University of Kentucky
UKnowledge

Theses and Dissertations--Manufacturing Systems Engineering

Manufacturing Systems Engineering

2012

A STUDY OF QUEUING THEORY IN LOW TO HIGH REWORK ENVIRONMENTS WITH PROCESS AVAILABILITY

Adam J. Brown
University of Kentucky, ajbrow4@g.uky.edu

Recommended Citation

Brown, Adam J., "A STUDY OF QUEUING THEORY IN LOW TO HIGH REWORK ENVIRONMENTS WITH PROCESS AVAILABILITY" (2012). *Theses and Dissertations--Manufacturing Systems Engineering*. 2.
https://uknowledge.uky.edu/ms_etds/2

Adam J. Brown, Student

Dr. Fazleena Badurdeen, Major Professor

Dr. Dusan Sekulic, Director of Graduate Studies

A STUDY OF QUEUING THEORY IN LOW TO HIGH REWORK ENVIRONMENTS WITH PROCESS AVAILABILITY

THESIS

A thesis submitted in partial fulfillment of the
requirements for the degree of Master of Science in
Manufacturing Systems Engineering in the College of Engineering
at the University of Kentucky

By

Adam Jerome Brown

Lexington, Kentucky

Director: Fazleena Badurdeen, Ph.D., Associate Professor

Lexington, Kentucky

ABSTRACT OF THESIS

A STUDY OF QUEING THOERY IN LOW TO HIGH REWORK ENVIRONMENTS WITH PROCESS AVAILABILITY

In manufacturing systems subject to machine and operator resource constraints the effects of rework can be profound. High levels of rework burden the resources unnecessarily and as the utilization of these resources increases the expected queuing time of work in process increases exponentially. Queuing models can help managers to understand and control the effects of rework, but often this tool is overlooked in part because of concerns over accuracy in complex environments and/or the need for limiting assumptions. One aim of this work is to increase understanding of system variables on the accuracy of simple queuing models. A queuing model is proposed that combines G/G/1 modeling techniques for rework with effective processing time techniques for machine availability and the accuracy of this model is tested under varying levels of rework, external arrival variability, and machine availability. Results show that the model performs best under exponential arrival patterns and can perform well even under high rework conditions. Generalizations are made with regards to the use of this tool for allocation of jobs to specific workers and/or machines based on known rework rates with the ultimate aim of queue time minimization.

KEYWORDS: Queuing, Simulation, Optimization, Rework, Immediate Feedback

Adam Jerome Brown

December 7, 2012

A STUDY OF QUEING THOERY IN LOW TO HIGH REWORK ENVIRONMENTS
WITH PROCESS AVAILABILITY

By

Adam Jerome Brown

Fazleena Badurdeen, Ph.D.
Director of Thesis

Dusan Sekulic, Ph.D.
Director of Graduate Studies

December 7, 2012
Date

For My Parents

ACKNOWLEDGEMENTS

I would like to start off these acknowledgements by thanking my advisor Dr. Badurdeen for her prompt feedback, healthy criticism, encouragement, and patience during the lengthy research process. I have gained invaluable experience since beginning work with Dr. Badurdeen in 2010 and overall I want to express my thanks to her for the extension of this excellent opportunity.

I also want to thank my colleagues for creating a positive environment in which to work. Without them I may never have reached this point. They have helped me to maintain my sanity, my grades, and my motivation. Specifically I thank those with whom I have had the privilege to work personally. The list is too long to express eloquently while giving each person his or her due. Thanks to you all. I cannot go without acknowledging friends and family away from work who have supported me in various ways. Thanks to Mom and Dad for backing me up even when I had no real plans. I must thank one friend specifically, Isaac Lee, for being interested in my research and willing to listen to my complaints about simulation, work, and other things.

I thank my committee members, Dr. Holloway and Dr. Walcott for their agreement to listen to my defense and for their constructive comments and feedback. I also thank Dr. Ashley who provided inspiration and direction in the early stages of the ideas presented here. At last, I want to thank any future readers of this document. Thank you for uncovering this work and I hope that in it you can find the information which you seek.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1 INTRODUCTION

In many cases an effective queuing model can provide accurate estimates for steady-state queuing times (Whitt 1983). Knowledge of these queuing relationships for systems with immediate-feedback rework can be used in resource and skill management to facilitate effective queue time reduction, thereby improving the ratio of value-added to non-value added production time (de Treville and van Ackere 2006). Using conventional symbols from queuing theory, immediate feedback rework is depicted in Figure 1.1. Methods to obtain exact solutions for expected queue times are analytically intractable without the use of limiting assumptions such as the requirement for exponential service times and for stationary arrival distributions (Jackman and Johnson 1993). Because of perceived limitations arising from these requirements, the use of queuing models for performance analysis has been outweighed by discrete event simulation (DES). When compared to the number of DES case studies, the use of queuing models is extremely limited especially for the analysis of complex systems with rework.

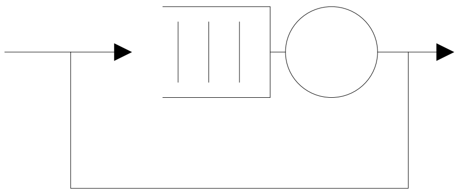


Figure 1.1: Single stage queuing diagram with immediate feedback rework

When conducted properly, DES modeling provides the ability to replicate high levels of system detail, but this ability comes with a cost of added complexity as shown in the modeling spectrum of Figure 1.2. To counter this argument, queuing researchers suggest an approximate analysis of realistic systems as an alternative to exact analysis of over-simplified systems (Whitt 1980; Kim et al 2005). In the literature regarding approximate analysis of queuing networks, a simple approach to handling feedback in the system is proposed and tested against simulation models with a fair amount of success (Takacs 1962; Keuhn 1979). Nonetheless, the technique is inexact for queues with non-exponential arrival distributions, and the full behavior of the accuracy of the calculated average queue time is not demonstrated for the full range of rework rates and varying arrival distributions.

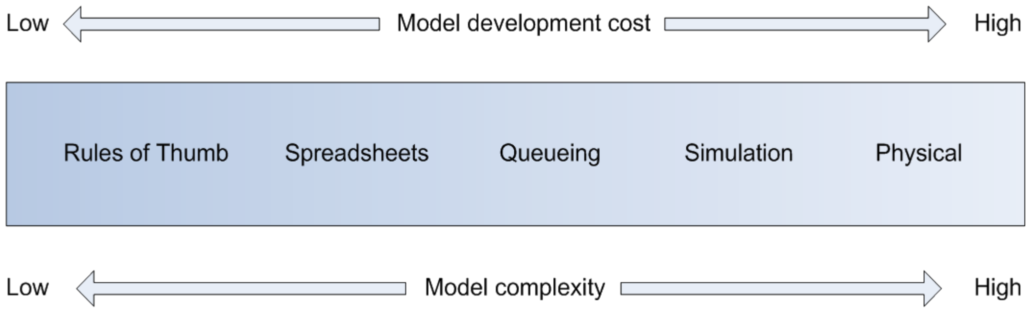


Figure 1.2: (Jackman and Johnson 1993), Spectrum of manufacturing systems modeling techniques

The purpose of this work is to provide a detailed specification of a modeling technique that captures the effects of rework on the important metric of queuing time, specifically when the rework process utilizes the same resources as the original job (immediate feedback rework). The full range of applicability is demonstrated for the method, which acts as a tool for lead time reduction. Although it may not be clear what combination of factors leads to a specific rework rate at a given workstation, certainly this rate is tied to worker skills, and if the rework rate can be monitored with any certainty, the effects of rework rate on the ever-important lead time can be examined. Designing policy for lead time reduction necessitates the examination of system variability. Reducing variability not only cuts lead time but allows more accurate prediction of lead time, which in turn improves customer satisfaction. Shorter lead times mean quicker response to the customer, less inventory in the system, and therefore less holding cost (Suri 1998). Most importantly lead time reduction eliminates non-value added waiting time.

In order to demonstrate the potential seriousness of the effects of rework, it may be useful to review the Lean manufacturing principle of waste reduction. Practitioners of Lean often refer to the seven deadly wastes: transportation, inventory, motion, waiting, overproduction, over processing, and defects (Womack and Jones 2003). Of the seven wastes, at least three (defects, waiting, and inventory) can be tied directly to rework activity. The value stream map (VSM) is a tool used for Lean implementation to capture the presence and location of these wasteful, non-value added activities such as queuing. Unfortunately, the VSM does not give a full depiction of the dynamic nature of the production line. Inventory is simply counted between operations at the time of study and divided by the average daily demand to obtain the approximate number of days' worth of inventory on hand (Rother and Shook 1999). For example, if 500 parts are on hand in the queue before a process and the process completes 50 parts per day, then the existing inventory could last for 10 days, and the last part in line would actually wait 10 days before being completed. This calculated time provides a decent portrayal of waiting typically seen in the system, but still it is a static reflection based on system status and the value could be drastically different from one day to the next. For example, consider the inventory at a station subject to a lot of tacit, manual work and how queuing time could change based on operator experience. Of course the Lean solution here would be to improve standard work, remove tacit knowledge requirements, and build quality into the system. Queuing analysis is no replacement for such efforts, but rather a tool to give these efforts better direction. Understanding queuing effects can allow management to direct limited resources while estimating the benefits of making improvements in certain areas.

In many cases the level of rework and the expected queue time links back to operator skill. It has been shown in the field of organizational behavior that job performance is linked directly to worker motivation, skill, and technical support (Mitchell 1982). Though important in any case, the benefits of skill flexibility are especially emphasized in cellular manufacturing environments where workers act as team communities on the shop floor. One key element of these cells is the cross-training of the employees to perform multiple tasks within the cell. This environment facilitates mutual problem solving, and increases resilience to sudden changes that may occur such as absenteeism or demand surge (Slomp et al. 2005). Nevertheless, for every skill that a worker adds, there is a sacrifice in specialization, there is a training cost, and overall there is a complication added in the need to manage the use of these varying skills. Since not every worker is equally trained, allocation of workers to cells and to stations within cells can have significant impact on rework rate and queuing (Kuo and Yang 2007). Queuing models help the management of these heterogeneous skills.

Consider the following scenario presented by Hopp and Spearman in Factory Physics (2001). In the example, two machines are considered conducting a similar process. The first machine requires no setups between jobs but has longer expected processing times than the second. Equation 1.1 demonstrates the contribution of setups to “effective processing time” t_e which is equal to the sum of the normally observed time t_0 and the expected setup time per part. Here t_s is the average setup time and N_s is the expected number of parts between setups.

$$t_e = t_0 + \frac{t_s}{N_s} \qquad \text{Equation 1.1}$$

Because of the tradeoff with machine 2 having setups and shorter processing and machine 1 having longer processing but no setups, the effective capacity in terms of parts per hour or production is equivalent for both machines. Now consider the standard deviation of processing time for each machine according to equation 1.2. Here the standard deviation is σ_e is the effective standard deviation with adjustments for setups. Furthermore, σ_0 is the naturally observed processing time standard deviation, and σ_s is the standard deviation of setup time.

$$\sigma_e^2 = \sigma_0^2 + \frac{\sigma_s^2}{N_s} + \frac{N_s - 1}{N_s^2} t_s^2 \qquad \text{Equation 1.2}$$

In the example, machine 2 has less variance in processing times but actually has the greater "effective variance" when the effect of setups is considered. Of course the answer to which machine is better (less variable) depends on the specific values for number of jobs between setups, average setup time, processing variability, etc. The example does not even incorporate rework rate, which if altered by changing any number of outside factors could alter the selection of best machine. This example shows how explicit knowledge of different forms of variability can reveal leverage points in a system. For example, questions about which setups should be reduced and where reduction of rework could have the biggest payoff can be answered. These answers would not be obvious through simple observation of inventory levels between processes. Managers may not fully comprehend the dynamic nature of the system and as a result take actions, such as increasing utilization, that act to increase lead times (Suri 1998). In an ideal situation, queuing models could be used in conjunction with an optimization technique to provide suggestions for the best allocation of heterogeneously-skilled workers. Before this can be done the queuing models and their accuracy and limitations must be fully understood.

The research question to be addressed in this work can be divided into five parts:

- What are the advantages of using queuing theory to model system performance?
- Can the proposed queuing model accurately represent the average waiting time per part in a system with rework?
- In what circumstances, if any, are more-refined analytic models required to improve accuracy?
- What relationships can be observed between system parameters and the accuracy of the proposed method?
- What generalizations can be made regarding the use of queuing estimates in optimization?

The remainder of this thesis will be presented as follows. A three part literature review begins with the discussion of analytical models considering workers with mixed skill levels, continues with a focus on simulation models for similar scenarios, and finally examines the use of queuing approximations to study complex systems where such factors as human skills may be present. Chapter 3 discusses development of the queuing model and associated assumptions. Chapter 4 demonstrates the observed relationships of the developed model, while Chapter 5 examines the cause of an observed discrepancy between the queuing calculations and initial simulation observations. Chapter 6 presents results on the accuracy of the queuing model as compared to a refined version of the simulation output. Finally, Chapter 7 concludes with a recap of the above research questions and brief discussion of directions that may be taken in future work.

2 LITERATURE REVIEW

Queuing time is related to rework which is in turn related to human resource management. Human resources can be considered to be homogeneous or heterogeneous with respect to human and technical skills. Heterogeneous workers are by nature skill-flexible. Realistically, any given set of workers must be heterogeneous to some extent, and there may be any number of ways to represent this fact. Some ways to represent skill-flexibility are by the number of skills each worker possesses, the overlapping of skill from worker to worker, and the degree to which each worker possesses the same number of skills (Yue et al. 2011). Most methods used to answer the question of how to manage a heterogeneous workforce involve some form of analytic modeling, discrete event simulation (DES), or hybrid simulation-analytic modeling. Analytic models are most suited for mathematical optimization (deciding the best allocation of workers), whereas DES models are more detailed but are typically limited to analysis through statistical examination of experimental scenarios. Hybrid models typically use analytic optimization to direct the search for optimal or near-optimal simulation scenarios, specifying parameters to improve a given performance measure. Queuing theory provides a unique type of analytic model in that it incorporates some of the stochastic nature of the production system by providing expected steady-state values, but is less complex than DES modeling.

This literature review attempts to outline the advantages of using queuing theory to model system performance. It focuses on three modeling techniques. Sections 2.1 and 2.2 examine deterministic analytic models and discrete event simulation, respectively. In 2.3 queuing theory is introduced in general and some light is shed on the potential pros and cons of its implementation, especially with regard to modeling of manufacturing systems with heterogeneous workers.

One of the resounding themes of this research is the importance of worker skills. Cross-training and the utilization of worker flexibility is one of the primary motivations behind cellular manufacturing, and its importance is also realized in dual resource constrained (DRC) job shop environments. For this reason, much of the literature on heterogeneous workers and their allocation to various tasks is presented in the context of cellular and DRC systems.

2.1 Analytic Models

One of the greatest advantages of the analytic approach is the applicability to optimization. Much of the work presented in this section can be seen as some variation on the classic worker assignment optimization problem well-known to the operations research field. In the classic problem some *n* tasks are assigned to *n* workers and there is an associated cost with each possible assignment. The goal of the optimization is to minimize the total cost of assigning workers to tasks. In the adaptations presented here, skill affects the assignment decision.

One adaptation of the classic version is the ‘assignment problem recognizing agent qualification’ for which not every worker is capable of doing every task (Pentico 2005). That is, not every worker-task assignment is feasible. This variant relates to skill distribution in the work force, as some operators may not be trained for certain jobs. One method of capturing this kind of worker flexibility in an analytical model is to assign a parameter to each worker that specifies whether or not he or she can be selected to perform a given job (Kuo and Yang 2007). In an extension to this idea, some models institute a measure of worker effectiveness or efficiency. In this way analytic models can represent workers with varying capability at performing any given job or set of jobs that exist in some manufacturing cell. Tiwari et al. (2009) presents an interesting model considering worker effectiveness based on a service organization where various skills are required at multiple stages in the service provision. Due to a constraint on the number of employees with each required skill, the optimization model must decide whether or not a less skilled worker should be assigned to a task in order to meet time objectives. Depending on the expertise of the worker assigned at a given stage, there may be a need for the more experienced workers to follow up with an enhancement activity. Although this model involves rework, it differs from the research of this thesis in that the enhancement effort (rework) is done at separate stations rather than at the original process with the same worker. Slomp et al. (2005) present an integer programming model to study the effectiveness of workers subject to various skill chaining patterns. This model does not consider varying effectiveness between workers at a given task, but rather decides for which machines each worker should be trained in relation to the others in the work cell so as to ensure a balanced work load.

Both technical and human skills such as communication and problem solving can be shown to have some effect on the cost of worker assignment (Norman et al. 2002). Furthermore, some studies discuss the importance of learning effects, where the skills of each worker can be enhanced over time subject to some training cost (Slomp et al. 2005). When speaking about human skills, it makes logical sense to include some learning ability, especially if the model result is expected to hold over changing conditions. The model presented by Norman et al. (2002) incorporates productivity, quality cost, and training cost into the objective function which attempts to maximize profit in assigning workers to cells. There is an associated productivity with each skill level as well as a quality cost which accounts for any rework or scrap that might occur. If advantageous, the model acts to increase worker skills at some predetermined cost.

Because of the dynamic nature of cells and DRC job shops where workers may shift jobs regularly, it is important to take careful consideration of the way in which productivity is captured in a deterministic model. Slomp et al. (2005) model the operating cost of a cell based on the workload of the bottleneck worker. This method is reasonable considering that the bottleneck of the cell would determine the length of time necessary to complete an order. A similar outlook is taken by Kuo and Yang (2007) in a model set to minimize multiplication of skill levels. Niemi (2009) discuses the optimal assignment of workers in make-to-order assembly cells considering congestion loss which accounts for difficulty encountered when trying to divide a single task among multiple workers. The optimization is a makespan minimization where the processing times and congestion loss measures are deterministic observed values. Huq et al. (2003) likewise show a makespan minimization model with deterministic processing times known according to the number of workers at a given station. This model also considers the effects of lot sizing on required setup times. A common shortcoming of analytic modeling of manufacturing systems is the reliance upon deterministic processing times. These models could be used for a kind of rough cut capacity analysis, such as to answer questions like how many machines will be needed to do a certain task in a certain time, but as soon as more detailed operating policies are included it becomes necessary, or at least highly compelling to resort to DES modeling.

2.2 Discrete Event Simulation

The importance of worker skill considerations on staffing production cells has been demonstrated through the use of DES modeling (Juran and Schruben 2004). DES models are useful for quantifying the effectiveness of certain operating policies for systems with flexible workers. Since cellular and job-shop type environments allow the workers some freedom to switch tasks, two common questions that arise in this context are when a worker should be permitted to change jobs and to which job he or she should move (Bobrowski and Park 1993). Two common “when” rules are known as centralization and decentralization. With centralization, workers can move after finishing each job. With decentralization, the workers should move only when the queue in front of their current processes are empty. “Where” rules might be always to move to the workstation with the longest queue or to the workstation at which the worker is most efficient. Operating under these flexible operating policies makes it difficult to predict what job each worker will be performing at a given time, especially when arrival and processing times are highly variable. By running several replications of experimental combinations of the “when” and “where” rules under different levels of variability, the importance of these factors on the flow time of each job can be determined (Bobrowski and Park 1993). These effects would be difficult to discern using an analytic model.

In addition to the advantage DES models have for representing detailed operating policy, they are also ideal for study of short-term transient effects that may not be discernible with analytic models. Stratman et al (2004) study the impact of temporary and permanent workers on manufacturing cost. The added cost of rework and variable processing time associated with the less experienced workers is somewhat offset by the reduced cost of labor. Also the effect of lot size and frequency of changeover is studied as it pertains to the workers with different learning rates. Short term impacts in this lot-to-lot example are ideal for simulation study. Furthermore, McCreery et al (2004) show that the complexity of work is important to the benefit of cross training and worker flexibility. As complexity increases, forgetting effects become significant and cross training is less valuable. In this flexible environment, workers often switch tasks and it may be difficult to model utilization without resorting to simulation.

One weakness of DES modeling is somewhat unquantifiable but has to do with stakeholder engagement. In their review of simulation practices from 1997-2006, Jahangirian et al (2010) show that although DES is the most popular simulation technique, the extensive data collection phase that is required has the tendency to divert the interest of the user. Stakeholder engagement is partially represented by the percentage of reviewed papers which involve the use of real data. Lower than expected levels of engagement may be attributed to the inability of DES models to directly reflect qualitative descriptions as is possible with other simulation techniques such as system dynamics. Similarly, de Treville and van Ackere (2006) study the use of DES and queuing models in the classroom, and come to the conclusion that queuing better equips the students with the inherent knowledge of how to reduce manufacturing lead times. Queuing models may provide a theoretical best and worst case scenario which may be invaluable to the process of DES model validation.

2.3 Queuing Models

Historically queuing theory begins in the early 1900's with the work of A.K. Erlang on telephone traffic. In this work Elrang sought to answer such questions as how many telephone circuits and operators are required to satisfy a given demand (Erlang 1909, 1917). Applications to manufacturing, however, begin largely with the later work of J.R. Jackson (1963) which outlines the now well-known Jackson queuing network. A solution for queue length probability distribution is provided for jobshop-like queuing networks. External arrivals enter the first workstation according to the Poisson distribution and are subsequently routed either to the next process with probability p_{ij} or out of the system with probability $1-p_{ij}$. Processing times are also Poisson, and there is infinite buffer capacity. Queue discipline must not rely on future routing or service time information, and thus is considered first come first serve (FCFS). Utilization of any station should not exceed 100%. Under these conditions, a product form solution exists stating that the probability that the network as a whole will be in a state, defined by the number of jobs waiting at each queue, is simply the product of the probabilities of each queue individually having said number of jobs waiting. However, the limitations of the necessary conditions lead researchers to seek out adaptations of the method so as to reflect more realistic systems.

Suri et al (1995) outline the evolution of queuing publications, applications, and software development from the 1960's onward. This includes a significant amount of work on closed-queuing networks in which the number of jobs is constant; jobs cycle repeatedly rather than being created and disposed (Gordon and Newell 1967). This work was largely applied to the parallel programming of resources in computer systems. Later, closed-queuing networks became useful in the modeling of flexible manufacturing systems (FMS). An FMS is a system of numerically controlled machines connected with an automated material handling system, typically with a single load/unload station where parts are mounted on specialty fixtures that travel throughout the system. Because of resource limitations and variability in processing from one part type to the next, understanding of queuing effects is highly important to the effectiveness of FMS operation. Solberg (1977) linked work in closed-queuing networks to FMS systems.

Although useful in many cases, closed-queuing networks are less applicable to typical manufacturing systems than open networks. Unfortunately, the open Jackson network is subject to highly limiting assumptions and exact solutions for more realistic systems are not available (Rabta 2009). This leads researchers to focus on approximations for open queuing networks that would provide results that are more representative of realistic situations (non-Poisson arrivals and processing times, limited buffer size, etc). As outlined by Rabta (2009) queuing approximation techniques include diffusion approximations, mean value analysis, operational analysis, exponentialization approximations, and decomposition methods.

Node decomposition techniques are the most widely used queuing network approximation method and form the basis of research in this thesis. The method allows the modeler to study performance of a queuing network with non-exponential arrival and service times. The approximation involves studying each queue in the network as if they were independent. Rabta (2009) describes the process in three steps: merging of arrivals from outside the system and from other queues into a single arrival flow at each station, computation of performance measures and departure times at each station, and splitting of the overall departure into individual flows to other stations and to the outside. In so doing, the departure times of one station determine the arrival times of any subsequent stations. The specific type of arrival and processing distribution is not specified but instead represented only by a mean and squared coefficient of variation.

With node decomposition, each individual queue in the system is approximated as a renewal process, that is where the interarrival intervals of jobs are independent, identically distributed (iid). A typical queuing network that might be analyzed with node decomposition is shown in Figure 2.1.

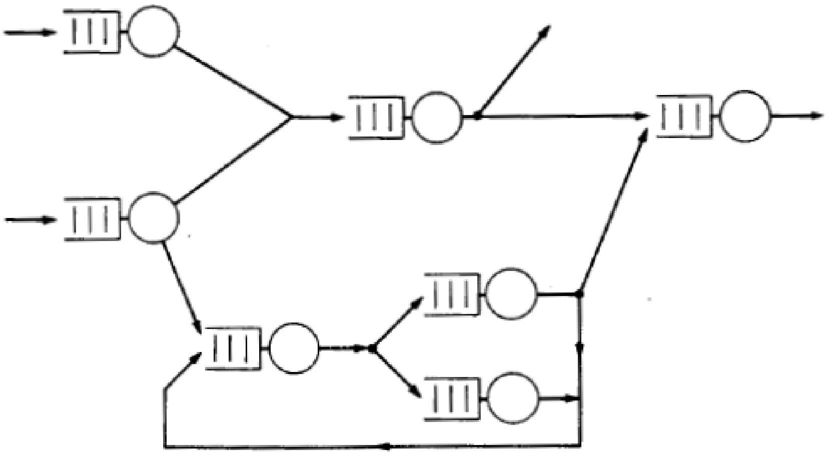


Figure 2.1: (Whitt 1983), Open Queuing Network

In order to consider the queues in isolation, the arrival variability must be calculated at each node according to three key equations for queuing, splitting, and superposition. These actions are depicted in Figure 2.2. The equations essentially keep track of the variability in flow as jobs merge, separate, and travel through variable processes.

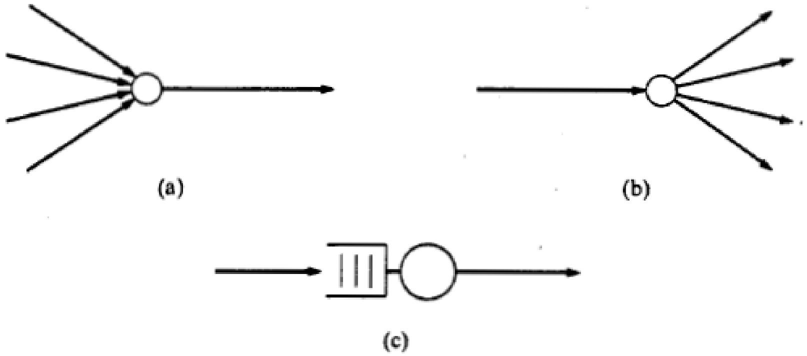


Figure 2.2: (Whitt 1983), The three actions of queuing networks (a) Superposition, (b) Splitting and (c) Queuing

Node decomposition can be applied to a number of types of problems with open queuing networks, and is particularly useful for job shop and dual resource constrained manufacturing environments. One particular application where queuing can be applied is capacity planning and control (Rao 1992). Rao states that capacity adjustments are typically performed through means of overtime, reallocation of workers, routing adjustments, and splitting and overlapping of operations. Queuing decomposition may help managers to successfully control WIP levels, bottleneck utilization, and lead times by enhancing the understanding of the effects these capacity adjustments can have on the system. Rajagopalan and Yu (2000) point out the lack of consideration of lead time performance by most capacity planning models. They present an optimization model to decide whether or not a new machine should be purchased and if so what percentage of a product's demand should be completed on the machine. The expected wait time with an allowance for variability, as determined with queuing theory, is constrained to be less than or equal to the desired lead time. The objective is to minimize the sum of fixed cost from buying machines and production cost.

Queuing theory provides long-run steady state performance measures and is thus a good fit for making long-term strategic decisions. Crowley et al (1994) present a queuing analysis performed during the initial design of a production facility for electromechanical devices. The procedure, described as flow ratio analysis, is based on Jackson queuing networks and provides an early estimate for labor and resource requirements before the construction of a more detailed simulation model. Anderson (1987) also shows the benefits that queuing models can have in the early stages of design for a printed circuit board test cell. Using minimal information about machine reliability, lot size, and routing the modeler very quickly obtains key information on expected flow time, WIP, resource utilization, etc. Because of the quick development time, queuing models are less restrictive in the early stages as compared to simulation, allowing the modeler to make significant structural changes in layout and plan without worrying about disrupting the statistical significance of the result. Furthermore, queuing theory was useful for determination of staffing levels at an L.L. Bean call center for catalog orders (Andrews and Parsons 1993). Previously the staffing level was determined by monitoring the percentage of calls answered within a pre-specified time range. The queuing analysis allows for an economic optimization based on the cost of labor, telephone use cost, and cost of lost customers due to excessive waiting, leading to incredible savings for the company.

Some researchers have found queuing models to be particularly useful in the modeling of systems with rework. Pradhan and Damodaran (2008) study an optoelectronics assembly line which is set up like a flow line except that jobs can fail at any stage and be rerouted back to the station of failure or any previous stage. In addition, multiple product classes are considered with some resource sharing, so the problem of predicting flow time and WIP levels becomes complex. In order to maintain customer satisfaction and reduce late penalties, improved predictions of flow time are required. Using node decomposition techniques the authors study the accuracy of lead time predictions vs. simulation results for 25 problem instances. The study shows an increase in error between queuing theory estimates and simulation observations as the number of nodes shared by different job classes increases. Hu and Chang (2003) study a similar situation in semiconductor wafer fabrication where jobs can fail and be rerouted to any prior station (re-entrant lines). The problem is unique in that it uses a backward queuing network analysis (BQNA) to derive the necessary means and variances required to obtain pre-specified cycle times and WIP levels.

Suri et al (1995) outline a shortcoming of many queuing models in their lack of consideration for learning effects when studying just-in-time (JIT) systems. Based on simulation studies conducted for this research systems can require several hundred hours to reach confidence intervals of one or two minutes for average queue time. Over that time frame (queuing models represent the steady state), learning effects would have changed rework and production rates from the initialization of the study. Reallocation of workers may still be studied assuming workers have reached the end of their learning curves, but short term effects will require simulation or some further queuing model refinement, possibly through use of diffusion approximation techniques. Furthermore, what is lacking from the reviewed case studies is the follow-up analysis once the later-stage simulation models and finally the production lines are put into place. This follow-up would be invaluable in assessing the accuracy of the early-stage queuing models and ultimately for improving the worth of future models.

3 MODEL DEVLOPMENT

The focus of this section is the methodology followed to address the second and third research questions: a) whether or not the proposed model can provide a good estimate for steady state queuing time in a system with immediate feedback rework, and b) whether or not more refined techniques are needed and in what instances. Several steps led up to the choice of queuing model that is examined. The first part of the methodology outlines the process leading to an analytic model while fully explaining the components of the model and assumptions involved. This involves the synthesis of ideas from Hopp and Spearman (2001) and Whitt (1983) regarding machine availability and rework with regards to queuing time. In chapter four a sample problem demonstrates how this model might be used to optimize the allocation of workers with different skill levels, and an initial simulation model is presented to test accuracy of the queuing model. In chapter five, the accuracy of the queuing model is discussed in greater detail. The complete outline of the methodology is depicted in Figure 3.1.

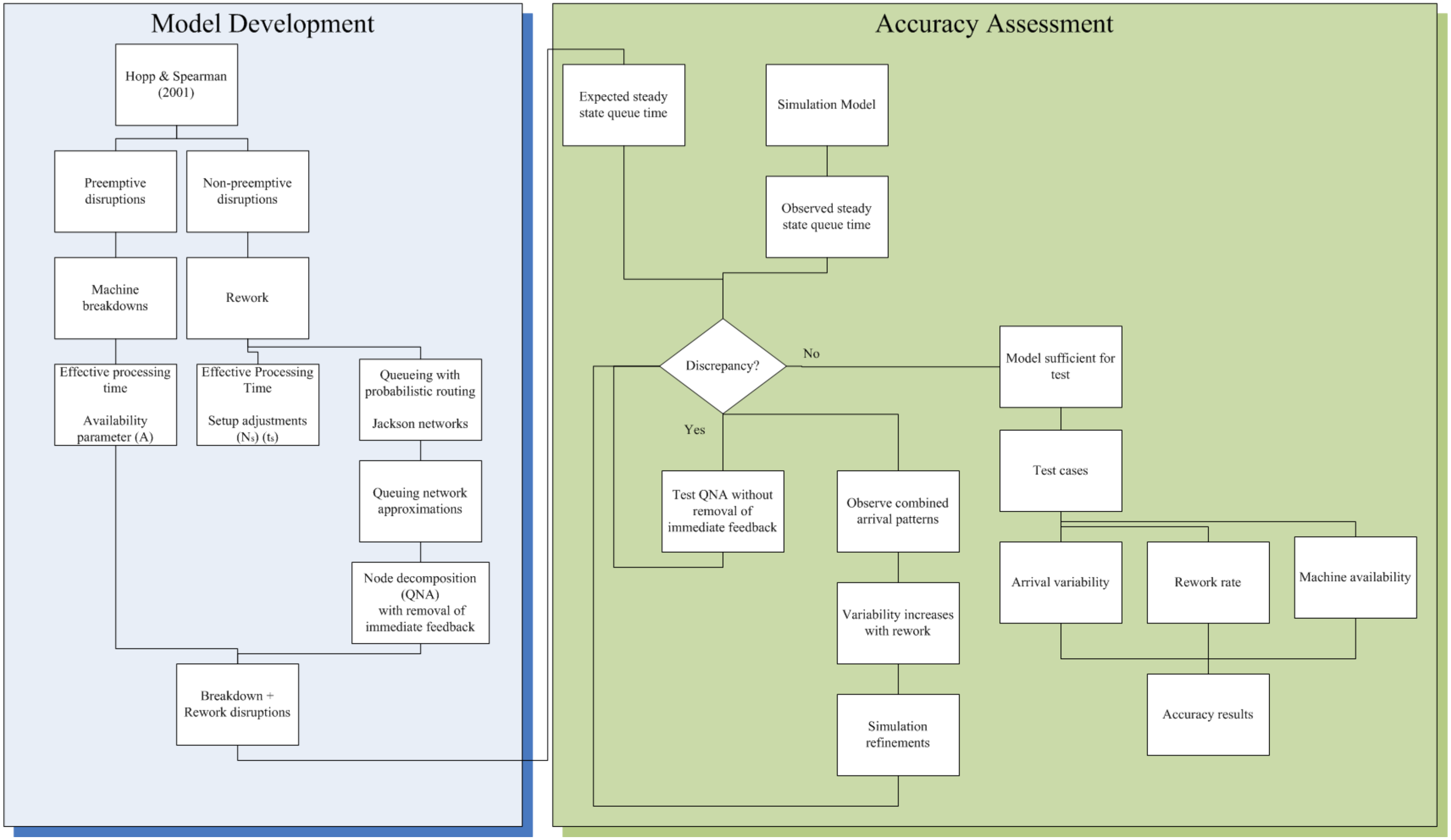

Model Development
Hopp & Spearman (2001)
Preemptive disruptions
Non-preemptive disruptions
Machine breakdowns
Rework
Effective processing time
Availability parameter (A)
Effective Processing Time
Setup adjustments (N_s) (t_s)
Queueing with probabilistic routing
Jackson networks
Queuing network approximations
Node decomposition (QNA) with removal of immediate feedback
Breakdown + Rework disruptions
Accuracy Assessment
Expected steady state queue time
Simulation Model
Observed steady state queue time
Discrepancy?
No
Yes
Model sufficient for test
Test QNA without removal of immediate feedback
Observe combined arrival patterns
Variability increases with rework
Simulation refinements
Test cases
Arrival variability
Rework rate
Machine availability
Accuracy results

Figure 3.1: Methodology framework

3.1 Queuing Model Specification

Inspiration for the use of queuing theory begins with Factory Physics. (Hopp and Spearman, 2001) Here the concept of measuring system variability with the squared coefficient of variation (SCV) is introduced. This quantity is equal to the squared standard deviation of a random variable divided by the mean squared. Random variables with low variability have coefficient of variation less than .75, medium variability have coefficient of variation 0.75-1.33, and high variability have coefficient of variation greater than 1.33 (Hopp and Spearman 2001). This measure is particularly useful because it provides a description of the importance of the variation with respect to the mean. SCV gives a fair reflection of the variation in a dimensionless value. Here standard deviation is σ and mean is μ, and the SCV can be represented as c^2.

$$SCV = c^2 = \frac{\sigma^2}{\mu^2}$$

Equation 3.1

Hopp and Spearman (2001) discuss two main types of disruption, preemptive and non-preemptive, both of which can be quantified using the SCV. Preemptive outages are issues that can occur during the processing of a job, where machine breakdown is the most commonly studied cause. Non-preemptive outages are disruptions that occur between the processing of jobs, such as machine setups. Before these outages can be considered, the "natural variability" should be measured over the long term. This entails variability seen on a regular basis due to unassignable causes and is represented by the SCV with symbol c_0^2which is described by the natural observed mean processing time t_0^2 and natural standard deviation σ_0^2.

$$c_0^2 = \frac{\sigma_0^2}{t_0^2}$$ Equation 3.2

Adjustments are made to the observed natural SCV to account for machine breakdowns and setups, and the resulting value is known as the effective SCV with symbol c_e^2. First, adjustments for machine breakdowns depend on a commonly used metric known as availability, A. Availability of a machine depends on the mean time to failure m_f and mean time to repair m_r.

$$A = \frac{m_f}{m_f + m_r}$$ Equation 3.3

The effective processing time t_e is equal to natural processing time t_0 divided by availability A.

$$t_e = \frac{t_0}{A}$$ Equation 3.4

The natural standard deviation must be adjusted as well as the natural processing time to obtain effective standard deviation of processing time σ_e^2. This adjustment also depends on the standard deviation of repair times σ_r.

$$\sigma_e^2 = \left(\frac{\sigma_0}{A}\right)^2 + \frac{\left(m_r^2 + \sigma_r^2\right)(1-A)t_0}{Am_r}$$ Equation 3.5

Finally the effective SCV c_e^2 for machine breakdowns is the ratio of the effective variability and mean processing times.

$$c_e^2 = \frac{\sigma_e^2}{t_e^2} = c_0^2 + \left(1 + c_r^2\right)A(1-A)\frac{m_r}{t_0}$$ Equation 3.6

In this work the emphasis is on the effects of rework which can be seen as a type of non-preemptive disruption. Hopp and Spearman (2001) also presents an adjustment for non-preemptive disruptions. This adjustment is designed with machine setups in mind, rather than rework, but it may be possible to view rework as equivalent to a setup event that would occur between processing of normal jobs. There are limitations to this technique, but the equations are presented here for completeness.

Where N_s represents the number of jobs completed between setups and t_s represents the average time for each setup, the effective processing time adjusted for non-preemptive disruptions can be specified as,

$$t_e = t_0 + \frac{t_s}{N_s}$$

Equation 3.7

The standard deviation of setup times σ_s^2 is also needed to capture the effective standard deviation with adjustment for setups. Therefore,

$$\sigma_e^2 = \sigma_0^2 + \frac{\sigma_s^2}{N_s} + \frac{N_s - 1}{N_s^2} t_s^2$$

Equation 3.8

Finally, because SCV is the ratio of effective variance to effective processing time,

$$c_e^2 = \frac{\sigma_e^2}{t_e^2} = \frac{N_s\sigma_0^2 + \sigma_s^2 + t_s^2 - \frac{t_s^2}{N_s}}{N_s t_0^2 + 2t_0 t_s + \frac{t_s^2}{N_s}}$$

Equation 3.9

Because the focus of this work is on rework rather than setups, there are some limitations to the above non-preemptive adjustments. Setups occur somewhat regularly as parts are produced (every 100 parts the cutting tool must be changed, etc.) so the number of parts between setups is a feasible measure. Rework is much more unpredictable and could occur in irregular patterns. In the case of rework it is much more convenient to rely on a simple measure such as the expected percentage of parts that will need to be reprocessed. This type of measure is also much more straightforward to model with DES software for the later purpose of validation of the analytic model. Furthermore, it can be noted that the majority of the literature involving queuing network calculations uses this technique, known as probabilistic routing. The jobs may be routed to the next process or to an earlier one for reprocessing based on a matrix of routing probabilities. This technique draws upon the seminal work of Jackson (1954) on queuing systems. In such systems with this probabilistic (Markovian) routing from process to process, the number of times a job has been through the cycle does not affect the likelihood of the next step that will be taken. A part that has been reworked 10 times is just as likely to be reworked again as a first run job.

At this point the work turns to the extensive field of queuing network approximations, the background of which was presented in the literature review, from which it can be recalled that the most commonly used queuing network approximation technique is node decomposition.

3.2 Node Decomposition With Removal of Immediate Feedback Rework

In the case of immediate feedback rework, the external arrival stream and the rework stream are merged together and the two input streams may have different parameters depending on the variability of the process and other factors. In reality, where multiple arrival streams are superimposed, the process is not renewal hence the approximation (Whitt 1980). Keuhn (1979) and Whitt (1983) suggest the implementation of a unique adjustment for processing time and SCV of processing time in the case of immediate feedback rework. Without this adjustment, Keuhn (1979) suggests unacceptable error in the queuing approximations will result due to the failings of the renewal approximation for the superimposed arrivals due to the strong correlation between input and output streams, i.e. arrival times from the rework stream tend to be equal to the external arrival time plus the time required for one processing cycle.

The suggested adjustment is known as the removal of immediate feedback, and essentially means that rework is processed immediately as opposed to being put back at the end of the queue after its first run. In short, all the rework time is administered in one cycle. Because all of the rework is conducted immediately, the jobs are never rerouted into the queue, but the average processing time and variance are adjusted to account for the extra-long processing requirements for reworked parts. The average number of parts in the queue is not affected by this change and thus the expected queue time remains unaltered. The technique deals with the issues of correlation between arrivals and departures, but there still may be error introduced by approximation of a renewal process. Using the same subscripts from Hopp and Spearman (2001) for natural and effective times, and with the parameter p representing the probability that a part must be reworked the adjusted values for processing time and processing variability are as follows.

$$t_e = \frac{t_0}{1-p}$$ Equation 3.10

$$\sigma_e^2 = \frac{\sigma_0^2}{1-p} + \frac{pt_0^2}{(1-p)^2}$$ Equation 3.11

$$c_e^2 = c_0^2 + p(1-c_0^2)$$ Equation 3.12

3.3 Synthesis of Techniques for Rework and Machine Availability

Thus far two adjustments have been described for processing time and variability. From Hopp and Spearman (2001) the adjustment accounts for preemptive disruptions seen in the case of machine breakdown. These adjustments are based on the machine availability parameter. Furthermore, a separate adjustment is seen in Whitt (1983) that accounts for non-preemptive disruptions from rework. Fortunately, both adjustments act on the processing time and variability, and thus can be combined into one analytic model that estimates steady state queuing time.

First consider an effective processing time that combines the adjustment shown in equation 3.4 with the one from equation 3.10. If these adjustments are conducted iteratively, such that the effective processing time from equation 3.4 takes the place of natural processing time variability in equation 3.10, then the result is the following equation 3.13.

$$t_e = \frac{t_0}{A(1-p)} \qquad \text{Equation 3.13}$$

In a similar fashion, the equations for effective SCV of processing time, equation 3.6 and equation 3.12, can be combined. Again, using the iterative technique, c_e^2 from equation 3.6 is used in place of c_0^2 in equation 3.12, and the following relationship results.

$$c_e^2 = c_0^2 + 2A(1-A)\frac{m_r}{t_0} + p\left(1 - c_0^2 - 2A(1-A)\frac{m_r}{t_0}\right) \qquad \text{Equation 3.14}$$

Note that when station availability is 1, meaning no failures, equation 3.14 reduces back down to equation 3.12, and when the probability of rework p is 0 then equation 3.14 reduces to equation 3.6 (assuming the SCV of repair times is equal to 1).

The performance measures for the queuing systems to be studied are average queue time per cycle and average entity wait time (wait time per part). Kingman's equation for queue cycle time $CT_q(G/G/1)$ is used, where the notation indicates applicability to generally distributed arrival and processing times with one server. There is infinite space and first come first serve (FCFS) queue discipline applies.

$$CT_q(G/G/1)=\left(\frac{c_a^2+c_e^2}{2}\right)\left(\frac{u}{1-u}\right)t_e$$

Equation 3.15

When the removal of immediate feedback approach is followed, the SCV of arrivals c_a^2 does not change with rework rate. Effective processing time t_e is derived from equation 3.13 and the effective SCV of processing time by equation 3.14. Routing for rework is removed, meaning the wait time is approximated with only one cycle, so the solution to this equation yields the total expected flow time per part. To divide the total queue time over the expected number of cycles, the expected wait time EW is adjusted as follows.

$$EW=(1-p)CT_q(G/G/1)$$

Equation 3.16

Station utilization is maintained at 80%. The decision regarding station utilization is derived from the relationship between utilization and lead time. As the utilization increases, the expected lead time increases exponentially, and it is at approximately the 80% utilization mark that the exponential effects start to take hold and increase the lead time drastically. In short, 80% utilization marks the point at which increased resource utilization does not pay off because of the inadvertent effects on lead time.

Using this method an expected queue time for each time a part cycles through the process is estimated, with the capability of taking into account both preemptive and non-preemptive disruptions.

4 QUEUING MODEL RESULTS

In order to test the relationship between expected rework rate and the machine availability on queue time, an array of scenarios was tested with rework ranging from 0 to 95 percent and availability from .5 to 1. The study begins with no rework and the percentage is increased in increments of 10% up until 90%. An additional data point at 95% rework is added to show the trend as rework approaches but cannot be allowed to reach 100%. Three cases are tested for machine availability starting at 50% availability and incrementing to 75% and then 100% availability. Queuing time was evaluated according to equation 3.15 using an Excel spreadsheet designed to read in as many as 9 inputs: mean time to failure, mean time to repair, probability of rework, average processing time, standard deviation of processing time, average interarrival time, standard deviation of interarrival time, standard deviation of repair time, and number of parallel stations. Of course, several inputs were held in control. Processing time is held constant with zero standard deviation, primarily to simplify the analysis of arrival variability where external and rework input streams are combined. Also, the standard deviation of repair times and number of parallel stations are constant for the following experiments.

4.1 Effects of Machine Availability and Rework Rate on Queue Time

Figure 4.1 demonstrates the relationship between per-entity queue time and both the probability of rework and machine availability. As expected, queue time increases exponentially with the probability of rework. The effects of machine availability are largely linear, given a constant rework rate. However, the effect of machine availability is more pronounced at higher rework rates. That is, the expected queue time decreases by about half as the availability of the machine increases from .5 to 1.

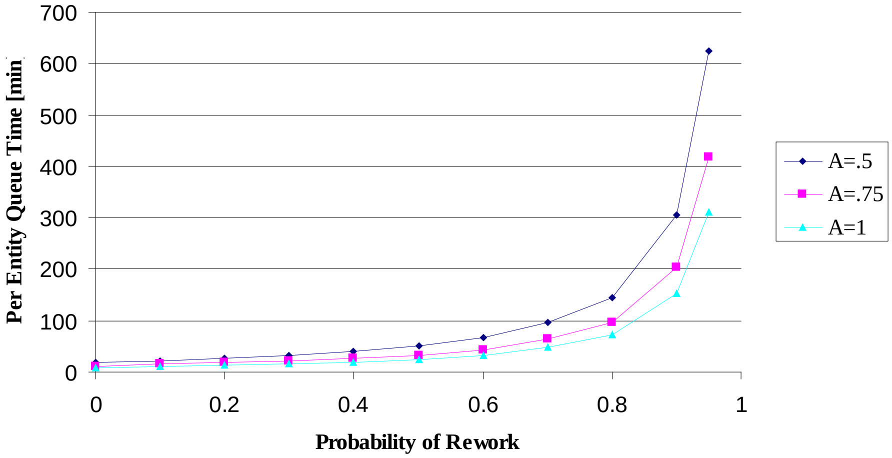


Figure 4.1:	Per-Entity queue time over varying levels of machine availability

The data behind the Figure 4.1 is shown in Table 4.1 for additional reference.

Table 4.1:	Per-Entity Queue Time [min] over levels of rework rate and machine availability

		Availability		
		0.5	0.75	1
Probability of Rework	0	18	11.64	8
	0.1	22.69	14.81	10.29
	0.2	27.08	17.67	12.5
	0.3	32.7	21.41	15.32
	0.4	40.14	26.38	19.05
	0.5	50	33.04	24
	0.6	66	43.62	32
	0.7	97.54	64.68	47.72
	0.8	146	97.05	72
	0.9	306	203.16	152
	0.95	626	417.52	312

Because the iid approximation required by the queuing approximation used, the accuracy of the model estimate may depend on the external arrival variability. For this reason the model output is also tested for three levels of SCV of arrivals (.5, 1, and 2.25). The arrival variability for the previous tests was held at 1. Similar to the machine availability case, the pattern observed with external arrival variability is largely linear with slope dependent upon the rework rate. Queuing time decreases as arrival variability decreases. The effect of arrival variability is less pronounced at low rework rates because it generally causes a four-fold increase in queuing time from the SCV of .5 to the SCV of 2.25. These trends are observed in Figure 4.2. Again, the data behind this plot is presented in Table 4.2

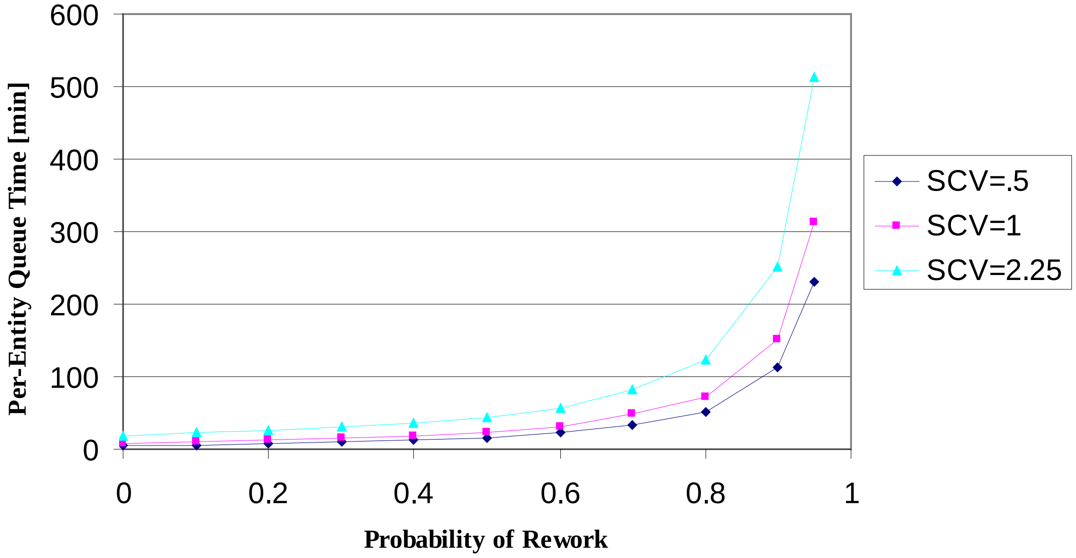


Figure 4.2: Per-Entity queue time over varying external arrival variability

Table 4.2: Per-Entity Queue Time [min] over levels of rework rate and external arrival variability

	SCV, external arrivals		
Probability of Rework	0.5	1	2.25
0	4	8	18
0.1	5.61	10.29	22
0.2	7.29	12.5	25.52
0.3	9.43	15.32	30.04
0.4	12.24	19.05	36.05
0.5	16	24	44
0.6	22	32	57
0.7	33.68	47.72	82.81
0.8	52	72	122
0.9	112	152	252
0.95	232	312	512

4.2 Simulation

An initial simulation study was conducted to assess the accuracy of the queuing model. The simulation model was built using Simul8 discrete event simulation software. For the test, only the effects of rework were included whereas machine availability was held constant at 100%. The results, seen in Figure 4.3 indicate a discrepancy between the calculated values for expected wait time and the observed values from simulation. The discrepancy appears to widen at an increasing rate as the probability of rework increases. These data were collected using only a single replication, and unfortunately in these early stages of study the warmup period and runtime are not noted, although these parameters were held constant for all cases. Later in the simulation refinements discussed in section 5.3 the importance of these parameters is noted.

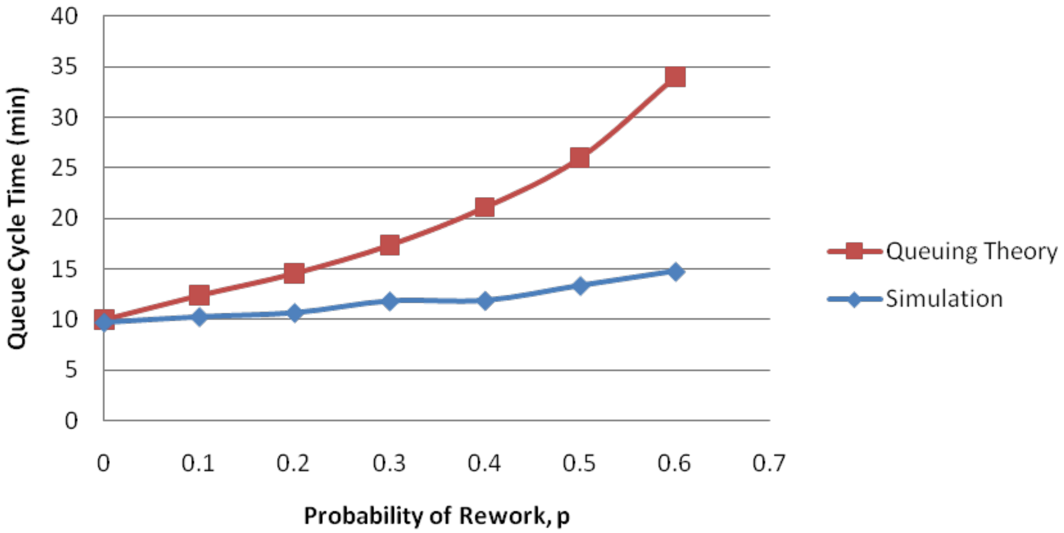


Figure 4.3: Initial finding for accuracy of queuing theory calculations as compared to simulation

Given the unexpected nature of the above result, further simulation was postponed until a better understanding of the system variability was developed.

5 RESOLVING THE GAP BETWEEN QUEING THEORY AND SIMULATION

Given the observed discrepancy between the calculated values and the observed values from simulation, it is desired to study further the possible effects of increasing rework rate on the queuing theory estimates. Of the literature studied on queuing networks and the node decomposition technique, there is little mention of the specific effects of rework rate on model accuracy, specifically how rework rate might affect accuracy of the immediate feedback removal technique. Typically, proposed queuing models are tested for an assortment of different types of queuing networks (with and without feedback, varying in size and complexity, single or multiple part types) and an assessment is made as to the worth of the proposed model over these types of networks. Though narrower in scope, the following analysis attempts to relate model accuracy specifically to system characteristics (rework rate, machine availability, and external arrival variability) which can be easily controlled in both the queuing model and the simulation.

5.1 Node Decomposition without Removal of Immediate Feedback

Because the processing step in these experiments is assumed to have a constant time requirement, as the rework rate approaches 100% the arrival stream from the rework loop becomes nearly deterministic. Every four minutes, a part is completed, and with near certainty is rerouted back to the processing step. Because it is important to these studies to maintain a process utilization of 80%, as the rework rate increases the external arrival rate must decrease. As a result, the external arrival stream which follows a specified statistical distribution will be replaced by regular arrivals from rework. This observation suggests arrival variability will be reduced in cases of high rework rate, which should have a limiting effect on the expected queuing time, which may explain the overestimation of the calculated values.

In the node decomposition approach with removal of immediate feedback, there is no adjustment of the arrival variability term c_a^2 to account for the merging of external and rework arrival streams. Rather, rework is accounted for by adjusting the variability of the processing time, where reworked parts are given longer times than parts without rework. The alternative approach is to allow routing back to the same process, with the job entering at the end of the queue. Using the splitting, queuing, and superposition equations from the node decomposition technique, the SCV of interarrival times c_a^2 is determined for the merged external and rework arrival streams.

In the original node decomposition approach without removal of immediate feedback, the arrival variability c_a^2 is adjusted for the merging of two arrival streams, and it is assumed the combined arrival distribution is stationary iid. Because the distribution of combined arrival streams will not be exactly iid, there is some error introduced at this stage. The hypothesis at hand suggests that the severity of this error from the original node decomposition approach could be less than the error caused by the alternate feedback removal method's lack of direct consideration for arrival variability. Ultimately, as seen in Figure 5.1 this hypothesis is proven to be erroneous. The queuing theory approach without removal of immediate feedback greatly underestimates the average per-cycle waiting time.

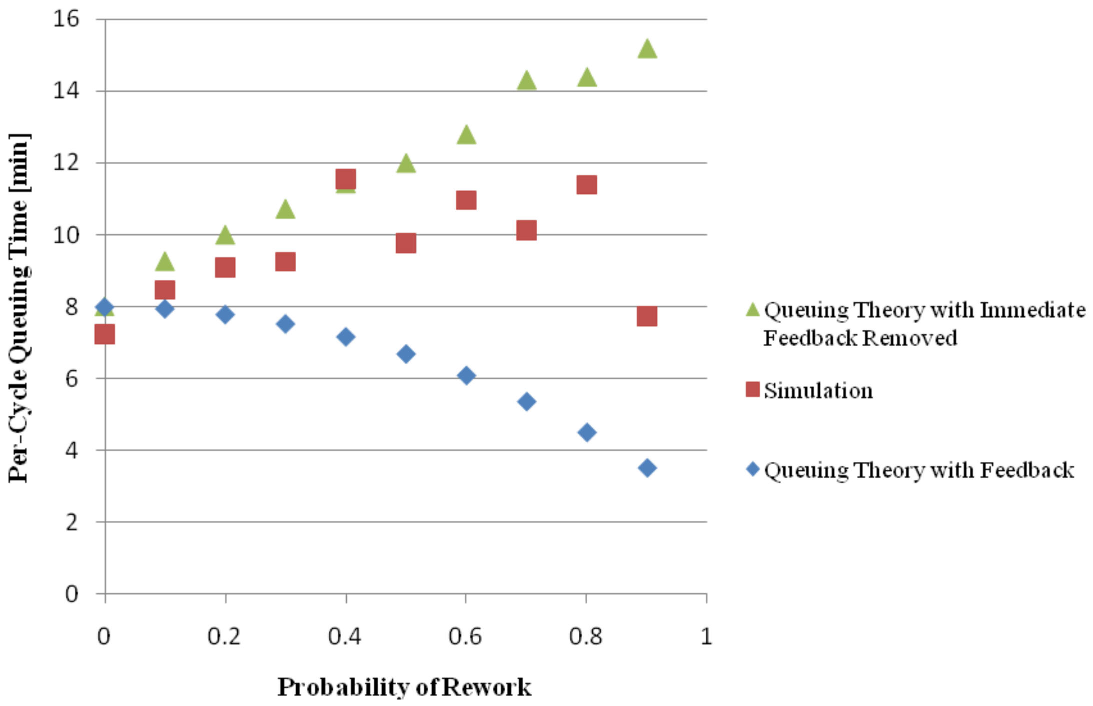


Figure 5.1: Queuing theory calculations with and without removal of immediate feedback for increasing levels of rework

5.2 Inquiry into the Failed Hypothesis

In order to improve the understanding of the discrepancies for both queuing approaches it was required to perform a detailed analysis of the arrival patterns observed in the simulation. To accomplish this, a new simulation model was created in Arena. The change in DES software from Simul8 was done primarily to take advantage of the data analysis capability. A ReadWrite module was placed in the Arena model directly in front of the process step, as shown in Figure 5.2. By writing the arrival times of each entity to

Excel, the arrival patterns from the two input streams, rework and external, could be studied.

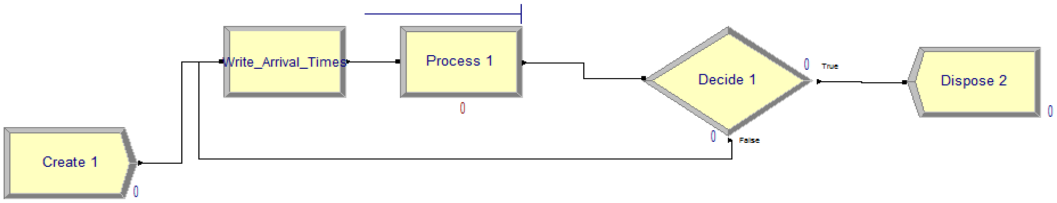
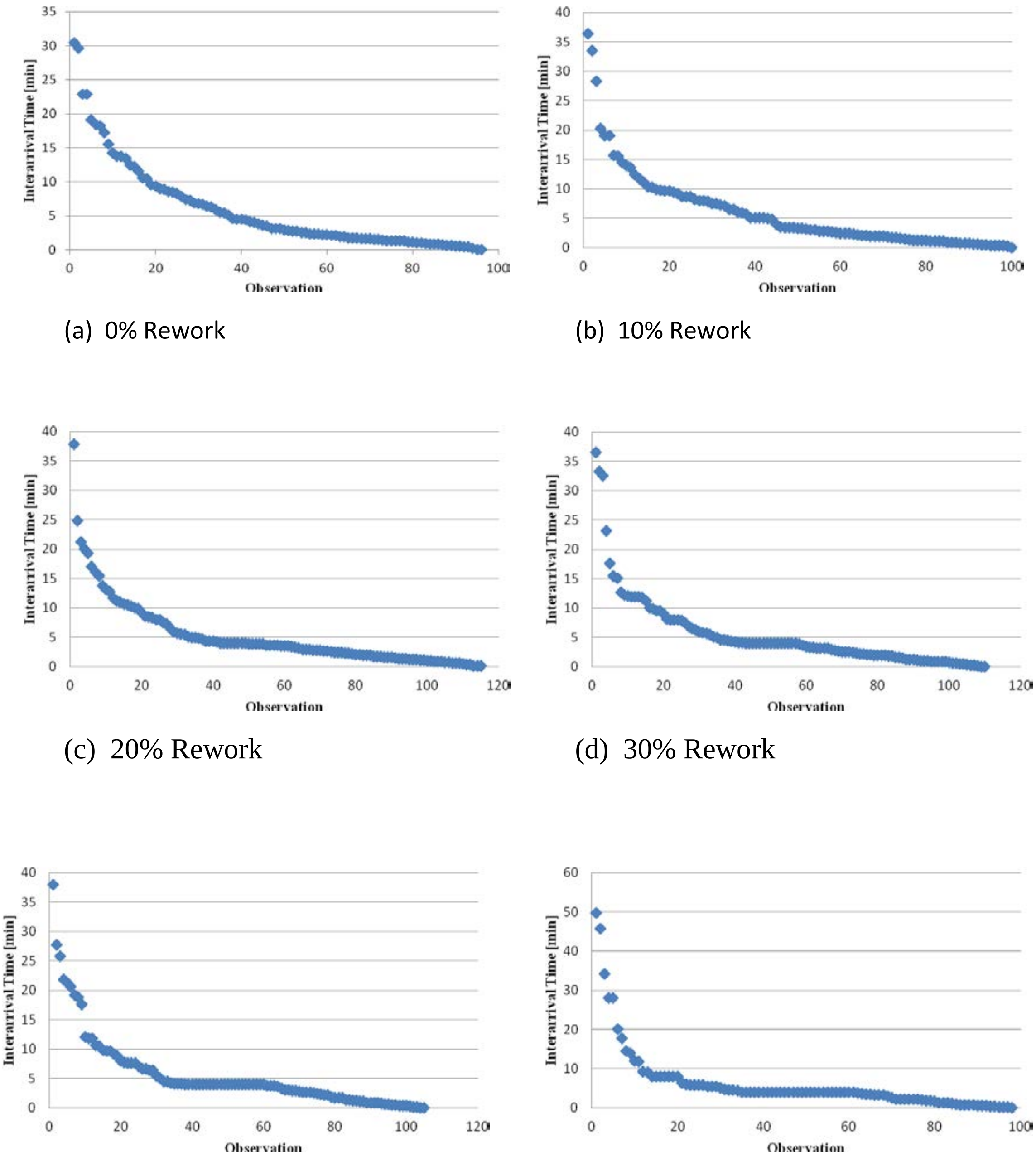


Figure 5.2: Rework loop with ReadWrite module to collect arrival times at process 1

In addition to the arrival pattern observed at the process, the c_a^2 can actually be calculated for the interarrival times. Repeating this for increasing probabilities of rework, one observes some interesting results as shown in figures 5.3(a)-(k). In the cases with low probabilities of rework, the range of observed interarrival times, when organized into descending order, fall into a relatively smooth curve. Starting around the 40% rework case, a tier in the curve can be observed at the interarrival time of 4 minutes. The 4 minute increment is a result of the constant 4 minute processing time at Process 1. As a control, the processing time was held constant so as to understand only the effects of arrival variability. An added benefit is that the proportion of arrivals from the external and rework streams can be discerned. The observed tier at 4 minutes is a result of the arrivals from rework becoming a more significant percentage of the total arrivals. The tier at 4 minutes grows longer with increasing rework rates, and starting at 50% rework a second tier is discernible at the interarrival time of 8 minutes. This occurs when no jobs arrive externally for several cycles. These tiers would increase in increments of 4 minutes if the simulation run time were increased so as to allow for more occurrences of long interarrival times.



(a) 0% Rework

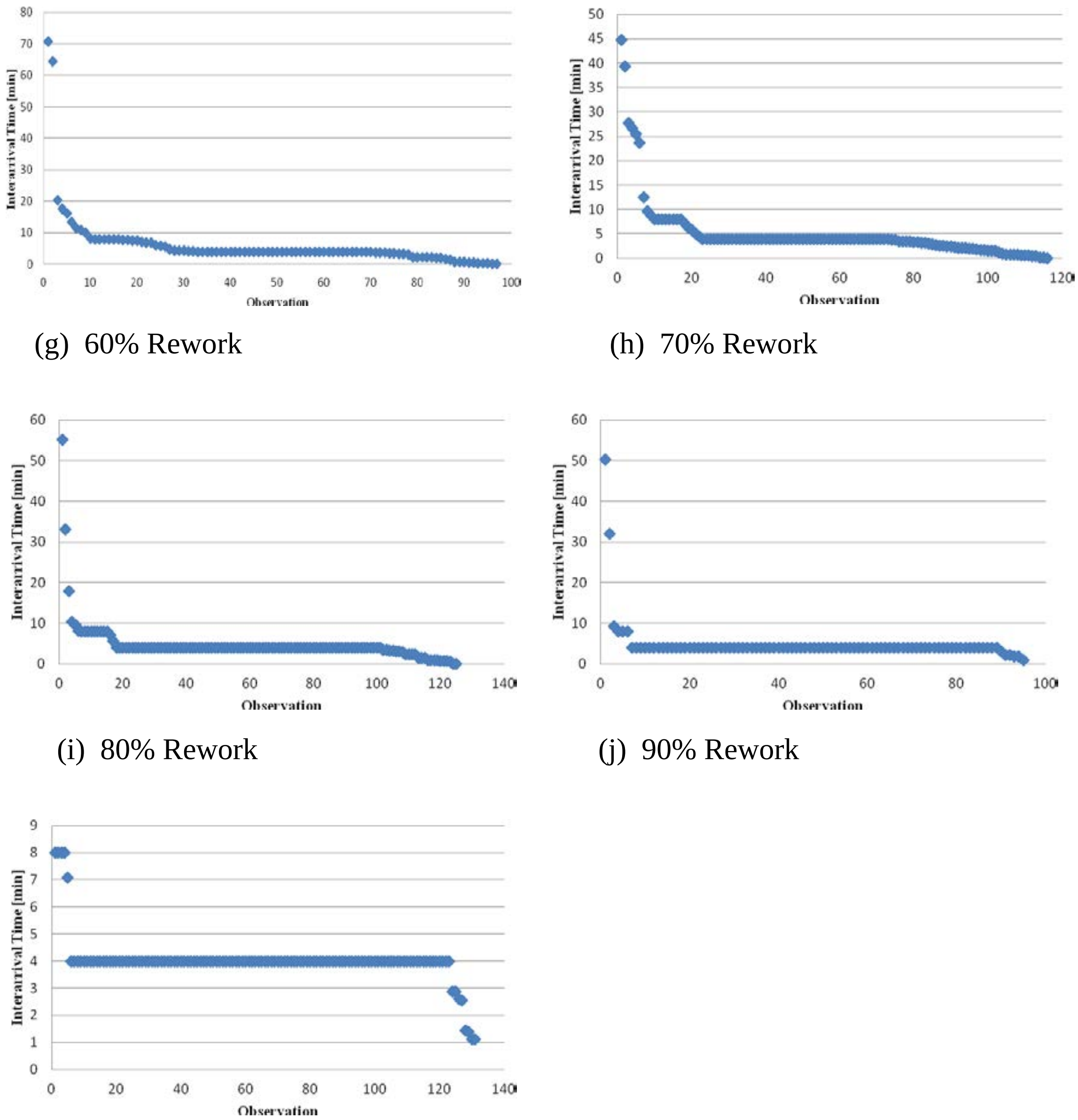
(b) 10% Rework

(c) 20% Rework

(d) 30% Rework

(e) 40% Rework

(f) 50% Rework



(g) 60% Rework

(h) 70% Rework

(i) 80% Rework

(j) 90% Rework

(k) 95% Rework

Figure 5.3 (a)-(k): Observed interarrival times from combined external and rework arrival streams

The SCV of interarrival times is calculated for each rework rate, and the results are shown in Figure 5.4. Clearly the variability is not decreasing with rework rate, despite the fact that a majority of arrivals fall along the 4 minute and 8 minute tiers in these cases. This can be attributed to significantly long interarrival times, i.e. 140 minutes, which can occur when several parts pass through the process without rework consecutively.

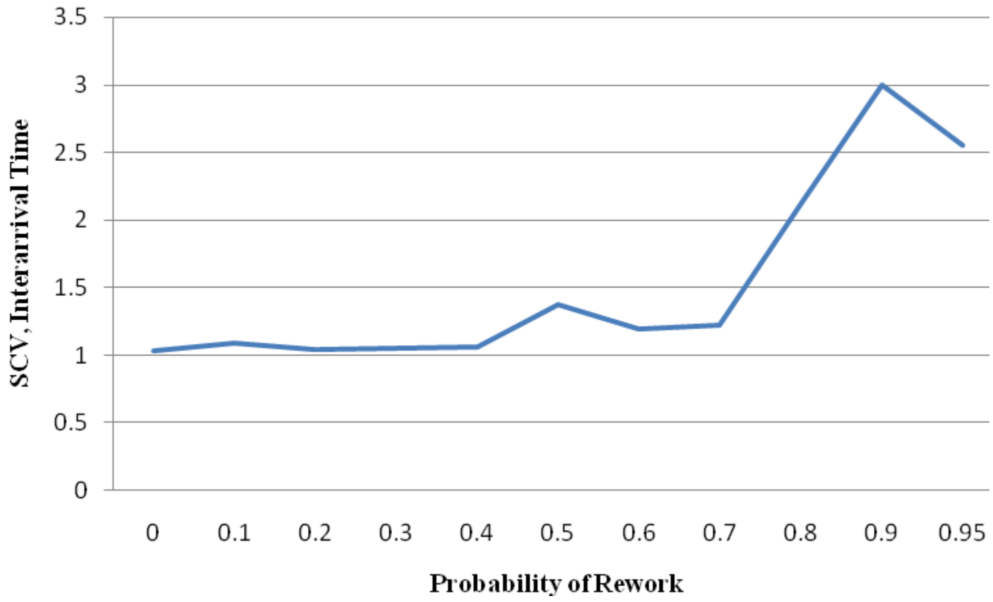


Figure 5.4: SCV of interarrival times for increasing rework rates calculated from simulation data

The drastic difference between these long interarrival times and the standard 4 minute time creates a high overall variability. Note that this may not be evident in figures 5.3(a)-(k) as these represent only one simulation replication each. The variability, in contrast, is calculated as an average of 10 replications to fully capture the effects of randomness. Ultimately, the conclusion from this observation is that queuing time should indeed be increasing with rework and that the node decomposition approach without removal of immediate feedback must show a decreasing trend for reasons other than decreasing arrival variability.

In order for the node decomposition approach without removal of immediate feedback to be accurate, the arrivals from multiple streams must be at least approximately equally distributed. The external arrival distribution is specified to be exponential, so a relevant test then would be to capture the arrival times solely from the rework stream so as to observe the actual distribution.

5.3 Simulation Refinements

It is desired to fully understand the effects of rework rate and external arrival distribution on the accuracy of the queuing theory estimates for steady state waiting time. To accomplish these tasks, a DES model is built using Arena software and statistical analyses are conducted to compare DES observations with the expected values as determined using equations 3.15 and 3.16. A screenshot of the Arena model structure is shown in Figure 5.5. The decide module is conducted by percentage and represents the stated rework rate.

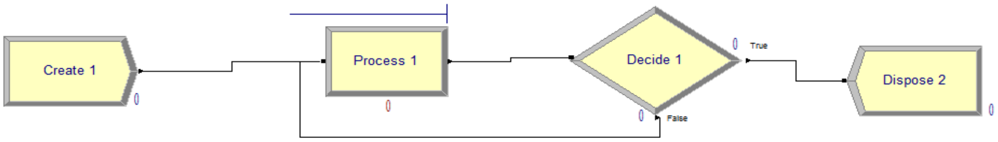


Figure 5.5: Arena model structure used to obtain queue time observations

As discussed in Kelton et al. (2010) several steps are required to ensure confidence in the results from a steady state type simulation. The first requirement is that the warmup period must be set so as to eliminate any bias caused by start-up conditions. Initially, there are no parts waiting in the queue for process 1, which causes the initial queuing time to be below average. Kelton et al (2010) suggests that warmup period should be set during which the simulation runs but data is not collected. To decide the warmup length, the simulation is initially run with no warmup period with 10 replications for a long time period, in this case 700 hours. For each replication, data on the average queue time is collected for the entire 700 hours, and this data is plotted over time using Arena's output analyzer. A typical plot of this data is shown in Figure 5.6. The time at which the average queue time settles for all 10 replications can be assumed to be a reasonable warmup period, after which startup conditions have been exhausted. Figure 5.6 shows the queue time leveling off after about 20,000 minutes or 333 hours.

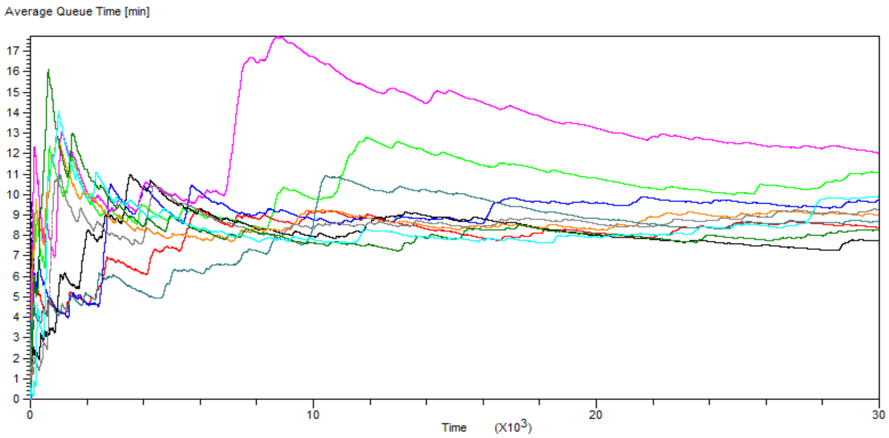


Figure 5.6: Transient behavior of average queue time for 20% rework, exponential arrival distribution

After setting the warmup period, a run time and number of replications must be specified in order to ensure the steady state value for queuing time has been achieved. Both techniques of adding run time and adding replications result in the collection of more data points, which facilitates reduction of the 95% confidence interval. That is, the range over which the average queue time will fall 95% of the time. Extreme values may occur due to the stochastic nature of the system, but with the collection of more data these extreme points will be outweighed by the many data points which fall much closer to the average value. With more data points, this half width of the confidence interval becomes more precise, so the best approach is to pre-specify the desired level of precision. For the case of these studies, the desired half width is specified so that it must be less than 1 minute. When a test scenario having parameters for warmup period, run time, and number of replications provides the desired confidence interval, then the performance measures (average queue time and average entity wait time) can be collected as the steady state values.

Figure 5.7 shows how the number of replications and run time affect the simulation value for average queue time. The scenario shown is for 80% rework rate with external interarrival time taking the exponential distribution. The value starts out sporadic in nature and eventually settles around the horizontal line which in this case represents the calculated queue time using the queuing theory model. The number of replications affects the run time requirement to reach the desired confidence interval, but in general is less crucial than the actual run time. Without a long enough run time, the steady state cannot be reached. Figure 5.8 shows the absolute percent error between the simulated values and queuing theory values for the various run times and number of replications, where the colors on the surface plot represent ranges (2% in height) of the absolute percent error. Note that as the number of replications increases, the percent error decreases but at short run times, such as 2400 minutes, even at high numbers of replications the system may not reach near zero percent error.

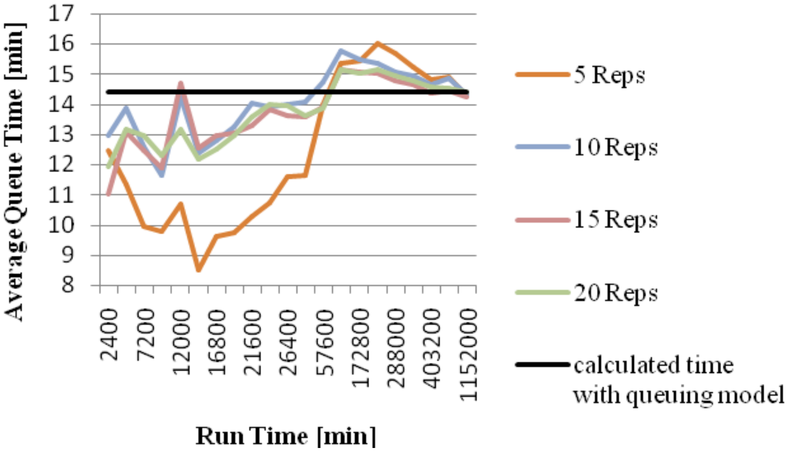


Figure 5.7: Effects of simulation run time and number of replications on average queue time, 80% rework rate and exponential arrival distribution

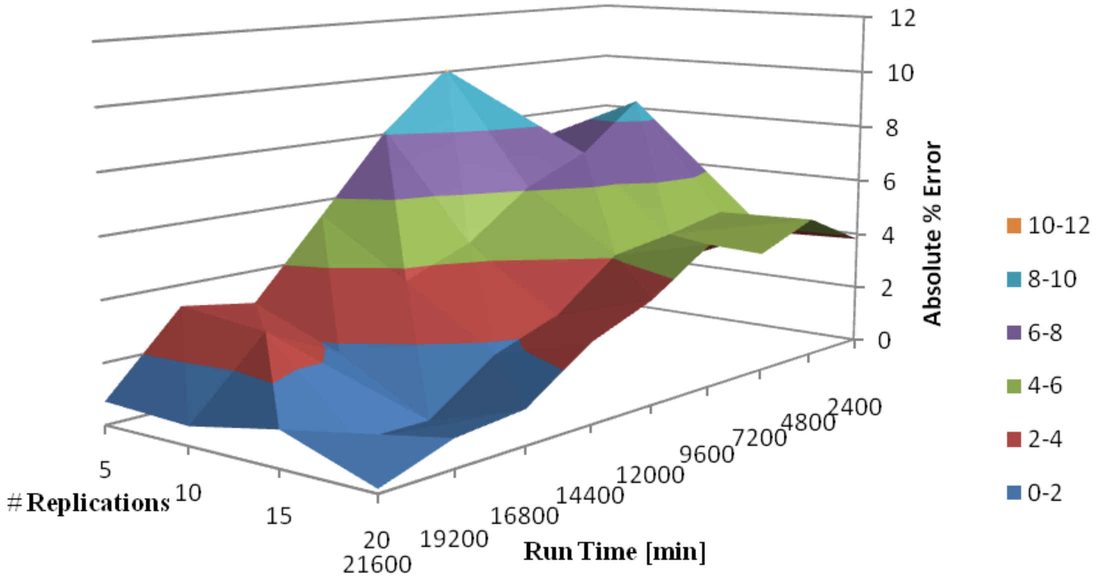
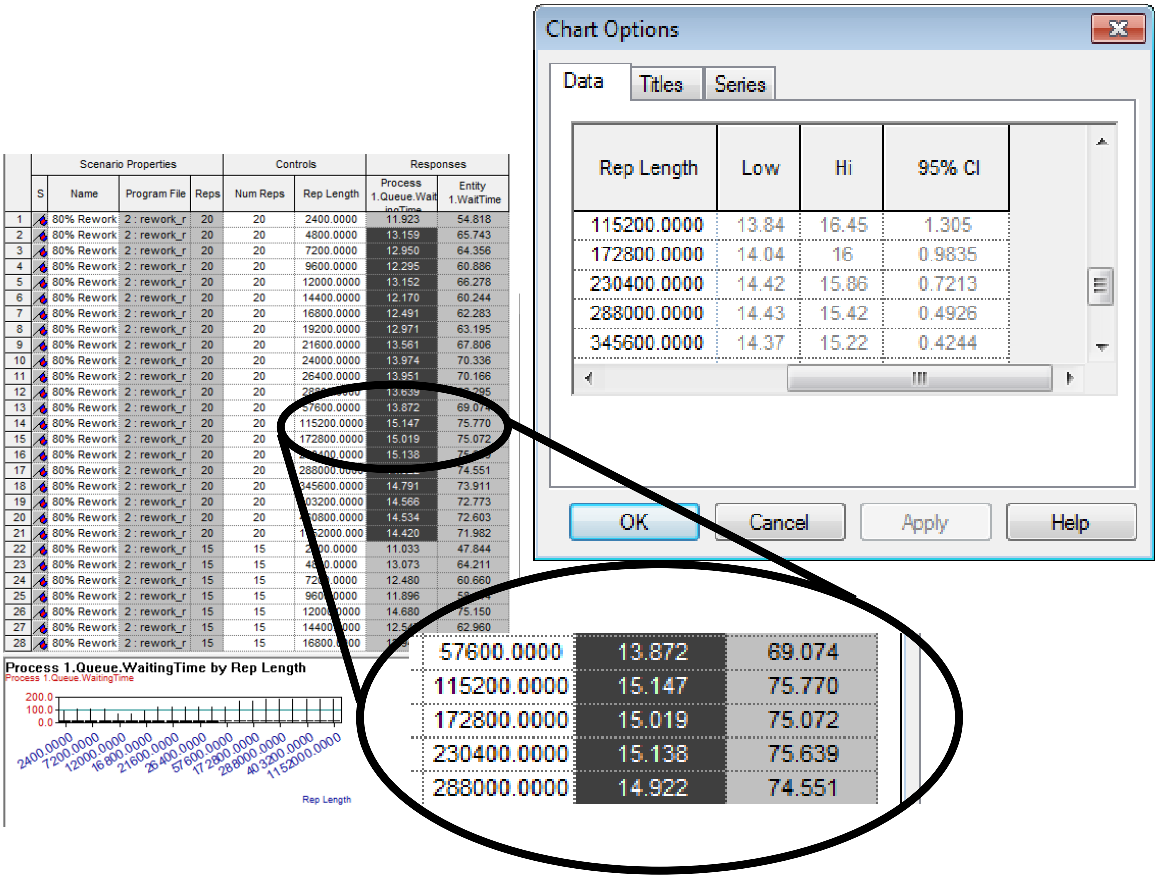


Figure 5.8: Surface plot of absolute % error vs. run time and # of replications, 80% rework rate and exponential arrival distribution

Using Arena’s process analyzer (PAN) tool, data can quickly be collected over a range of increasing run time and, if needed, increasing number of replications. At first, 5 replications are run over increasing run time and the half width of the confidence interval is observed for each run. When the half width drops below 1 minute and stays at this width for two consecutive scenarios, then for the second scenario the steady state values are taken. If after an exceedingly long run time, here 800 days, the half width does not fall and stay below 1 minute, the number of replications is increased by 5 and the procedure is repeated. Figure 5.9 shows a screen shot of the process analyzer for 80% rework and external interarrival times set to the exponential probability distribution.



	Scenario Properties				Controls		Responses	
	S	Name	Program File	Reps	Num Reps	Rep Length	Process 1.Queue.WaitingTime	Entity 1.WaitTime
1		80% Rework	2 : rework_r	20	20	2400.0000	11.923	54.818
2		80% Rework	2 : rework_r	20	20	4800.0000	13.159	65.743
3		80% Rework	2 : rework_r	20	20	7200.0000	12.950	64.356
4		80% Rework	2 : rework_r	20	20	9600.0000	12.295	60.886
5		80% Rework	2 : rework_r	20	20	12000.0000	13.152	66.278
6		80% Rework	2 : rework_r	20	20	14400.0000	12.170	60.244
7		80% Rework	2 : rework_r	20	20	16800.0000	12.491	62.283
8		80% Rework	2 : rework_r	20	20	19200.0000	12.971	63.195
9		80% Rework	2 : rework_r	20	20	21600.0000	13.561	67.806
10		80% Rework	2 : rework_r	20	20	24000.0000	13.974	70.336
11		80% Rework	2 : rework_r	20	20	26400.0000	13.951	70.166
12		80% Rework	2 : rework_r	20	20	[illegible]	13.639	[illegible]
13		80% Rework	2 : rework_r	20	20	57600.0000	13.872	69.074
14		80% Rework	2 : rework_r	20	20	115200.0000	15.147	75.770
15		80% Rework	2 : rework_r	20	20	172800.0000	15.019	75.072
16		80% Rework	2 : rework_r	20	20	230400.0000	15.138	75.639
17		80% Rework	2 : rework_r	20	20	288000.0000	14.922	74.551
18		80% Rework	2 : rework_r	20	20	345600.0000	14.791	73.911
19		80% Rework	2 : rework_r	20	20	403200.0000	14.566	72.773
20		80% Rework	2 : rework_r	20	20	460800.0000	14.534	72.603
21		80% Rework	2 : rework_r	20	20	1152000.000	14.420	71.982
22		80% Rework	2 : rework_r	15	15	2400.0000	11.033	47.844
23		80% Rework	2 : rework_r	15	15	4800.0000	13.073	64.211
24		80% Rework	2 : rework_r	15	15	7200.0000	12.480	60.660
25		80% Rework	2 : rework_r	15	15	9600.0000	11.896	[illegible]
26		80% Rework	2 : rework_r	15	15	12000.0000	14.680	75.150
27		80% Rework	2 : rework_r	15	15	14400.0000	[illegible]	62.960
28		80% Rework	2 : rework_r	15	15	16800.0000	[illegible]	[illegible]

Process 1.Queue.WaitingTime by Rep Length

Chart Options

Rep Length	Low	Hi	95% CI
115200.0000	13.84	16.45	1.305
172800.0000	14.04	16	0.9835
230400.0000	14.42	15.86	0.7213
288000.0000	14.43	15.42	0.4926
345600.0000	14.37	15.22	0.4244

57600.0000	13.872	69.074
115200.0000	15.147	75.770
172800.0000	15.019	75.072
230400.0000	15.138	75.639
288000.0000	14.922	74.551

Figure 5.9: Screen shot of process analyzer used to gather data

A box and whisker plot is created for each run time (called rep length in the PAN) for a given number of replications. The plot shown in the figure 5.9 is for 20 replications. In the chart options window, seen to the right of Figure 5.9, the 95% confidence interval on the observed average queue time is given. Note that for run time of 172800 minutes, the confidence interval falls below 1 minute for the first time at .9835 minutes. To ensure that the confidence interval is consistently below 1 minute, the average queue time is taken at the subsequent run time of 230400 minutes where the confidence interval is .7213 minutes. The average queue time for the simulation with 20 replications of length 230400 minutes is 15.138 minutes and the average queue time per part is 75.639 minutes.

5.4 Testing High and Low Arrival Variability Cases

This procedure of determining the steady state queue time values from simulation are repeated for rework rates ranging from 0% to 90% in increments of 10%, and finally again for 95% rework. Furthermore, it is desired to observe the effects of the external arrival distribution, or the arrival variability, on the accuracy of the queuing theory estimates. For this reason the queuing theory estimates are computed for low, medium, and high arrival variability. These correspond to SCV of arrivals of 0.5, 1.0, and 2.25 respectively. However, because the queuing theory estimate is based only on the moments of the distribution and the simulation requires specification of some probability density function, appropriate distributions are fit with parameters corresponding to these arrival SCV's. For the exponential distribution, the mean and standard deviation are equal to each other and the SCV is then naturally equal to 1.0. Exponential distribution is then assumed for the case where SCV equals 1.0. As demonstrated in Whitt (1983) the Erlang distribution is convenient for low variability arrivals as its SCV is always less than 1. Furthermore, the hyperexponential distribution will always have SCV greater than 1. Therefore these are the distributions fitted in the simulation for low and high variability.

Chinnaswamy (2005) demonstrates techniques for fitting parameters to the Erlang and Hyperexpnential distributions so that a desired SCV can be achieved. The Erlang distribution is used to represent the summation of *r* arrival streams, where each individual stream is exponential. In addition to the number of streams, the rate parameter λ is needed which is the arrival rate of each input stream. This is the inverse of the interarrival time. Equations 5.1 and 5.2 represent the mean and squared standard deviation of the Erlang distribution.

$$\mu = \frac{r}{\lambda} \qquad \text{Equation 5.1}$$

$$\sigma^2 = \frac{r}{\lambda^2} \qquad \text{Equation 5.2}$$

Subsequently, the SCV is as shown in equation 5.3.

$$c_a^2 = \frac{\frac{r}{\lambda^2}}{\frac{r^2}{\lambda^2}} = \frac{1}{r} \qquad \text{Equation 5.3}$$

As an example, when the process has 0% rework, the external arrival rate which allows 80% utilization of the process is one part every 5 minutes with exponential arrivals. If the desired SCV is .5, then the number of arrival streams must be 2. Since the average interarrival time must remain 5 minutes to maintain the desired 80% utilization, the interarrival time for each of the 2 streams must then be 2.5 minutes. Using Arena, the expression used in the Create module is then ERLA(2.5, 2).

The case for the Hyperexponential distribution is somewhat more complicated. This distribution is used to represent a mixture of r arrival streams with exponential distributions. The mean of the Hyperexponential distribution is shown in equation 5.4 where p_i is the probability that arrivals will come from arrival stream i.

$$\mu = \sum_{i=1}^{r} \frac{p_i}{\lambda_i}$$ Equation 5.4

When there are 2 arrival streams being mixed together, the variance of the Hyperexponential distribution can be stated according to equation 5.5

$$\sigma^2 = 2\left(p_1 \lambda_1^{-2} + p_2 \lambda_2^{-2}\right)$$

Equation 5.5

Finally, the SCV of the Hyperexponential is shown in equation 5.6.

$$c_a^2 = \frac{2\left(p_1 \lambda_1^{-2} + p_2 \lambda_2^{-2}\right)}{\left(\frac{p_1}{\lambda_1}\right)^2 + \left(\frac{p_2}{\lambda_2}\right)^2} - 1$$ Equation 5.6

As described in Whitt (1980), the probabilities p_i can be fitted to a desired SCV independently from the rate parameters of each stream under the assumption of balanced means. Once the probabilities are specified to provide the desired SCV, the rate parameters can be calculated based on the interarrival time required to attain 80% utilization. The concept of balanced means is represented by equation 5.7.

$$p_1 \lambda_1^{-1} = p_2 \lambda_2^{-1}$$ Equation 5.7

Equation 5.8 shows how the probability of each arrival stream can be determined given some desired SCV.

$$p_1 = \frac{1 \pm \sqrt{\frac{(c^2 - 1)}{(c^2 + 1)}}}{2} \qquad \text{Equation 5.8}$$

In the test cases presented an SCV of 1.5 was used, which corresponds to probabilities of .81 for p_1 and .19 for p_2. Finally, the arrival rate parameter for the two arrival streams can be found using equation 5.9.

$$\lambda_1 = \frac{2 p_1}{\mu_1} \qquad \text{Equation 5.9}$$

For each rework rate, there is an associated μ_1 interarrival time that ensures 80% process utilization.

5.5 Simulation test with rework and breakdowns

In addition to the simulation test for varying arrival variability, simulation is performed taking into account machine availability. This process is relatively straightforward. In the Arena simulation model, a resource associated with the processing step is given a time-based failure pattern such that the mean time to failure is equal to 1 minute and the mean time to repair is also 1 minute, giving an availability of 50%. This change has the effect that the interarrival times must be doubled relative to the 100% availability case in order to ensure the process does not become over-utilized.

6 RESULTS ON QUEING THEORY ACCURACY

6.1 Observed Error for high, medium, and low external arrival variability

Accuracy of queuing estimates is provided over the full range of rework rate from 0% to 95% for three levels of arrival variability. These levels of external arrival variability are specified by the squared coefficient of variation of arrivals. Specifically the values are .5, 1, and 2.25 for low, medium and high. In the simulation model, the low, medium, and high levels must be fitted with a distribution and those used are Erlang, Exponential, and Hyperexponential, respectively. Figure 6.1 shows the first result for queuing model accuracy vs. the simulation with exponential arrival distribution. The error bars shown around the simulation data points indicate the 95% confidence interval half width. 95% of the time the average per-cycle queue time will be within this error bar.

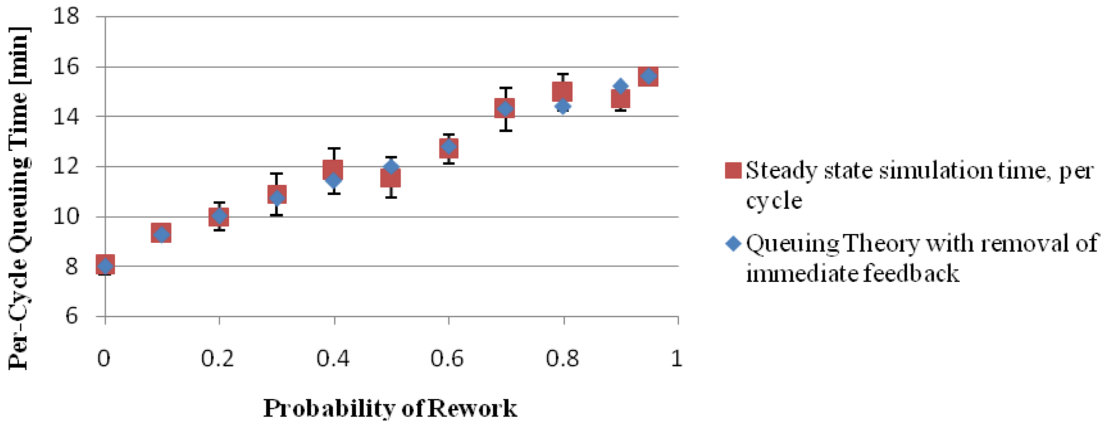


Figure 6.1: Comparison of per-cycle queuing time for queuing calculations and simulation for exponential arrival distribution, SCV=1

The primary metric used to establish accuracy of the queuing calculations is the per-cycle queue time. This represents the average time a job waits before processing each time it cycles through the system. Figure 6.1 demonstrates this per-cycle queue time over incremental levels of rework for the exponential arrival distribution. With external arrivals fitting the exponential distribution the queuing calculations should be at their most accurate as this is the case in which the node decomposition technique with removal of immediate feedback is proven to be exact as compared to the system with feedback (Kuehn 1979). The percent error at each data point is shown in Figure 6.2.

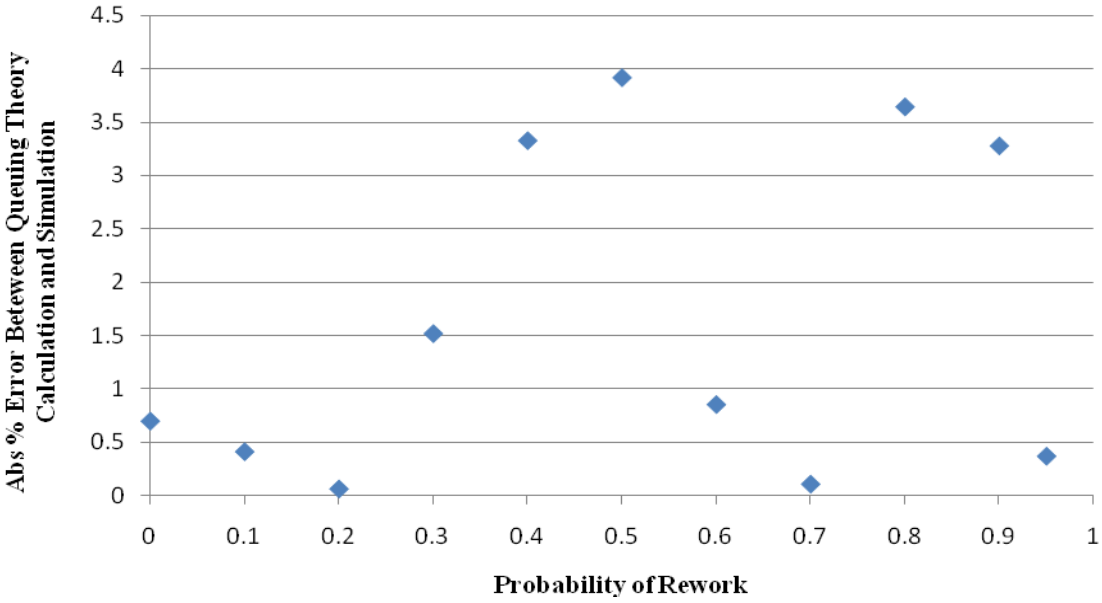


Figure 6.2: Percent error for per-cycle queue time with exponential arrival distribution, SCV=1

There is no discernable relationship between probability of rework and accuracy of the queuing calculations according to Figure 6.2. Less than 4% error is observed in all cases. It should be noted, however, that with increasing probability of rework the run time required by the simulation to resolve to the specified confidence interval with half-width equal to one minute significantly increases, and in the cases of .9 and .95 probability of rework the number of simulation replications also had to be increased to reach the desired resolution. Table 6.1 demonstrates the run times and number of replications used to aquire each data point.

Table 6.1: Simulation run time and number of replications, Exponential arrival distribtion, SCV=1

Probability of Rework	half width, per cycle [min]	Run Time [min]	# Replications
0	0.3947	230400	5
0.1	0.3256	115200	5
0.2	0.5406	230400	5
0.3	0.8416	345600	5
0.4	0.9109	230400	5
0.5	0.8111	230400	5
0.6	0.5912	1152000	5
0.7	0.8551	460800	5
0.8	0.7403	345600	5
0.9	0.5131	11520000	10
0.95	0.3256	460200	10

In addition to the per-cycle queue time, it is also interesting to look at the comparison between the per-entity queue time as obtained from the queing theory calculation and from the simulation. To the practicioner, this metric may be more relevant as it demonstrates the potential cost per part as well as the feasibility of meeting a delivery deadline. Figure 6.3 shows average total waiting time per entity as summed over the total number of cycles through the system. Accuracy remains good for the per-entity case, however it should be noted that the corresponding half widths for each data point are considerably higher than in the per-cycle case. This is attributed to the added uncertainty in the number of times an entity may cycle throughout the system. With very high rework rates, there can be a significant difference in the minimum number of cycles required and the maximum, creating a wider range of total queing time per entity.

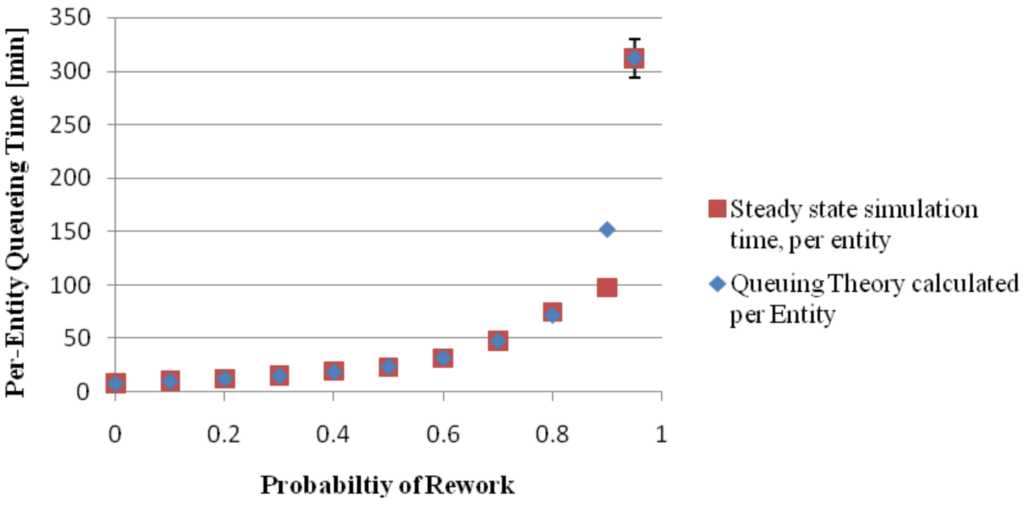


Figure 6.3: Average queuing time per entity for exponential arrival distribution

Figure 6.4 demonstrates the percent error in the per-entity queue times shown in Figure 6.3. Error remains below 4% for all cases other than for the instance of .9 probability of rework where the error is over 50%.

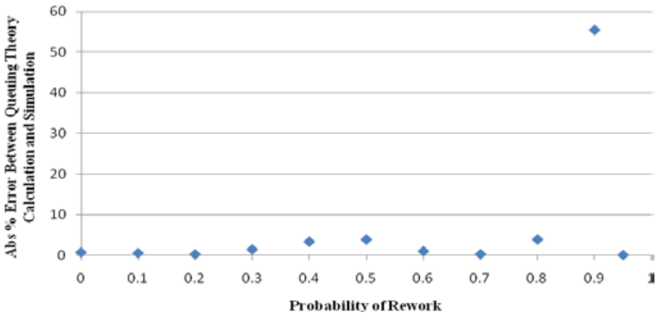


Figure 6.4: Percent error for per-entity queue time with exponential arrival distribution, SCV=1

A similar presentation of results is repeated for the low and high arrival variability cases. Figure 6.5 shows the per-cycle accuracy of the queuing calculations for the Erlang arrival pattern, which along with Figure 6.6 shows a decreasing percent error as probability of rework increases. Table 6.2 shows the simulation run time and replication requirements needed to reach the desired confidence interval half width of less than one minute.

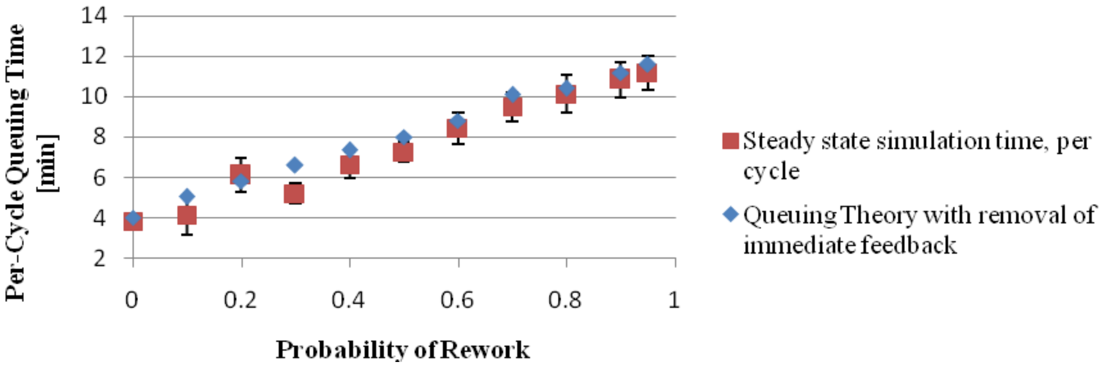


Figure 6.5: Comparison of per-cycle queuing time for queuing calculaitons and simulation for Erlang arrival distribution, SCV=.5

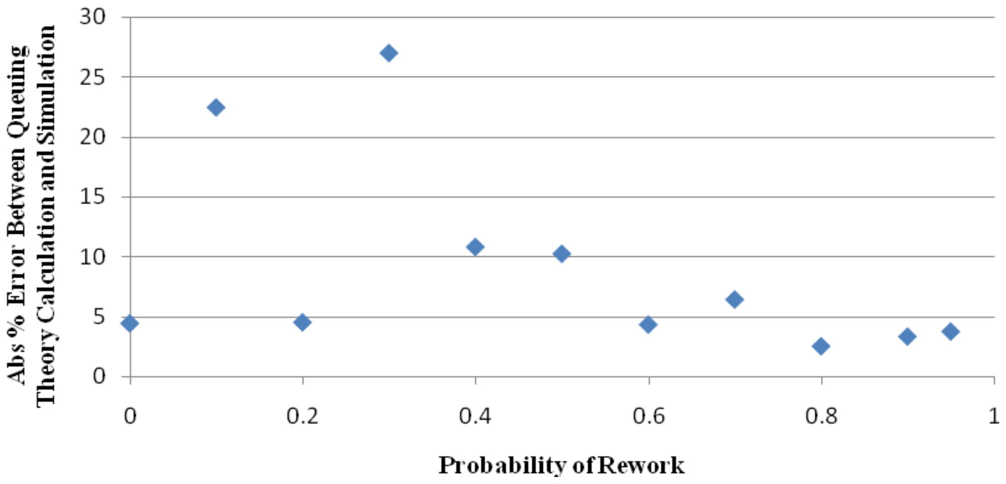


Figure 6.6: Percent error for per-cycle queue time for Erlang arrival distribution, SCV=.5

Table 6.2: Simulation run time and number of replications, Erlang arrival distribution, SCV=.5

Probability of Rework	Half Width, per-cycle [min]	Run Time [min]	# Replications
0	0.2619	7200	5
0.1	0.9643	12000	5
0.2	0.8588	12000	5
0.3	0.5136	24000	5
0.4	0.6555	172800	5
0.5	0.4919	28800	5
0.6	0.7568	230400	5
0.7	0.7061	288000	5
0.8	0.9056	345600	5
0.9	0.8892	403200	5
0.95	0.8223	345600	15

For the Erlang arrival distribution (low external arrival variability) the pattern of percent error between queing calculations and simulation continues for the per-entity queuing times. Figure 6.7 demonstrates good accuracy despite increasing width of the confidence interval as rework rate increases.

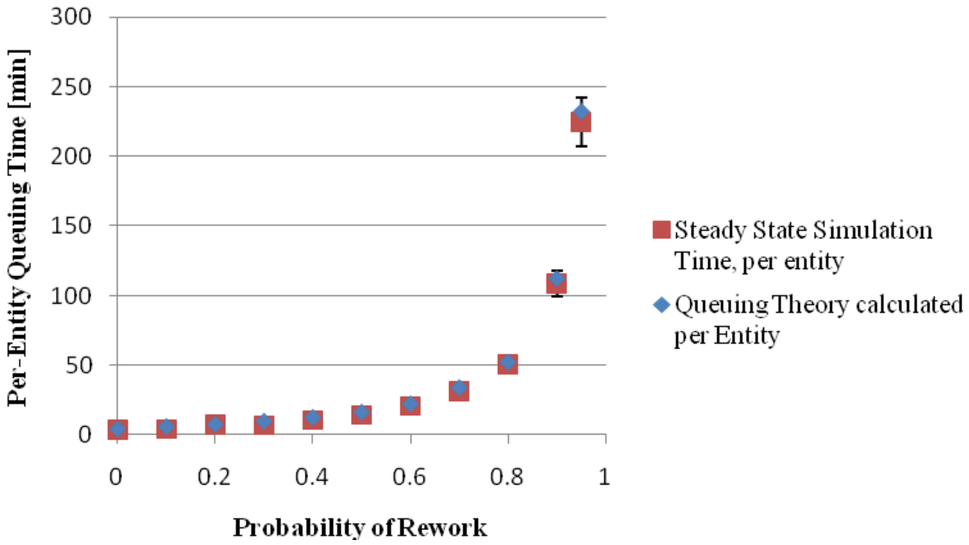


Figure 6.7: Average queuing time per-entity for Erlang arrival distribution, SCV=.5

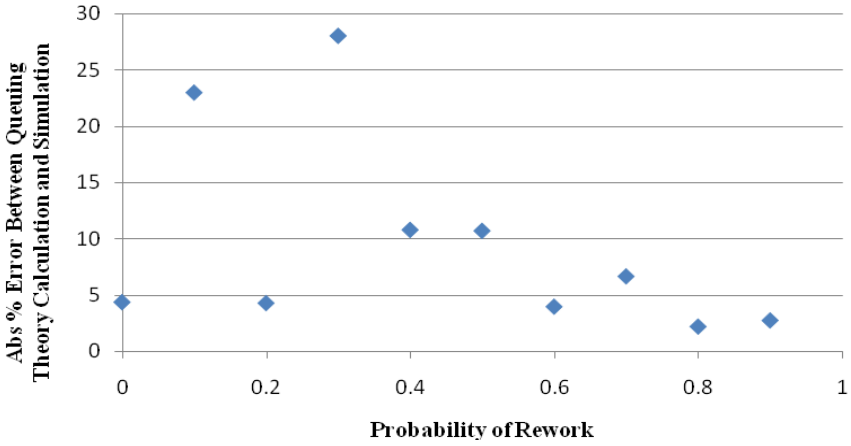


Figure 6.8: Error for per-entity queue time for Erlang arrival distribution, SCV=.5

Although more erratic, the accuracy of the queuing calculations for highly variable arrivals also tend to increase as the probability of rework increases, seen in Figure 6.9. This pattern is observed most clearly in the plot of percent error over increasing levels of rework in Figure 6.10.

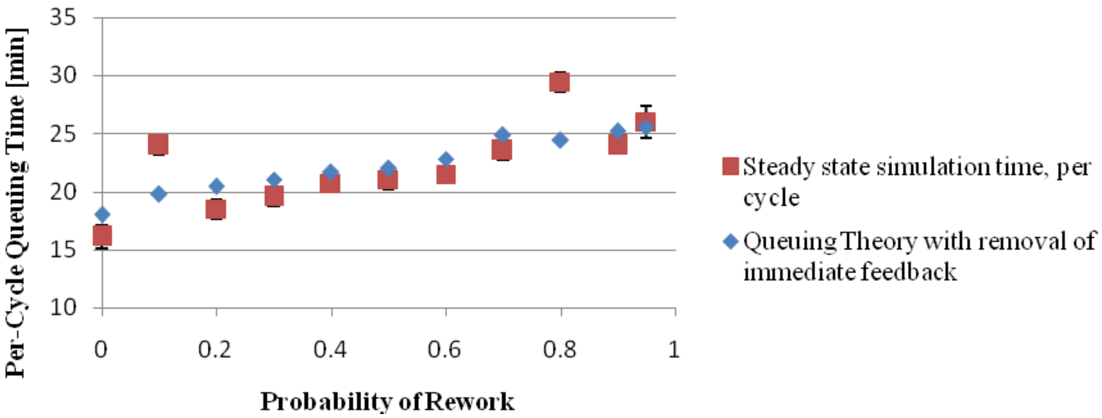


Figure 6.9: Comparison of per-cycle queuing time for queuing calculaitons and simulation for Hyperexponential arrival distribution, SCV=2.25

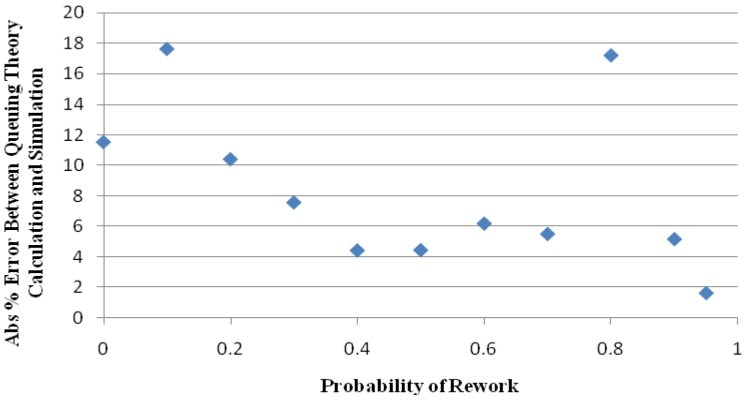


Figure 6.10: Error for per-entity queue time for Hyperexponential arrival distribution, SCV=2.25

Table 6.3: Simulation run time and number of replications, Hyperexponential arrival distribution, SCV=2.25

Probability of Rework	Half Width, per cycle [min]	Run Time [min]	# Replications
0	0.9885	172800	5
0.1	0.8214	345600	10
0.2	0.8372	230400	10
0.3	0.7731	288000	10
0.4	0.3939	1152000	10
0.5	0.8645	345600	10
0.6	0.546	230400	10
0.7	0.8378	460800	10
0.8	0.8256	1152000	25
0.9	0.6689	1152000	25
0.95	1.368	1152000	25

Again, the accuracy of the per-cycle queuing time translates to the per-entity time as seen in Figure 6.11 and 6.12. Given the highly variable external arrival pattern, the Hypergeometric case proved to be the most cumbersome in terms of resolving the simulation to the desired confidence interval. This can be observed in Table 6.3, especially in the final cases of .8 to .95 probability of rework where the return on adding replications diminishes greatly.

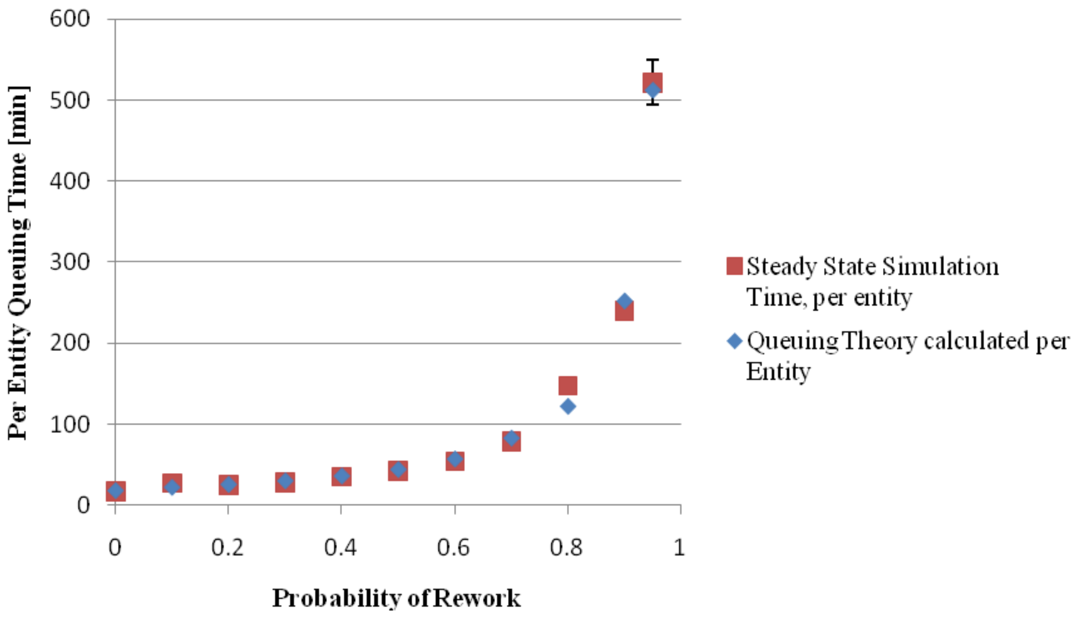


Figure 6.11: Average queuing time per-entity for Hyperexponential arrival distribution, SCV=2.25

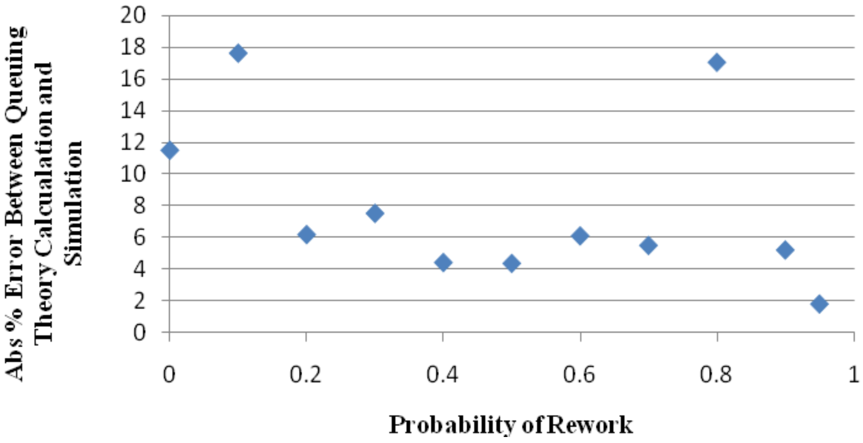


Figure 6.12: Percent error for per-entity queue time for Hyperexponential arrival distribution, SCV=2.25

In order to capture the effects of only the external arrival distribution, Figure 6.13 shows the average percent error aross all rework rates. From this plot, it is clear that the exponential arrival pattern yields the most accurate queing calculations, wheras the cases of low and high arrival variability each show reduced accuracy of approximately the same magnitude.

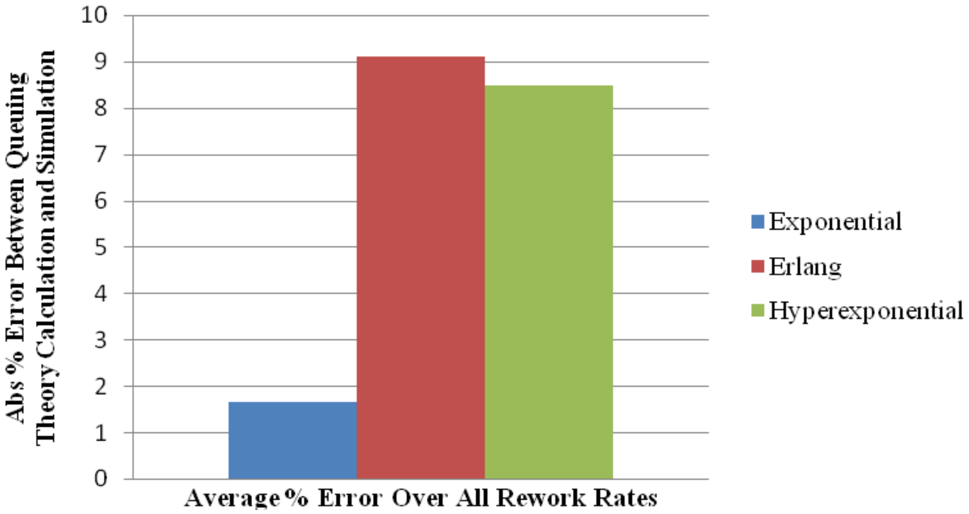


Figure 6.13: Average of % error for all rework rates according to external arrival distribution

6.2 Observed Error for rework and breakdown case

Finally the performance of the queuing model over ranges of rework rate and subject to machine breakdowns is analyzed. Figure 6.14 shows that given the right run time and number of replications, the percent error is quite good, even at high rework rates and 50% machine availability. In fact the highest observed percent error occurs on the 0% rework scenario at 12% error. Figure 6.15 demonstrates to error for each scenario having machine availability of 50%, with mean time to failure and mean time to repair both equal to 1 minute. It would appear that there is a downward trend in the % error as rework increases. Tighter tolerances in the confidence interval cannot explain this as the 95% confidence interval does not show a decreasing trend with rework rate. Table 6.4 shown the run times and number of replications needed to obtain these data points.

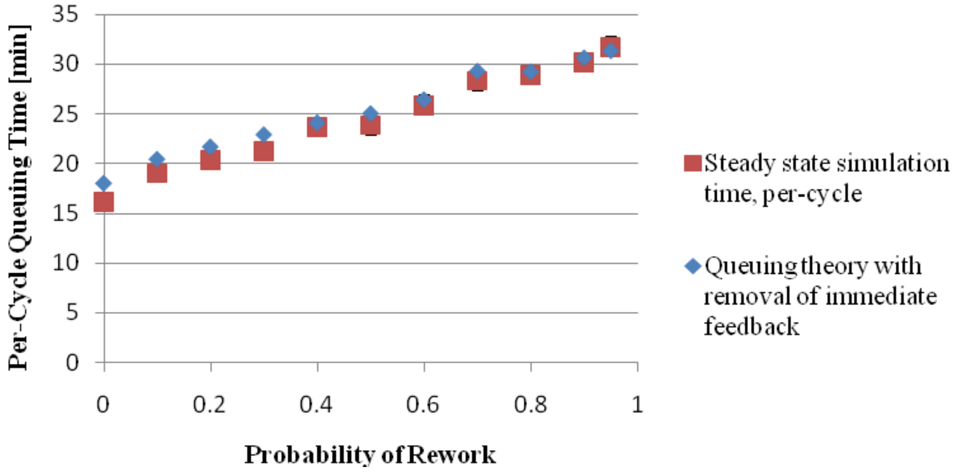


Figure 6.14: Comparison of per-cycle queuing time for queuing calculaitons and simulation for Availability=.5, exponential arrival distribution, SCV=1

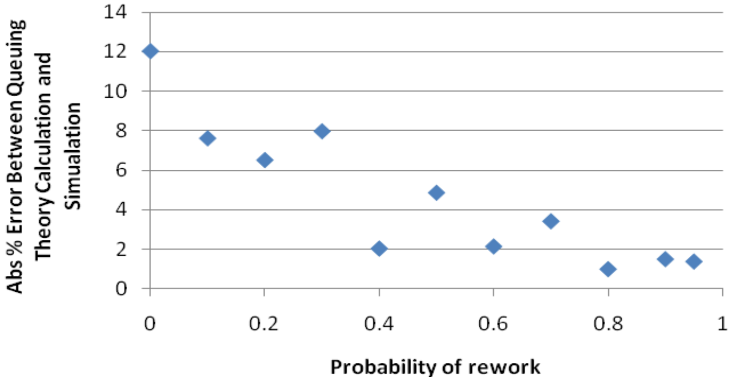


Figure 6.15: Error for per-cycle queue time for availability=.5, Exponential arrival distribution, SCV=1

7 CONCLUSIONS AND FUTURE WORK

7.1 Conclusions: Recap of research questions

Here the initial research discussion questions are reviewed, the first of which asked the advantage of using queuing theory to model system performance. From the test cases and results, it was seen that queuing models using the node decomposition technique with removal of immediate feedback can provide good results when compared to simulation observations. In nearly all cases 10% error or less was observed. This includes test cases with rework rate ranging from 0% to 95%, varying levels of arrival variability, and varying levels of machine availability. In short, the queuing model can provide a good estimate of steady state queuing time. The queuing model offers a quicker result than a discrete event simulation model which would prove especially useful if several scenarios need to be tested in a short time. Furthermore, the queuing model incorporates the random nature of the system as opposed to more simplified optimization models which typically require estimates for deterministic processing times.

The second research question asks if the proposed queuing model provides an accurate representation of the average waiting time per part in a system that may be subject to high levels of rework. This was tested meticulously using the Arena DES model and the result showed no significant relationship between the rework rate and the accuracy of the queuing model. Specifically, the model does not show deteriorating accuracy as the probability of rework increases, as was the initial suspicion. As a follow up, the third question asks if more refined models would be needed to improve accuracy in any specific circumstances. The more significant response here may be as to the relevance of the proposed steady state value for average queuing time. Of course, the steady state value represents the waiting time that would be expected on average over a long period of time, without changing conditions. That is, the external arrival distribution should remain constant, the processing time distribution should remain constant, the rework rate should not change, and the machine availability should be consistent. In cases where the rework rate is quite high (above 50%) and where the machine availability is quite low (50%) the simulation suggests that it may take the system several hundred hours to reach a steady state with desirably low confidence interval. In reality, no system will perform for such a long time without being subjected to change (worker learning, machine deterioration, etc) and this may be the primary reason for the limited application of queuing theory. For the cases studied, however, the proposed method is sufficient.

The final two research questions are with regard to the relationship between system parameters and the accuracy of the queuing model, and any generalizations that might be made with regards to the use of queuing theory in optimization problems. As stated previously, no clear relationship emerged between the rework rate and machine availability with regards to percent error of the queuing model. Still, if the results of this model are to be integrated into an optimization model, it should be noted that if a real system is being monitored, if the system is highly variable it is likely that some observed wait times will skew the average away from the expected steady state. Only over extended observations would the effects of these extreme results be rectified. Other relevant information would be to look at the standard deviation about the expected steady state value.

7.2 Future Work

Valuable additions to this work would include integration of the result with an optimization model designed to schedule jobs to work stations with varying rework rates depending on the job type. Some of this work was completed in the formulation of a sample problem of this type (Brown 2011) and this should be continued with a detailed analysis of the accuracy of the predictions in light of this work and the results of the optimization. Secondly, the accuracy of the outlined technique should be studied for more complex networks, such as those that occur in job shops. The ability of the queuing methods to provide useful information for systems subject to both human and machine constraints is imperative. Finally, incorporation of these concepts with those of manufacturing cell formation, wherein the skill of the workers and the capability of the machines assigned to a given cell would help in their initial designs. In conclusion, the methods outlined in this work provide a strong background for future inquiry into the expanded application of queuing theory in manufacturing. The results show that a relatively simple and fast-running model can provide good results in cases where realistic conditions apply.

Appendix A: Table of Notations

effective processing time	t_e
natural processing time	t_0
average setup time	t_s
number of parts between setups	N_s
effective variance of processing time	σ_e^2
natural variance of processing time	σ_0^2
variance of setup time	σ_s^2
probability of job being routed from process i to process j	p_{ij}
squared coefficient of variation	SCV or c^2
standard deviation	σ
Mean	μ
natural coefficient of variation	c_0
machine availability	A
mean time to failure	m_f
mean time to repair	m_r
variance of repair time	σ_r^2
squared coefficient of variation of processing time	c_e^2
probability of rework	p
squared coefficient of variation of interarrival times	c_a^2
Utilization	u
expected wait time	EW
queue cycle time	CT_q

REFERENCES

Anderson, K., 1987, A method for planning analysis and design simulation of CIM systems, Proceedings of the 1987 Winter Simulation Conference

Andrews, B., Parsons, H., 1993, Establishing telephone agent staffing levels through economic optimization, Interfaces 23(2): 14-20

Bobrowski, P., Park, P., 1993, An evaluation of labor assignment rules when workers are not perfectly interchangeable, Journal of Operations Management (11): 257-268

Brown, A. Badurdeen, F., 2011, Optimization of queuing theory wait time through multi-skilled worker assignments, Industrial Engineering and Engineering Management (IEEM), 2011 IEEE International Conference on, pp.1416-1420, 6-9 Dec. 2011

Chinnaswamy, M., 2005, Aggregation approaches for incorporating e-mail processing history in queuing models of customer contact centers, Master's thesis, Oklahoma St. University

Crowley, D., Bard, J., Jensen, P., 1995, Using flow ratio analysis and discrete event simulation to design a medium volume production facility, Computers and Engineering 28(2): 379-397

de Treville S, Van Ackere A, 2006, Equipping students to reduce lead times: The role of queuing-theory-based modeling. Interfaces 36(2):165–173

Erlang, A., 1909, The theory of probabilities and telephone conversations, Nyt Tidsskrift for Matematik 20: 131-137

Erlang, A., 1917, Solution of some problems in the theory of probabilities of significance in automatic telephone exchanges, Elektroteknikeren 13:138-155

Gordon, W., Newell, G., 1967, Closed queuing systems with exponential servers, Operations Research 15(2): 254-265

Hopp, J., Spearman, M., 2001, Factory physics: foundation of manufacturing management, 2nd Edition, McGraw-Hill, New York, NY.

Hu, M, Chang, S., 2003, Translating overall production goals into distributed flow control parameters for semiconductor manufacturing, Journal of Manufacturing Systems 22(1): 46-63

Huq, F., Cutright, K., Martin, C., 2003, Employee scheduling and makespan minimization in a flow shop with multi-processor work stations: a case study, Omega 32: 121-129

Jackman, J, Johnson, E., 1993, The Role of Queueing Network Models in Performance Evaluation of Manufacturing Systems. The Journal of the Operational Research Society, 44(8): 797-807

Jackson, J., 1954, Queuing Systems with phase type service, Operational Research Society 5(4): 109-120

Jackson, J., 1963, Jobshop-Like queuing systems, Management Science 10(1): 131-142

Jahangirian, M., Eldabi, T., Naseer, A., Stergioulas, L., Young, T., 2010, Simulation in manufacturing and business: a review, European Journal of Operational Research 203: 1-13

Juran and Schruben 2004: Using worker personality and demographic information to improve system performance prediction, Journal of Operations Management 22(4): 355-367

Kelton, W., Sadowski, R, Swets, N., 2010, Simulation with Arena, 5th Edition, McGraw-Hill, New York, NY

Keuhn, P., 1979, Approximate analysis of general queuing networks by decomposition, IEEE Transaction on Communications 27(1): 113-126

Kim, S., Muralidharan, R., O'Cinneide, C., Taking Account of Correlations Between Streams
in Queueing Network Approximation. Queueing Systems 49: 261-281

Kuo, Y., Yang, T., 2007, Optimization of mixed-skill multi-line operator allocation problem, Computers & Industrial Engineering 53: 386-393

McCreery, J., Krajewski, L., Leong, G., Ward, P., 2004, Performance implications of assembly work teams, Journal of Operations Management 22: 387-412

Mitchell, T., 1982, Motivation: New Directions for Theory, Research, and Practice. The Academy of Management Review 7(1): 80-88

Niemi, E., 2009, Worker allocation in make-to-order assembly cells, Robotics and Computer-Integrated Manufacturing 25: 932-936

Norman, B., Tharmmaphornapilas, W., Needy, K., Bidanda, B., Warner, C., Worker assignment in cellular manufacturing considering technical and human skills, 2010, International Journal of Production Research 40(6): 1479-1492

Pentico, D., 2007, Assignment problems: A golden anniversary survey, European Journal of Operational Research (176): 774-793

Pradhan, S., Damodaran, P., 2009, Performace characterization of complex manufacturing systems with general distributions and job failures, European Journal of Operational Research 197: 588-598

Rabta, B., 2009, A review of decomposition methods for open queuing methods in Rapid Modeling for Increased Competitiveness: Tools and Mindset, G Reiner, Ed., New York: Springer

Rajagopalan, S., Yue, H., 2001, Capacity planning with congestion effects, European Journal of Operational Research 134: 365-377

Rao, S, Gunaskearan, A., Goyal, S.K., Marikainen, T., 1998, Waiting line model applications in manufacturing, International Journal of Production Economics 54: 1-28

Rother, M., Shook, J., 1998, Learning to see: value stream mapping to add value and eliminate muda. The Lean Enterprise Institute, Brookline, MA.

Slomp, J., Bokhorst, J., Molleman, E., 2005, Cross-training in a cellular manufacturing environment, Computers & Industrial Engineering 48: 609-624

Solberg, J., 1977, A mathematical model of computerized manufacturing systems, Proceedings of the 4th International Conference on Production Research, Tokyo, Japan: 1265-1275

Stratman, J., Roth, A., Gilland, W., 2004, The deployment of temporary production workers in assembly operations: a case study of the hidden costs of learning and forgetting, Journal of Operations Management 21: 689-707

Suri, R., Diehl, G., de Treville, S., Tomsicek, M., 1995, From CAN-Q to MPX: Evolution of queuing software for manufacturing

Suri, R., 1998, Quick response manufacturing: a companywide approach to reducing lead times. Productivity Press, Portland, OR.

Takacs, L, 1962, A Single Server Queue with Feedback. The Bell System Technical Journal: 505-519

Tiwari, V., Patterson, J., Mabert, V., 2009, Scheduling projects with heterogeneous resources to meet time and quality objectives, European Journal of Operational Research 193: 780-790

Whitt, W., 1980, Approximating a point process by a renewal process, I: Two Basic Methods. Operations Research 30(1): 125-147

Whitt, W., 1983, The Queuing Network Analyzer. The Bell System Technical Journal 62(9): 2779-2815

Womack, J., Jones, D., 2003, Lean thinking: Banish waste and create wealth in your corporation, Free Press

Yue, H., Slomp, J., Molleman, E., Van der Zee, D.J., 2008, Worker flexibility in a parallel dual resource constrained job shop. International Journal of Production Research 46(2): 451-467

VITA

Place of Birth

Princeton, KY, USA

Education

Bachelors of Science in Mechanical Engineering
University of Kentucky, USA 2008

Professional Positions

Co-op (Separate rotations Fall, Spring, Summer 2005-2007)
GE Aviation: Madisonville, KY

Research Assistant (2010-Present)
University of Kentucky Institute for Sustainable Manufacturing: Lexington, KY

Publications

Brown, A.; Badurdeen, F.; "Optimization of queuing theory wait time through multi-skilled worker assignments," *Industrial Engineering and Engineering Management (IEEM), 2011 IEEE International Conference on,* pp.1416-1420, 6-9 Dec. 2011

Badurdeen F., Shuaib M., Wijekoon K., Brown A., Faulkner W., Amundson J., Jawahir I. S., Goldsby T., Iyengar D., Boden B. (2012) Quantitative Modeling and Analysis of Supply Chain Risks Using Bayesian Theory, submitted to Journal of Manufacturing Technology Management

Badurdeen F., Brown A., Faulkner W., Amundson J., Goldsby T., Boden B. (2012) Assesssment of supply chain risk in make and buy scenarios using Bayesian Belief Networks, in progress to be submitted to Supply Chain Management: An International Journal