an Even Buffer Allocation when it is not Optimal for Eight-Station Line with Bottleneck Mean = 2.50

	8 Stations, Bottleneck Mean = 2.50					
	Processing Time Distribution	Closed	Open: Position of Bottleneck			
			1	2	3	4
	exponential	2.7862%	3.9475%	4.0546%	3.4674%	3.8417%
const. Var.	lognormal(1,.25)	0.0085%	0.0000%	0.0060%	0.0180%	0.0202%
	lognormal(1,1)	1.3502%	1.2762%	1.7053%	1.8857%	1.9016%
	lognormal(1,4)	1.6649%	5.7872%	4.1517%	1.6695%	1.1601%
const. CV	lognormal(1,.25)	0.1142%	0.0562%	0.1035%	0.1088%	0.1105%
	lognormal(1,1)	2.1136%	2.6701%	3.0636%	2.7148%	2.7712%
	lognormal(1,4)	0.7966%	5.5368%	3.7381%	1.6226%	1.0771%

Table 66: Percentage of Throughput Lost by Using an Even Buffer Allocation when it is not Optimal for Eight-Station Line with Bottleneck Mean = 2.75

	8 Stations, Bottleneck Mean = 2.75					
	Processing Time Distribution	**Closed**	**Open: Position of Bottleneck**			
			1	**2**	**3**	**4**
	exponential	2.7531%	3.3935%	3.6377%	3.8000%	3.4799%
const. Var.	lognormal(1,.25)	0.0036%	0.0000%	0.0000%	0.0093%	0.0118%
	lognormal(1,1)	1.1998%	0.8627%	1.2637%	1.3681%	1.3695%
	lognormal(1,4)	1.9377%	6.6039%	4.7539%	2.3213%	1.6504%
const. CV	lognormal(1,.25)	0.0610%	0.0289%	0.0514%	0.0553%	0.0569%
	lognormal(1,1)	1.9728%	2.3309%	2.7328%	2.8931%	2.7008%
	lognormal(1,4)	1.6479%	6.6890%	4.2990%	1.7337%	1.1392%

APPENDIX F: EXPERIMENT 4 LINE COMPARISONS

In experiment 4, unreliable lines were compared to equivalent reliable lines (same t_e and CV). The data for these comparisons can be found in this appendix. Table 65 presents the data for balanced closed production lines with realistic unreliable values for MTTR and MTBF. Table 68 and Table 69 present the data for closed production lines with various bottleneck levels and workstation CVs of 1 and 0.5 respectively.

Table 67: Reliable and Unreliable Line Comparisons of Throughput for the Best, Second Best, and Worst Buffer Allocations in Various Balanced Four-Station Production Lines

CV	Setup	Optimal Allocation	Optimal TH	2^{nd} Best Allocation	2^{nd} Best TH	Worst Allocation	Worst TH
0.196	MTTR = 1, MTBF = 50	(1-1-1-1)	0.9482	(2-0-1-1)	0.9446	(0-0-4-0)	0.9275
	log(1.02,.04)	(1-1-1-1)	0.9316	(0-1-1-2)	0.8824	(4-0-0-0)	0.8283
0.140	MTTR = 1, MTBF = 100	(1-1-1-1)	0.9731	(0-1-1-2)	0.9712	(4-0-0-0)	0.9618
	log(1.01,.02)	(1-1-1-1)	0.9626	(0-1-1-2)	0.9178	(4-0-0-0)	0.8760
0.089	MTTR = 1, MTBF = 250	(1-1-1-1)	0.9890	(1-2-0-1)	0.9882	(4-0-0-0)	0.9842
	log(1.004,.008)	(1-1-1-1)	0.9838	(0-1-1-2)	0.9485	(4-0-0-0)	0.9204
0.063	MTTR = 1, MTBF = 500	(1-1-1-1)	0.9944	(0-1-1-2)	0.9940	(0-4-0-0)	0.9920
	log(1.002,.004)	(1-1-1-1)	0.9916	(0-1-1-2)	0.9637	(4-0-0-0)	0.9433
0.909	MTTR = 5, MTBF = 50	(1-1-1-1)	0.7382	(2-0-2-0)	0.7367	(0-4-0-0)	0.7190
	log(1.1,1)	(1-1-1-1)	0.5892	(0-2-1-1)	0.5663	(4-0-0-0)	0.5022
0.673	MTTR = 5, MTBF = 100	(1-1-1-1)	0.8470	(0-2-0-2)	0.8461	(4-0-0-0)	0.8343
	log(1.05,.5)	(1-1-1-1)	0.6933	(0-2-1-1)	0.6614	(4-0-0-0)	0.5866
0.438	MTTR = 5, MTBF = 250	(1-1-1-1)	0.9320	(2-0-2-0)	0.9316	(0-0-0-4)	0.9259
	log(1.02,.2)	(1-1-1-1)	0.8162	(0-2-1-1)	0.7721	(4-0-0-0)	0.6934
0.313	MTTR = 5, MTBF = 500	(1-1-1-1)	0.9644	(0-2-0-2)	0.9642	(0-4-0-0)	0.9611
	log(1.01,.1)	(1-1-1-1)	0.8862	(0-2-1-1)	0.8364	(4-0-0-0)	0.7642
1.667	MTTR = 10, MTBF = 50	(1-1-1-1)	0.5747	(2-0-2-0)	0.5741	(0-0-4-0)	0.5613
	log(1.2,4)	(1-1-1-1)	0.4220	(0-2-1-1)	0.4114	(4-0-0-0)	0.3705
1.286	MTTR = 10, MTBF = 100	(1-1-1-1)	0.7261	(0-2-0-2)	0.7257	(4-0-0-0)	0.7157
	log(1.1,2)	(1-1-1-1)	0.5107	(0-2-1-1)	0.4951	(4-0-0-0)	0.4421
0.860	MTTR = 10, MTBF = 250	(1-1-1-1)	0.8678	(2-0-2-0)	0.8676	(0-0-0-4)	0.8620
	log(1.04,.8)	(1-1-1-1)	0.6373	(0-2-1-1)	0.6117	(4-0-0-0)	0.5422
0.620	MTTR = 10, MTBF = 500	(1-1-1-1)	0.9285	(0-2-0-2)	0.9283	(0-4-0-0)	0.9252
	log(1.02,.4)	(1-1-1-1)	0.7349	(0-2-1-1)	0.6997	(4-0-0-0)	0.6214

Table 68: Reliable and Unreliable Line Comparisons of Throughput for the Optimal, Second Best, and Worst Buffer Allocations in Various Four-Station Production Lines with Workstation CV = 1

Setup		Optimal	Optimal	2^{nd} Best	2^{nd} Best	Worst	Worst
te	Line Type	Allocation	TH	Allocation	TH	Allocation	TH
1.01	unreliable	(0-2-0-2)	0.9586	(2-0-2-0)	0.9586	(0-0-4-0)	0.9582
	reliable	(1-1-1-1)	0.6165	(2-0-1-1)	0.5940	(4-0-0-0)	0.5275
1.05	unreliable	(2-0-0-2)	0.9246	(1-1-0-2)	0.9245	(0-4-0-0)	0.9229
	reliable	(1-1-1-1)	0.6103	(2-0-1-1)	0.5897	(0-4-0-0)	0.5216
1.10	unreliable	(2-0-0-2)	0.8837	(1-1-0-2)	0.8836	(0-4-0-0)	0.8824
	reliable	(1-1-1-1)	0.6022	(2-0-1-1)	0.5840	(0-4-0-0)	0.5141
1.25	unreliable	(2-0-0-2)	0.7811	(1-1-0-2)	0.7810	(0-4-0-0)	0.7796
	reliable	(1-1-1-1)	0.5766	(2-0-1-1)	0.5647	(0-4-0-0)	0.4920
1.50	unreliable	(2-0-0-2)	0.6535	(1-1-0-2)	0.6533	(0-0-4-0)	0.6528
	reliable	(1-1-1-1)	0.5317	(2-0-1-1)	0.5274	(0-4-0-0)	0.4561
1.75	unreliable	(2-0-0-2)	0.5620	(1-1-0-2)	0.5619	(0-0-4-0)	0.5611
	reliable	(1-1-1-1)	0.4874	(2-0-1-1)	0.4874	(0-4-0-0)	0.4226
2.00	unreliable	(2-0-0-2)	0.4930	(1-1-0-2)	0.4930	(0-0-4-0)	0.4923
	reliable	(2-0-1-1)	0.4481	(1-1-0-2)	0.4481	(0-4-0-0)	0.3918

Table 69: Reliable and Unreliable Line Comparisons of Throughput for the Optimal, Second Best, and Worst Buffer Allocations in Various Four-Station Production Lines with Workstation CV = 0.5

Setup		Optimal	Optimal	2^{nd} Best	2^{nd} Best	Worst	Worst
te	Line Type	Allocation	TH	Allocation	TH	Allocation	TH
1.01	unreliable	(1-1-1-1)	0.9622	(0-2-0-2)	0.9622	(0-0-0-4)	0.9608
	reliable	(1-1-1-1)	0.7944	(2-0-1-1)	0.7532	(0-4-0-0)	0.6727
1.05	unreliable	(2-0-0-2)	0.9289	(1-1-0-2)	0.9285	(0-0-4-0)	0.9255
	reliable	(1-1-1-1)	0.7859	(2-0-1-1)	0.7495	(0-4-0-0)	0.6647
1.10	unreliable	(2-0-0-2)	0.8885	(1-1-0-2)	0.8880	(0-0-4-0)	0.8849
	reliable	(1-1-1-1)	0.7739	(2-0-1-1)	0.7437	(0-4-0-0)	0.6545
1.25	unreliable	(2-0-0-2)	0.7846	(1-1-0-2)	0.7841	(0-0-4-0)	0.7814
	reliable	(1-1-1-1)	0.7299	(2-0-1-1)	0.7162	(0-4-0-0)	0.6224
1.50	unreliable	(2-0-0-2)	0.6565	(1-1-0-2)	0.6561	(0-0-4-0)	0.6540
	reliable	(1-1-1-1)	0.6441	(1-1-0-2)	0.6438	(0-4-0-0)	0.5673
1.75	unreliable	(2-0-0-2)	0.5642	(1-1-0-2)	0.5639	(0-4-0-0)	0.5623
	reliable	(2-0-0-2)	0.5661	(2-0-1-1)	0.5653	(0-4-0-0)	0.5143
2.00	unreliable	(2-0-0-2)	0.4948	(1-1-0-2)	0.4945	(0-0-4-0)	0.4932
	reliable	(2-0-0-2)	0.4990	(2-0-1-1)	0.4981	(0-4-0-0)	0.4663