UNIVERSITY OF CALGARY

Input Control for the M/M/1 System via Judicious Order Rejection and Order Release

by

Yannai Zev Romer Segal

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE

DEGREE OF MASTER OF SCIENCE

DEPARTMENT OF MECHANICAL AND MANUFACTURING ENGINEERING

CALGARY, ALBERTA

MAY, 2003

UNIVERSITY OF CALGARY

FACULTY OF GRADUATE STUDIES

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies for acceptance, a thesis entitled "Input Control for the M/M/1 System via Judicious Order Rejection and Order Release" submitted by Yannai Zev Romer Segal in partial fulfilment of the requirements of the degree of Master of Science.

Paul Rog

Supervisor, Dr. P. Rogers, Department of Mechanical and Manufacturing Engineering

Dr. S.T. Enns, Department of Mechanical and Manufacturing Engineering

Dr. D.P. Bischak, Haskayne School of Business

17 June 2003

Date

Abstract

Input Control via the Order Review and Release (ORR) mechanism is a known method of improving make-to-order manufacturing performance. A broad view of the ORR mechanism includes two components, the first concerned with accepting/rejecting arriving jobs (order review), and the second with determining when to release jobs (order release). The goal of this research is to explore the fundamental behaviour of the order review and order release components using a simple testbed system and a parametric profit model. By better understanding the role each component plays in improving the performance of the simple test system, valuable insights into the generic use of ORR are obtained. The order release component is shown to enable reduction in earliness costs, while judicious order rejection is shown to permit excessive tardiness costs to be avoided. The combination of both components is shown to outperform the best of the individual components under most conditions.

Acknowledgements

My thesis was completed with the help and support of many people whom I wish to acknowledge here.

I would like to thank Dr. Paul Rogers for his supervision and direction, and for his assistance in securing financial support for my studies. I would also like to thank him for his excellent undergraduate teaching, which directed me towards operations research and a graduate degree in Manufacturing Systems.

I would also like to thank my examining committee – Dr. Paul Rogers, Dr. Van Enns, and Dr. Diane Bischak.

My wonderful wife, Marina, deserves my utmost gratitude and appreciation for her love and support throughout my program.

Finally, a special thanks the Natural Sciences and Engineering Research Council of Canada, to SMED International, and to the Department of Mechanical and Manufacturing Engineering for their financial support which has made this research possible.

Dedicated to my Family

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Managing a manufacturing system in a competitive environment requires continuous improvement in productivity, quality and cost. In recent years, the focus of manufacturing improvement has shifted to reducing inventories and work-in-process (WIP), shortening lead times, and improving delivery performance.

For over thirty years, "input control" has been recognized as a key component of achieving these performance improvements. The concepts of input control have been most often applied to make-to-order manufacturing systems in the context of the Order Review and Release (ORR) mechanism, which manages the transition of orders from arrival at the planning system, through to the shop floor. The ORR mechanism consists of the careful review of arriving jobs, and the holding of accepted jobs in a pre-release pool, releasing them to the manufacturing floor only when necessary. A broad view of the ORR mechanism includes two components, the first concerned with accepting/rejecting arriving jobs (order review), and the second with determining when to release jobs (order release). The bulk of ORR research has focused on the order release component of the mechanism, with relatively little attention paid to order review via the judicious acceptance/rejection of orders.

The primary criticism of ORR within the literature is that while the use of a pre-release pool and careful order release has been shown to reduce the time spent by jobs on the shop floor (and thus the levels of WIP on the floor), the total *customer* lead time is not decreased, because of the time spent in the pool. Researchers have shown that control of input variance is required for the order release function to be truly effective. Acceptance/rejection of jobs in the context of order review has been suggested as an effective form of input variance control, but this area has been largely ignored in the literature.

A major problem with the state of ORR research is that the literature is too focused on experiment-based testing of specific release rules under specific conditions. There is a significant lack of general insight and the conclusions of authors are rarely generically applicable. Practitioners have identified the need for more generally insightful investigations of the ORR mechanism.

The present thesis is intended to address these two criticisms of ORR. The goal of this research is to explore the fundamental behaviour of the order review and order release components using a simple testbed system. By better understanding the role each component plays in improving the performance of the simple test system, valuable insights into the generic use of ORR can be obtained.

The remainder of the thesis is arranged in the following manner. Chapter 2 presents a literature review of input control and the ORR mechanism, and further highlights the objectives of the research. Chapter 3 describes the manufacturing system modelled and the performance measures considered, and also presents analytical derivations and numerical results for the uncontrolled system. Chapter 4 presents analytical derivations and numerical results for two analytically tractable control policies. Chapter 5 presents experimental results for two intractable control policies explored using discrete-event simulation. Finally, Chapter 6 concludes this thesis by highlighting the main contributions of the work and identifying some suggested directions for future work.

Chapter 2

Review of Relevant Literature

This chapter contains a review of relevant literature focusing on the area of input control, and on the Order Review and Release (ORR) mechanism through which it is applied. Criticisms of ORR and of the state of ORR research are presented, which serve to provide the motivation for the current research. The chapter concludes with a summary of specific objectives for the research reported in the thesis.

2.1 Input Control via Order Review and Release

2.1.1 Input Control

Common problems facing manufacturing facilities include high levels of both work in process (WIP) and finished goods inventory, excessive expediting, and production plans regularly falling behind schedule resulting in missed delivery dates. In his seminal paper, Wight (1970) identifies these problems as frequently being caused by what he referred to as long Manufacturing Lead Time (MLT). Wight shows that the actual time a job spends being processed typically accounts for less than ten percent of its time on the shop floor. Most of a job's time is spent waiting in queues for the opportunity to be processed by equipment currently occupied by a job of higher priority. Wight identifies the long queuing times seen in most plants as having three causes: (i) lead time inflation; (ii) erratic plant input; (iii) the inability to plan and control output rates effectively. The three causes are further described in the following paragraph.

Lead time inflation is caused by the misconception that longer lead times will result in a greater likelihood of meeting customer due dates. Instead, longer lead times result in an increase in work on the shop floor, causing congestion and lengthening queue times. Releasing jobs to the shop floor as soon as the system generates them usually results in highly erratic input to the shop. Because of the relatively fixed capacity of most resources, periods in which demand is higher than capacity will result in a large number of tardy orders. Additionally, the carryover of uncompleted jobs to future periods will have undesirable effects as these jobs will interfere with the priorities and schedules of future work. The relatively fixed output capacity of most manufacturing resources is a symptom of the general inability to control output in most manufacturing environments. Outside of the ability to run overtime, there is little that most manufacturing managers can do to alter short-term capacity. Hiring and training (or eliminating) workers or acquiring and installing new equipment can involve considerable costs, and require longer time horizons than typical load forecasts offer. Altering output capacity is therefore not an appropriate method of addressing short-term variation in demand.

Wight proposes one simple rule for eliminating these three problems – ensure that "the input to a shop be equal or less than the output". This simple rule requires carefully reviewing planned orders and only releasing accepted orders to the floor when the time is right.

Wight's principle of Workload Control (WLC), also called input/output control, has attracted much interest in the research community. The majority of work in this area has been devoted to manufacturing systems where output capacity remains constant in the short term, and is thus referred to only as input control.

2.1.2 The Order Review and Release Mechanism

Input control, when applied to make-to-order (MTO) production systems, has typically been studied in the context of the ORR mechanism. In MTO systems production can only begin after a customer's order is placed, as opposed to make-to-stock (MTS) systems, where customer orders are (ideally) instantly filled from existing stocks, and production orders are only used to replenish those stocks. Thus, for MTO systems, a primary goal is the completion of jobs within the lead time promised to the customer. Additionally, an MTO manufacturer can usually gain a competitive advantage by reducing its lead time in relation to competitors (provided, of course, that this does not reduce the ability to meet the lead time).

The ORR mechanism is the method by which the system: (i) carefully reviews arriving jobs to ensure that they have a "good" chance of being completed on time; (ii) releases them to the manufacturing floor at the appropriate time. Thus, ORR consists of the group of activities taking place between the initial customer request until the job, if accepted, is released to the floor. According to Melnyk and Carter (1987), ORR is one of the five major components of the Production Activity Control (PAC) system, in addition to detailed scheduling, data collection and monitoring, control and feedback, and order disposition. Figure 2.1 shows a diagram of the ORR mechanism.

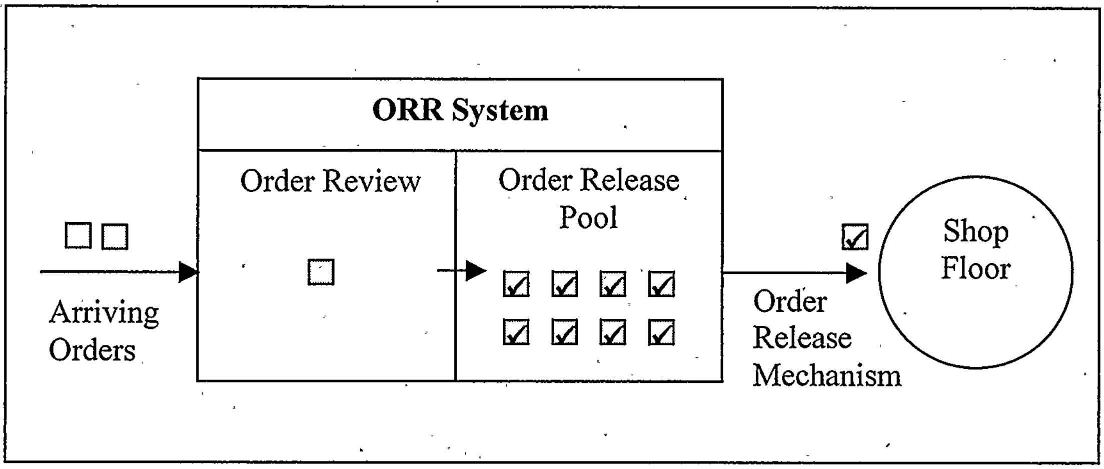


Figure 2.1: The ORR Mechanism

Two primary frameworks for ORR can be found in the literature, the first developed by Melnyk and Carter (1987), and the second by Bechte (1988). Melnyk and Carter view ORR as consisting of three major activities: (i) order preparation; (ii) review and evaluation of orders; (iii) load levelling. Bechte views the three main parts of ORR as: (i) order entry phase; (ii) pre-shop pool management phase; (iii) order release phase. These frameworks are highly similar in function.

Melnyk and Ragatz (1988) identify four major components of ORR functionality: (i) the order release pool; (ii) the shop floor; (iii) the planning system; (iv) the information system.

The order release pool holds all jobs which have been accepted by the planning system, but have not yet been released to the shop floor. Exit from the pool is governed by a triggering mechanism, which decides when to release an order, and an order selection rule, which decides what order(s) to release. Triggering mechanisms may operate under a continuous or bucketed (periodic review) timing convention. The triggering mechanism may be pool-based, shop-based, or pool and shop-based. Pool-based triggering mechanisms are dependant only on information about the jobs in the pool, shop-based mechanisms are based only on information about jobs on the shop floor, and pool and shop-based mechanisms use information on both. The order selection rule can likewise be local (only based on information on jobs in the pool) or global (also based on information on the shop floor[1]).

ORR attempts to balance the release of work to the shop floor against the available capacity on the shop floor. Typically, it is not the remaining capacity, but rather the current load that is monitored by the ORR system. Information about the load on the shop can either be expressed individually for each workcentre, individually for select

[1] The selection rule cannot be solely based on information on the shop floor, as it is required to have at least minimal knowledge of the pool to select a job from it.

workcentres (usually known bottlenecks), or aggregated as total shop load. The ORR system may monitor instantaneous load, or the planned load profile over a given time horizon.

The planning system is an important component of ORR because planned orders represent future shop loads. Jobs still in the planning system may or may not be visible to the ORR mechanism. Schedule visibility helps the ORR mechanism make predictions of future demands on the shop floor by examining planned orders. The period-to-period feasibility of plans generated by the planning system is also of importance to the ORR mechanism, and this feasibility may be controlled or uncontrolled by the planning system.

The information system is the final major component of the ORR mechanism, providing data on the state of the pre-release pool and/or the shop floor as discussed above. The timeliness, accuracy and completeness of this information may have a significant impact on ORR functionality. Timeliness refers to the speed at which changes are reflected in the information available to the ORR mechanism. Accuracy may be compromised by measurement or data-entry errors, and completeness reflects the extent to which generally available data may occasionally be missing.

2.2 Brief Survey of ORR Research

2.2.1 General ORR Research

There are a large number of research papers consisting of general discussions of ORR and surveys of the literature. As discussed earlier, Melnyk and Carter (1987), Bechte (1988), and Melnyk and Ragatz (1988) provide early frameworks for input control via ORR. This framework has been re-examined and expanded in subsequent literature by Melnk and Ragatz (1989) and more recently in Bergamaschi *et al.* (1997).

Wisner (1995) and Bergamaschi *et al.* (1997) provide reviews of existing ORR literature. Wisner (1995) reviews descriptive research (general discussion, case study and industry surveys), analytical research, and simulation-based research. Simulation-based research is classified by routing type, number of workcentres and worker resources, release rules used, performance criteria, type of statistical analysis, and the real/hypothetical nature of system characteristics. Bergamaschi *et al.* (1997) review the descriptive research, and classify experimental research on the basis of an expanded version of the framework originally introduced by Melnyk and Ragatz (1988, 1989).

The bulk of ORR literature is in the form of simulation-based exploration of order release rules (triggering and selection) in small shop environments. Some research compares performance of different rules on the same system, while some explores the effect of other system characteristics (such as priority dispatch rules, due date setting, shop size and routing type). The literature reviews listed above provide thorough summaries of most of these works.

2.2.2 Criticisms of ORR

Research into the order-release component of the ORR framework has found that controlled release can definitely result in significant reductions in shop congestion and manufacturing flowtime. However, critics of ORR, such as Bertrand (1983a, 1983b), show that ORR may increase overall flowtime because of excessive delays in the pre-release pool. Even proponents of workload control, such as Melnyk and Ragatz (1989), have found that the reduction in manufacturing flowtime may be more than offset by the time spent in the order release pool. Therefore, while order release strategies may result in smoother operations in the shop due to the anticipated effects of decreased congestion, they may not reduce the lead time that can be promised to customers, and therefore might not result in a direct competitive advantage. Additionally, Baker (1984) notes that reduced congestion on the floor may make scheduling and dispatching less effective. His results for a single-server system show that under some conditions selective order release may not be advantageous because it degrades the performance of certain dispatching rules.

These criticism are addressed by proponents of ORR, who conclude that order release is an effective technique when combined with variance control at the planning and shop floor level. Fredendall and Melnyk (1995) show that variance in the rate at which work content is received from the planning system is still the main cause of cost[2] variances, even when an ORR system is in place. The authors show that the delay-and-release function of the pre-release pool serves only as a secondary mode of variance control, and that the planning system is required to serve as the primary source of variance control. The authors show that ORR functions properly only when input variance is controlled by the planning system, and confirm Melnyk *et al.* (1992) who show that the performance of simple dispatching rules is far better when input variance is controlled.

A major criticism of the current state of ORR research, is that the focus of most literature is on testing specific methods in particular situations. Conclusions are specific to the combination of methods and systems tested, and there is a significant lack of generic insight. Gaalman and Perona (2002) note in their introduction to a special issue of *Production Planning and Control* focused on workload control in job shops:

> Though many Workload Control methods are presented in the literature, relatively little is known about their performance in specific production contexts. The large number of aspects that are necessary to describe a particular situation and which make generic conclusions difficult to apply can explain this. From a practical perspective, the need for structural insights in the performance of the methods is quite large.

[2] The authors use a cost model that includes tardiness costs, WIP costs, and worker transfer costs (a penalty for moving a worker between workcentres), but self-admittedly select a ratio of parameter values that place, by far, the greatest priority on minimizing tardiness.

Cigolini and Portioli-Staudacher (2002) reiterate this message elsewhere in the same special issue:

> Anyway, ORR techniques are various and the literature shows quite a difference in performance among the different techniques. Moreover, until now no research has proven an ORR technique to outperform all others, and very little research is available about the strengths and weaknesses of each technique. Common questions arising when literature about ORR is analysed, refer to why a specific policy performs better than another and why some research results contrast with others.

The overly-specific nature of ORR research, and the lack of generally applicable insights found in the literature may be a barrier to the implementation of ORR in industrial systems.

2.2.3 The Accept/Reject Decision as Part of ORR

While the majority of the ORR literature focuses on order release, order review is also an important component of input control, particularly in the context of variance reduction. Control of input variance by selectively rejecting jobs is commonly found in the queuing literature, and is enumerated as one of five input control methods in a survey of queuing literature conducted by Crabill *et al.* (1977)[3]. Bergamaschi *et al.* (1997) place the accept/reject decision in the context of the order review phase of ORR, as shown in

[3] The others being: (i) directly affecting the arrival rate; (ii) indirectly affecting the arrival rate (i.e. through pricing policy); (iii) encouraging customers to behave in a socially optimal manner; (iv) "closing down" the system.

Figure 2.2. However, the accept/reject decision has not been extensively explored in this context. Philipoom and Fry (1992) are the first to relax the assumption that all orders must be accepted. The authors experiment with rejection based on (i) the aggregate shop load; (ii) the load on the workcentres on the arriving job's routing. A work limit is set for the system in the case of (i), or for individual workcentres in the case of (ii). The authors conclude that selective rejection of a small percentage of the arriving work can result in dramatic improvements in shop performance. The authors also find that workcentre-based rejection is better than rejection based on aggregate shop load.

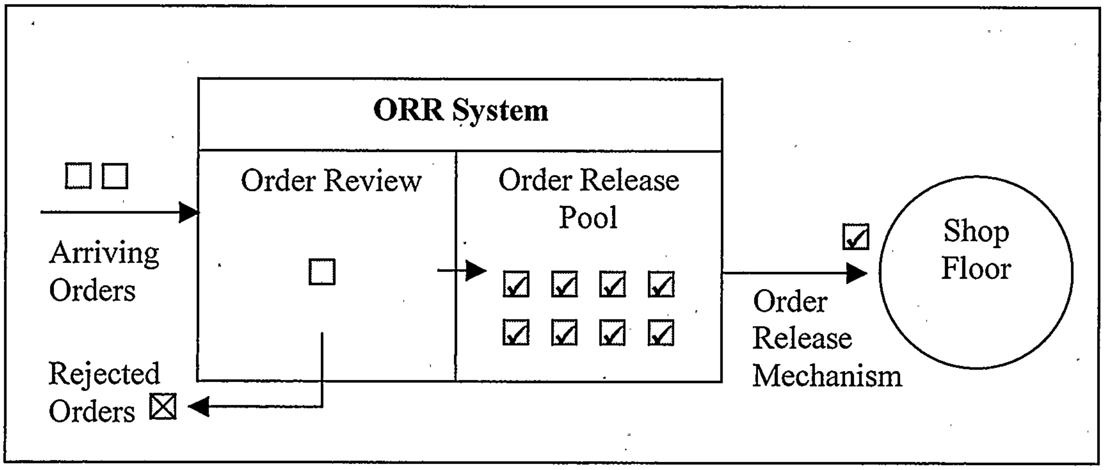


Figure 2.2: The ORR Mechanism with Rejection

Nandi (2000) builds on the work of Philipoom and Fry, addressing a manufacturing system with two classes of jobs ("urgent" and "regular"), and exploring more complex accept/reject rules. The author finds that input control through judicious order rejection can result in significant performance improvements for jobs that are accepted. Rogers

(2002) and Rogers and Segal (2003) build on Nandi's work, finding that more complex rejection rules are capable of further performance improvements.

The accept/reject decision is also studied outside the context of the ORR framework. There is a substantial body of early queuing theory work involving the accept/reject decision in the context of multiple job classes in simple service systems. For example, Scott (1969, 1970) analyzes a two-class single-server queuing system limit such that when the queue length is at or beyond some limit, lower priority jobs are rejected by the system. Miller (1969) and Lippman and Ross (1971) explore the case of multiple job classes of differing value, and no backlog permitted, for multiple- and single-server systems. Wester *et al.* (1992) consider the accept/reject decision in the context of sequence-dependent setup times for a single-server system. The authors find that by intelligent rejection of jobs based on setup time considerations, performance in a high-load system can be improved. This work is expanded by ten Kate (1994) who shows that under high-load, low lead-time conditions, selective rejection based on sequence-dependent setup time considerations results in improved performance in a more complex system. While this research explores order review and the accept/reject decision, none of it is in the context of ORR, where accepted jobs are then subject to an order release mechanism (beyond immediate release).

2.3 Motivation and Objectives for the Present Research

The present research is motivated by the clearly identified need for increased knowledge of the fundamental functions of input control via ORR. Additionally, there is a need to examine the role of the accept/reject decision in the context of order review (and variance reduction) within the ORR framework.

The objectives of the present research are as follows:

(i) To explore the fundamental functionality of ORR using a simple testbed system that will provide generic insights into the benefits of ORR. The well known M/M/1 queuing system will be used (i.e. single server system with exponentially distributed interarrival and service).

(ii) To explore the order release mechanism and the accept/reject decision independently, and in combination, to better understand the circumstances under which each component can be of benefit. Simple analytically-tractable abstractions of each of the release and reject mechanisms will be investigated, and their effect on the system will be determined. Discrete-event simulation will be used to model and explore more complicated (intractable) combinations of the two components.

(iii) To perform the above analysis under a range of environmental factors. A cost/profit model will be used that considers the various aspects of system performance affected by the ORR mechanism. The exogenous environmental parameters to be considered are: (i) the shop load; (i) the flow allowance; (ii) the severity of tardiness costs; (iv) the severity of earliness costs. Analysis will occur under multiple combinations of cost parameters and flow allowance, and under a wide range of shop loads, so that the effect of these parameters on control policy performance can be evaluated.

Chapter 3

Test System

The previous chapter presented the motivation for, and objectives of, this research. This chapter describes the system being modelled and the performance measures that will be considered relevant. The performance of the uncontrolled system is presented in this chapter, with candidate control policies presented and evaluated in the following two chapters.

3.1 A Hypothetical Manufacturing System

For this research, a hypothetical manufacturing system is being used as a testbed to explore various control policies under various operating conditions. This section describes the basic operating characteristics as well as the financial performance measures relevant to optimal system control.

3.1.1 Basic Operating Characteristics

The test system is a single-server M/M/1 queuing system, characterized by the following:

(a) Interarrival times are exponentially distributed with a mean interarrival time of $1/\lambda$ (and mean arrival rate of λ).

(b) Service times are exponentially distributed with a mean service time of $1/\mu$ (and mean service rate of μ).

(c) There is a fixed flow allowance M, such that an arriving job is assigned a due date M time units from its arrival time. We can also express the flow allowance in normalized form (m) in terms of a multiple of the mean service time (m=$M\ \mu$).

(d) The profit contribution (before costs) for each completed job is R (currency units). In the remainder of this thesis this will be referred to as job revenue.

(e) Jobs are processed on a first-come, first-served basis.

The system operates under ideal conditions with no breakdowns, resource unavailability, or similar disruptions.

3.1.2 System Costs

We are interested in constructing a cost/profit model that is capable of capturing the benefits and limitations of the various combinations of order release rules, order review rules, and environmental factors to be tested. The literature shows a great variety of cost and profit models used in the analysis of manufacturing systems. Enns (1995) lists the cost categories for a manufacturing system as: (i) variable production costs; (ii) WIP holding costs; (iii) lead time costs; (iv) due-date deviation costs; (v) fixed overhead costs.

For the system under consideration variable production costs and fixed overhead are not affected by the ORR mechanism, and are not considered. Likewise lead time costs (the costs of having an uncompetitive lead time) are not considered, as it is assumed that the firm is operating with a competitive lead time, and the flow allowance (*M*) is set accordingly. Rejection of arriving jobs may be considered as being an indirect lead time cost, but we will assume no costs to rejecting incoming jobs beyond the loss of potential revenue, as per Nandi (2000). Due-date deviation costs will be modelled, as the primary functionality of ORR is to improve the due-date performance of the manufacturing system. Due-date deviation costs are incurred for both tardiness and earliness of jobs. We assume that for a single-server system WIP costs for items in process are equal to the cost of holding the equivalent raw material in inventory, and can therefore be ignored. Note that holding costs for finished goods inventory are included under the earliness cost component of due-date deviation costs.

The modelling of tardiness costs and earliness costs (or the equivalent holding costs) as linear with time is generally accepted in the literature. While some advocate for the use of non-linear cost functions, a large number of practitioners use linear functions to model these costs. We will assume that these costs accrue linearly with time.

Tardiness costs are incurred when a job is completed after its promised due date. We will model tardiness costs as increasing linearly with job tardiness. In reality, tardiness costs may be due to: (i) contract-specific penalties; (ii) loss of customer goodwill and company reputation; (iii) expediting costs (i.e. faster, more expensive shipping).

Earliness costs are incurred when jobs are completed before the promised due date. Because our system is single-stage, earliness costs are equivalent to finished goods holding costs. We will model earliness costs as increasing linearly with job earliness. In reality, earliness costs may be due to: (i) direct storage costs (space, manpower); (ii) increased chance of spoilage, obsolescence, damage, theft and loss; (iii) cost of capital tied up in inventory.

3.1.2.1 Expressing Tardiness Cost

We model tardiness costs as increasing linearly with time once a job's due date has passed. The cost parameter C_T, with dimensions currency per unit time, controls the steepness of the tardiness cost function. We define I_T (critical tardiness interval) as the time interval over which accrued tardiness costs equal job revenue and i_T (normalized critical tardiness interval) as this interval expressed as a multiple of the mean service time (i.e. i_T=$I_T\ \mu$). As can be seen in Figure 3.1, the slope of the tardiness cost curve (C_T) is equal to the job revenue (R) divided by I_T. The expected tardiness costs of an accepted job equal the cost parameter C_T multiplied by the expected tardiness of an accepted job (*AvgTardiness*).

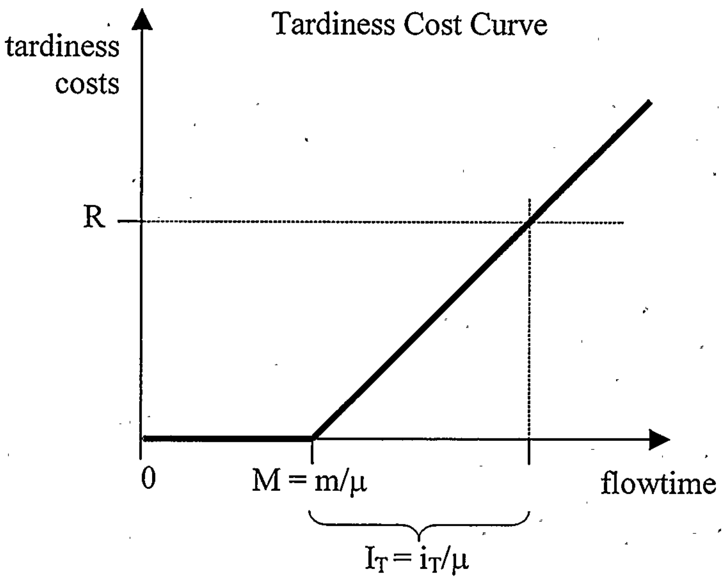


Figure 3.1: Tardiness Cost Function

We can minimize the interdependence of parameter value choices if we express the normalized critical tardiness as a multiple of the normalized flow allowance *m*, such that i_T=k_m *m*, where k_m is the cost magnitude factor[4]. By expressing the critical tardiness interval proportional to flow allowance we can use the same cost parameter (k_m) to compare scenarios under differing flow allowances. Note that decreasing k_m results in an *increase* in the severity of tardiness costs.

[4] Note that $k_m = \dfrac{\mu R}{m C_T}$.

3.1.2.2 Expressing Earliness Cost

We model earliness costs as increasing linearly with the time interval between the job's completion time and its due date. The cost parameter C_E, with dimensions currency per unit time, controls the steepness of the earliness cost function. We define I_E (critical earliness interval) as the time interval over which accrued earliness costs equal job revenue and i_E (normalized critical earliness interval) as this interval expressed as a multiple of the mean service time (i.e. i_E=I_E μ). As can be seen in Figure 3.2, the negative slope of the earliness cost function (C_E) is equal to the job revenue (R) divided by I_E. The expected earliness costs of an accepted job equal the cost parameter C_E multiplied by the expected earliness of an accepted job (*AvgEarliness*).

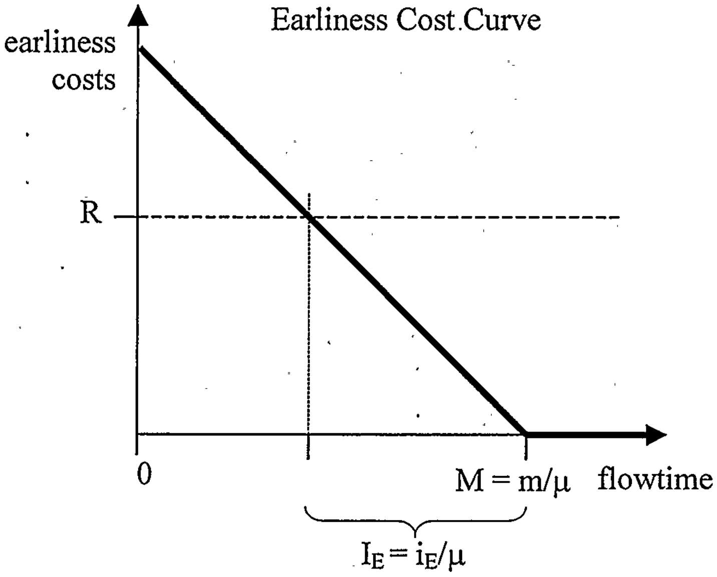


Figure 3.2: Earliness Cost Function

We can again minimize the interdependence of parameter value choices, by expressing the normalized critical earliness as a multiple of the normalized critical tardiness, such that $i_E = k_r\, k_m\, m$, where k_r is the relative earliness cost factor[5]. By setting the earliness interval proportional to flow allowance we can use the same cost parameters (k_m, k_r) to compare scenarios under differing flow allowances.

3.1.3 Profit Models

The goal is to maximize some measure of profit, however there are many ways of defining profit, even within our limited cost model. In order to simplify the analysis and make the results as meaningful as possible, we wish to reduce the number of 'arbitrarily' chosen parameters required to express our profit measure.

3.1.3.1 Profit Per Arriving Job

One approach that can be taken is to maximize the expected profit per arriving job (PPAJ), which is equal to:

$$PPAJ = Pr(acc) \times (R - C_T \times AvgTardiness - C_E \times AvgEarliness) \qquad (3.1)$$

Where *Pr(acc)* is the probability of an arriving job being accepted.

[5] Note that $k_r = \dfrac{\mu R}{k_m m C_E}$.

3.1.3.2 Profit Per Unit Arriving Revenue (PPUAR)

In the interests of reducing the number of parameters in the analysis, we may wish to express profit per unit arriving revenue (PPUAR) instead of per job. We do this by dividing PPAJ by job revenue:

$$PPUAR = Pr(acc) \times (1 - \frac{C_T}{R} AvgTardiness - \frac{C_E}{R} AvgEarliness) \qquad (3.2)$$

Or, in terms of the cost magnitude and relative earliness cost factors:

$$PPUAR = Pr(acc) \times (1 - \frac{\mu}{k_m m} AvgTardiness - \frac{\mu}{k_r k_m m} AvgEarliness) \qquad (3.3)$$

We can therefore express profit per unit arriving revenue in terms of only two cost parameters (k_m and k_r), which are independent of service time and of job revenue. This means that PPUAR is a function of three operational performance measures (percent accepted, expected tardiness and expected earliness) and two cost parameters.

3.1.3.3 Cost Per Unit Arriving Revenue (CPUAR)

It may be of interest to examine the costs contributing to PPUAR being less than 100%. We define the cost per unit arriving revenue (CPUAR) as this difference:

$$CPUAR = 1 - PPUAR \qquad (3.4)$$

What is of greater interest, is the breakdown of CPUAR into its three cost components: (i) rejection costs; (ii) earliness costs; (iii) tardiness costs. These three costs components are defined as:

$$\begin{aligned} CPUAR_{Rejection} &= 1 - Pr(acc) = Pr(rej) \\ CPUAR_{Earliness} &= Pr(acc) \times \frac{\mu}{k_m\, k_r\, m} AvgEarliness \\ CPUAR_{Tardiness} &= Pr(acc) \times \frac{\mu}{k_m\ m} AvgTardiness \\ CPUAR &= CPUAR_{Rejection} + CPUAR_{Earliness} + CPUAR_{Tardiness} \end{aligned} \tag{3.5}$$

Where *Pr(rej)* is the probability of rejecting an arriving job.

By examining the individual cost components we can gain insight into how the control policies under investigation improve system performance.

3.1.3.4 Absolute Profit Rate

The true goal of the firm is not to maximize the profit per arriving job, or per unit arriving revenue, but rather to maximize the rate of profit over time (so that the total profit in an interval is maximized). If the firm has no control over the arrival rate, then this is equivalent to maximizing the profit per arriving job or per unit arriving revenue. The actual profit rate (PR, dimensions currency per unit time) is the profit per arriving job (PPAJ) multiplied by the rate of job arrivals:

$$PR = \lambda \times Pr(acc) \times (R - C_T \times AvgTardiness - C_E \times AvgEarliness) \tag{3.6}$$

We will define the measure of absolute profit rate (APR) such that:

$$APR = \rho \times PPUAR \tag{3.7}$$

Where $\rho=\lambda/\mu$ is the traffic intensity. Note that the absolute profit rate is dimensionless, but for a given set of exogenous and control parameters can be interpreted as the ratio of the current profit rate, to the rate of arriving revenue when the traffic intensity is 100%.

It should be noted that in real systems there are costs associated with increasing the arrival rate that are not modelled here, and therefore arrival rate is considered an exogenous variable. However, it is desirable that a good control policy will yield an equal or better absolute profit rate as the arrival rate increases, even though PPUAR itself might decrease.

3.2 Chosen Parameter Values

The uncontrolled M/M/1 system involves four parameters considered exogenous in this analysis: (i) the traffic intensity, ρ; (ii) the normalized flow allowance, m; (iii) the cost magnitude factor, k_m; (iv) the relative earliness cost factor, k_r. This section establishes reasonable ranges for these parameter values, and selects specific values to be used in the numerical analysis of the experimental control policies investigated.

3.2.1 Traffic Intensity

Assuming that the mean service time is fixed, traffic intensity is determined by external demand. Since traffic intensity is externally determined, we will experiment with a full range of values, particularly for control systems for which there is an analytical representation of optimal PPUAR. In cases where generation of optimal PPUAR is more time-intensive, performance will be examined over a wide and reasonably-spaced range of traffic intensities.

3.2.2 Flow Allowance

Choosing appropriate test values for the flow allowance (*M* in natural time units, *m* as a multiple of mean service time) is important because we need to identify the parameter space where control can be beneficial. Additionally, we do not want to base the choice of flow allowance *value* on the PPUAR for any given set of parameter values, as this would bias performance towards that operating range. We will choose the flow allowance in a similar manner to the method Jensen *et al.* (1995) used to set the Total Work Content (TWK[6]) allowance factor for their dispatch-centred workload control research:

> For this experiment the TWK allowance factor was established through the use of pilot simulation runs after model validity checks had been completed. Multiple runs were made with SPT dispatching procedure, which can be

[6] Where there are multiple job types with varying expected service times, flow allowance can be assigned proportional to the expected total work content for the job over all operations. The flow allowance is equal to k_{TWK}*TWK, with k_{TWK} being the TWK allowance factor the authors are trying to determine.

> used to set a reasonably wide range of due dates (Philipoom *et al.*, 1993). Each run held the TWK allowance factor at different levels. The TWK allowance factor that corresponded to 5% of all jobs being tardy was 12.141, while the allowance that corresponded to 20% of all jobs being tardy was 4.065. These allowance factors were incorporated into the experimental design as loose and tight due-date settings, respectively.

Since the proportion tardy is not considered by our cost model, this method allows us to set *reasonable* values for loose and tight flow allowances that do not depend on, nor bias performance towards, the profit measure for any one set of parameter values.

For the M/M/1 system, the expected proportion tardy (PropTard) is equal to the integral from the due date to infinity of the probability density function (pdf) of flowtime (see section 3.3.2.1 for a derivation of this pdf):

$$PropTard_{MM1} = \int_{\frac{m}{\mu}}^{\infty} pdf_{MM1}(t)\,dt = \int_{\frac{m}{\mu}}^{\infty} (1-\rho)\,\mu\,e^{-\mu t(1-\rho)}\,dt = e^{-m(1-\rho)} \tag{3.8}$$

We therefore wish to set *m* to be:

$$m(\rho, PropTard_{MM1}) = -\frac{Ln(PropTard)}{(1-\rho)} \tag{3.9}$$

We will experiment with two settings for *m*, such that: (i) *m* representing a loose due date results in 5% of jobs being tardy at 90% traffic intensity; (ii) *m* representing a tight due results in 20% of jobs being tardy at 90% traffic intensity. This results in the following values:

1. m_{loose}=29.9573
2. m_{tight}=16.0944

3.2.3 Cost Magnitude Factor

The cost magnitude factor, k_m, represents the extent to which poor delivery performance results in actual costs. In real systems, the cost of poor delivery performance is based on many factors including the performance of competitors, the actual cost of tardiness to customers, the importance given to due dates in comparison to other factors such as quality, and the explicit promises made by marketing and sales. We desire to choose values for k_m that represent a realistic range of operating conditions for our testbed system, which leads to the selection of the following values:

1. k_m=1

   Represents a system with "low" tardiness costs. The tardiness costs of a job equal job revenue if the job takes 100% longer than promised to complete.

2. k_m=1/2

   Represents a system with "medium" tardiness costs. The tardiness costs of a job equal job revenue if the job takes 50% longer than promised to complete.

3. k_m=1/4

   Represents a system with "high" tardiness costs. The tardiness costs of a job equal job revenue if the job takes 25% longer than promised to complete.

3.2.4 Relative Earliness Cost Factor

In the literature, tardiness and earliness cost factors are frequently related to each other. Azizoglu and Webster (1997) fix the earliness interval at ten times the tardiness interval (or the cost per unit per unit time early at one-tenth the cost per unit per unit time tardy). Elhafsi (2002) fixes the earliness interval at five times the tardiness interval. These are assumed to be "typical" scenarios where the primary sources of earliness costs are the cost of capital, storage and handling and there are no special considerations for earliness (such as high perishability).

Dessouky *et al.* (1999) are motivated by a proposed chemical plant with extremely high earliness costs due to both high spoilage resulting from stability time constraints of chemical properties, and high investment costs in purchasing storage tanks. Planners for this plant estimated that tardiness costs would be "at least as high as earliness costs" indicating that setting the earliness interval equal to the tardiness interval is representative of a system with special circumstances that dictate extremely high earliness costs.

There are also scenarios where there are reasons for extremely low earliness costs relative to tardiness costs. This may occur under high tardiness costs (i.e. contractually-specified penalties) and/or under low earliness costs (i.e. non-physical 'manufacturing' scenarios such as data processing machines). We will not consider situations where earliness costs are negligible in comparison to tardiness costs.

Based on the above, we will use three settings for the relative earliness cost factor, k_r:

1. k_r=10

   Represents a system with no special cost considerations at the lower range of earliness costs (relative to tardiness costs). Earliness costs would equal 10% of tardiness costs accrued over the same interval.

2. k_r=5

   Represents a system with no special cost considerations at the higher range of earliness costs (relative to tardiness costs). Earliness costs would equal 20% of tardiness costs accrued over the same interval.

3. k_r=1

   Represents a system with extreme characteristics (such as high perishability or special storage considerations) that result in high earliness penalties. Earliness costs would equal 100% of tardiness costs accrued over the same interval.

3.3 M/M/1 Analytical Derivations

3.3.1 System Description

We will begin by investigating the performance of the M/M/1 system without any workload control under our cost models. The uncontrolled system will be used as a baseline to identify the range of exogenous parameters over which control is required (profit is negative), and over which operating ranges control policies may be of significant benefit. A diagram of the uncontrolled system is found in Figure 3.3.

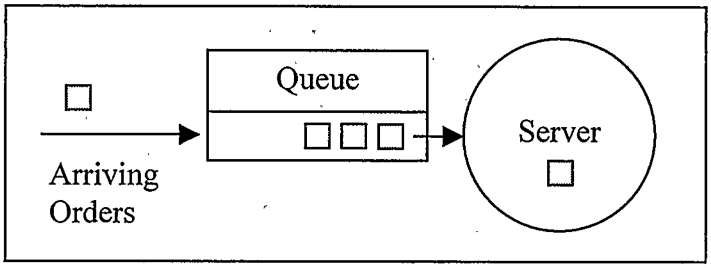


Figure 3.3: Diagram of the Uncontrolled M/M/1 System

3.3.2 Analytical Derivations for the M/M/1 System

3.3.2.1 M/M/1 Basics

The following section makes use of fundamental queuing theory. The reader is referred to the standard queuing theory books, Klienrock (1975) and Papadopoulos *et al.* (1993).

Recall that for an M/M/1 system, the probability of there being n items in the system (queue plus service) is π_n, given by the following relationship:

$$\pi_n = \rho^n \times (1-\rho) \tag{3.10}$$

Given n jobs currently in the system at arrival, the flowtime of an arriving job is the sum of n+1 service times sampled from an exponential distribution with mean $1/\mu$. The flowtime of the entering job when there are n jobs in the system is therefore distributed according to an Erlang distribution of order n+1:

$$f_{MM1}(n,t) = \frac{\mu^{n+1}\, t^n\, e^{-\mu t}}{n!} \tag{3.11}$$

The flowtime probability density function of any entering job is the sum of $f_{\mathrm{MM1}}(n,t)$ weighted by π_n for all n from zero to infinity:

$$pdf_{MM1}(t) = \sum_{n=0}^{\infty} \pi_n \times f_{MM1}(n,t) = \sum_{n=0}^{\infty} \rho^n (1-\rho) \times \frac{\mu^{n+1} t^n e^{-\mu t}}{n!} \tag{3.12}$$

This evaluates to an exponential distribution with mean $(\mu\,(1\text{-}\rho))^{-1}$:

$$pdf_{MM1}(t) = \mu(1-\rho)\ e^{-\mu(1-\rho)t} \tag{3.13}$$

3.3.2.2 Expected Tardiness

The expected tardiness of a job if flow allowance is $M=m/\mu$ is:

$$AvgTardiness_{MM1} = \int_{\frac{m}{\mu}}^{\infty} (t-\frac{m}{\mu})\ pdf_{MM1}(t)\ dt = \int_{\frac{m}{\mu}}^{\infty} (t-\frac{m}{\mu})(1-\rho)\,\mu e^{-\mu t(1-\rho)}\ dt \tag{3.14}$$

The integral evaluates to:

$$AvgTardiness_{MM1} = \frac{e^{-m(1-\rho)}}{\mu(1-\rho)} \tag{3.15}$$

As traffic intensity approaches zero, the expected tardiness becomes:

$$\lim_{\rho\to 0} AvgTardiness_{MM1} = \frac{e^{-m}}{\mu} \tag{3.16}$$

This will be an extremely small value relative to the flow allowance for reasonable values of *m*.

As traffic intensity approaches one, the expected tardiness becomes:

$$\lim_{\rho\to 1} AvgTardiness_{MM1} = \lim_{\rho\to 1} = \frac{e^{-m(1-\rho)}}{\mu(1-\rho)} = \infty \qquad (3.17)$$

Figure 3.4 shows the expected tardiness vs. traffic intensity for three values of *m* (5, 10 and 15) when $\mu = 1$. We see how expected tardiness slowly increases from negligible, and then spikes rapidly to infinity as traffic intensity approaches 100%. The lower the flow allowance, the greater the expected tardiness, and the lower the traffic intensity at which expected tardiness becomes non-negligible.

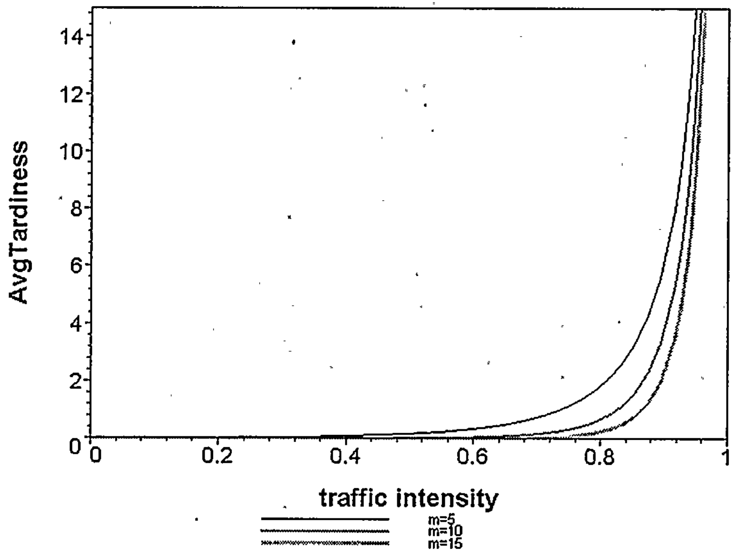


Figure 3.4: Expected Tardiness vs. ρ for M/M/1 (μ=1)

3.3.2.3 Expected Earliness

The expected earliness of a job is:

$$AvgEarliness_{MM1} = \int_{0}^{\frac{m}{\mu}} (\frac{m}{\mu} - t)\, pdf_{MM1}(t)\; dt = \int_{0}^{\frac{m}{\mu}} (\frac{m}{\mu} - t)(1-\rho)\,\mu\, e^{-\mu t(1-\rho)}\, dt \qquad (3.18)$$

The integral evaluates to:

$$AvgEarliness_{MM1} = \frac{e^{-m(1-\rho)} + m(1-\rho) - 1}{\mu(1-\rho)} \qquad (3.19)$$

Note that as traffic intensity approaches zero, the expected earliness approaches:

$$\lim_{\rho \to 0} AvgEarliness_{MM1} = \frac{e^{-m} + m - 1}{\mu} \qquad (3.20)$$

For reasonable values of *m*, the term e^{-m} will be very small. Therefore, as traffic intensity approaches zero, the expected tardiness becomes approximately (slightly greater than) the flow allowance minus the mean service time:

$$\lim_{\rho \to 0} AvgEarliness_{MM1} \approx \frac{m-1}{\mu} = M - \frac{1}{\mu} \qquad (3.21)$$

As traffic intensity approaches one, the average earliness becomes zero:

$$\lim_{\rho \to 1} AvgEarliness_{MM1} = \lim_{\rho \to 1} \frac{e^{m(1-\rho)} + m\left(1-\rho\right) - 1}{\mu\left(1-\rho\right)} = 0 \qquad (3.22)$$

Figure 3.5 shows the expected earliness vs. traffic intensity for three values of m (5, 10 and 15) when $\mu=1$. We see how expected earliness decreases from approximately $(m-1)/\mu$ to 0 as traffic intensity goes from 0 to 100%. We also see that increasing the flow allowance causes expected earliness to increase. It should be noted that while expected earliness is constrained by an upper limit, expected tardiness is unconstrained.

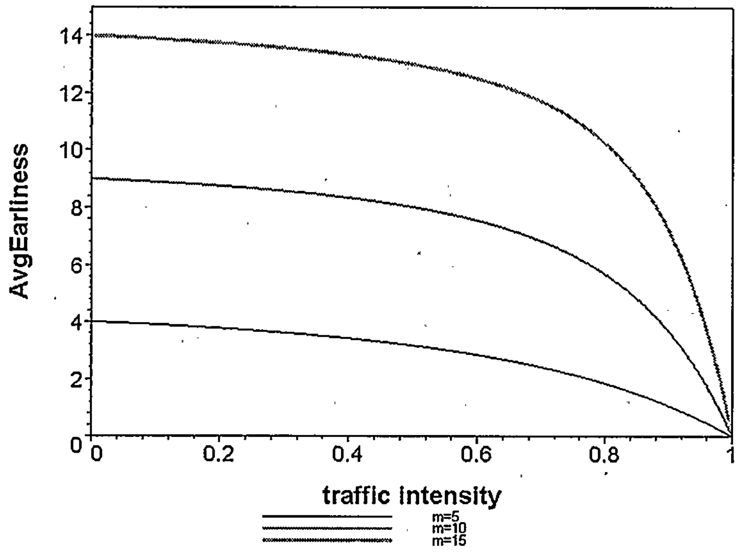


Figure 3.5: Expected Earliness vs. ρ for M/M/1 (μ=1)

3.3.2.4 M/M/1 PPUAR

Since there are no rejected jobs, our profit per unit arriving revenue is:

$$PPUAR_{MM1}=1-\frac{1}{k_m\,m}\frac{e^{-m(1-\rho)}}{(1-\rho)}-\frac{1}{k_r\,k_m\,m}\frac{e^{-m(1-\rho)}+m(1-\rho)-1}{(1-\rho)} \tag{3.23}$$

Based on our observations in the previous section, we expect that at high traffic intensities, PPUAR will be lowered from 100% primarily by tardiness costs, and at low traffic intensities, primarily by earliness costs. We also expect PPUAR to plunge to negative infinity as tardiness costs increase infinitely (when traffic intensity approaches 100%).

We can try to find the traffic intensity that yields the best PPUAR performance by taking the derivative of PPUAR with respect to ρ, and solving for its root:

$$\frac{dPPUAR}{d\rho}=\frac{1+e^{-m(1-\rho)}\left(1+k_r\right)\left(m\left(1-\rho\right)-1\right)}{k_m\,k_r\,m\left(1-\rho\right)^2}=0 \qquad (3.24)$$

Since we know ρ<1 and that k_m and k_r are real numbers, this can be reduced further:

$$1-e^{-m(1-\rho)}\left(1+k_r\right)\left(1-m\left(1-\rho\right)\right)=0 \qquad (3.25)$$

Note that k_m is no longer in the equation, and that the traffic intensity yielding the maximum PPUAR does not depend at all on k_m. The solution to Equation (3.25) that yields an optimal traffic intensity that is in the valid range of 0 to 100% is:

$$\rho_{opt}=\frac{W\left(-1,-\dfrac{1}{e\left(1+k_r\right)}\right)+m+1}{m} \qquad (3.26)$$

Where W(-1,x) is the –1 branch of Lambert's *W*-function (the primary branch, W(0,x), yields another real solution that is greater than 1). Since k_m is not part of the solution, the traffic intensity at which optimal PPUAR performance is achieved does not depend on the cost magnitude factor.

Substituting ρ_{opt} for ρ in Equation (3.23),and simplifying, gives us an optimal PPUAR value of:

$$PPUAR_{opt} = 1 - \frac{W\left(-1, \frac{-1}{e\left(k_r+1\right)}\right) - \left(k_r+1\right) e^{\left(W\left(-1, \frac{-1}{e\left(k_r+1\right)}\right)+1\right)} + 2}{k_m\, k_r \left(W\left(-1, \frac{-1}{e\left(k_r+1\right)}\right)+1\right)} \tag{3.27}$$

This is interestingly independent of m. This means that the optimal value of PPUAR does not change with flow allowance, although the traffic intensity at which this PPUAR is achieved does. Table 1 shows optimal ρ and PPUAR for all combinations of k_r, k_m and m under consideration.

Table 1: Optimal ρ and PPUAR for the Uncontrolled M/M/1 System

	Optimal ρ		Optimal PPUAR		
kr	**m=29.9573**	**m=16.0944**	**km=1**	**km=0.5**	**km=0.25**
10	86.62%	75.09%	92.00%	83.99%	67.99%
5	89.20%	79.90%	84.72%	69.44%	38.89%
1	94.40%	89.57%	37.34%	-25.33%	-150.65%

3.4 M/M/1 Results

In this section we explore the performance of the uncontrolled M/M/1 system under our profit model. We will explore a full range of earliness and tardiness cost factor combinations for all traffic intensities. Our goals for this analysis are to: (i) identify where the control of earliness is required; (ii) identify where the control of tardiness is required; (iii) gauge the effect of flow allowance on the performance of the system.

3.4.1 PPUAR vs. ρ

Figure 3.6 shows how PPUAR varies with traffic intensity when k_m=1, for all other explored combinations of parameter values. Figure 3.7 shows how PPUAR varies with traffic intensity when k_m=0.5, for all other explored combinations of parameter values. Figure 3.8 shows how PPUAR varies with traffic intensity when k_m=*0.25*, for all other explored combinations of parameter values. The main results of interest from these plots can be summarized as follows:

- When relative earliness costs are low or medium, PPUAR increases slowly with traffic intensity until peaking at the values (and traffic intensities) shown in Table 1. After peaking, PPUAR performance begins to degrade rapidly as traffic intensity approaches 100%.
- When relative earliness cost is high, PPUAR performance not only degrades at higher traffic intensity, but is also low at lower traffic intensity, with a relatively narrow range in which performance peaks (as per Table 1).
- Increasing the flow allowance increases the traffic intensity at which the (same) optimal PPUAR is achieved. Loosening the flow allowance also results in worse

PPUAR performance when operating below the optimal traffic intensity, and better performance when operating above the optimal traffic intensity.

- When the relative earliness cost is high, and the cost magnitude factor is medium or high, the uncontrolled system cannot operate profitably (with a positive PPUAR) at any traffic intensity.

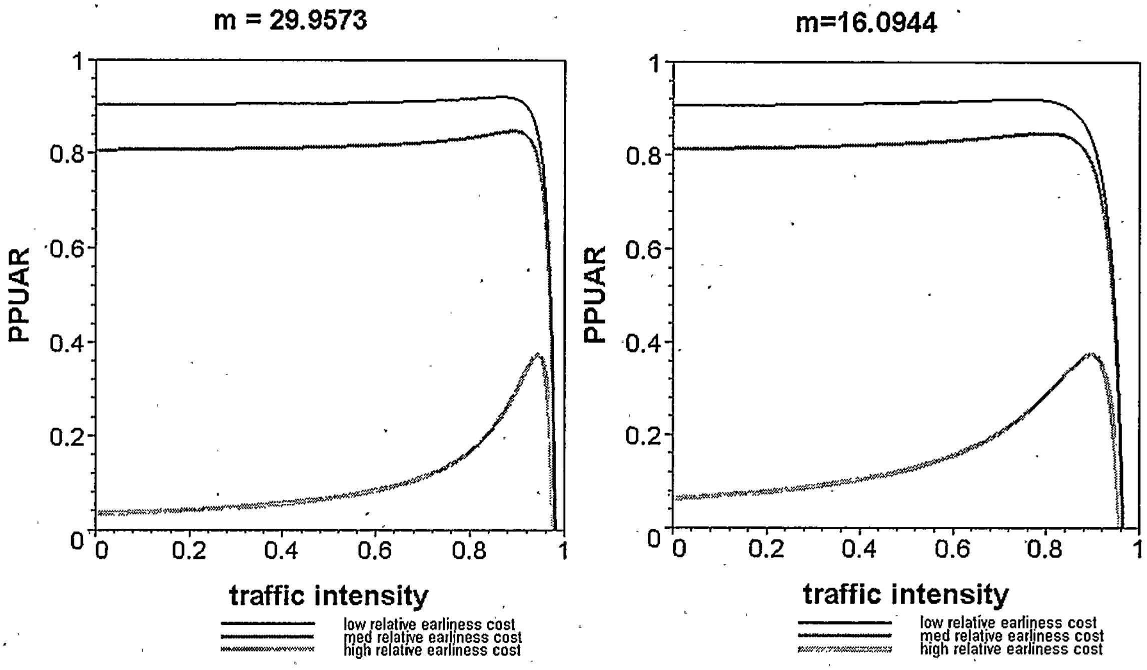


Figure 3.6: PPUAR vs. ρ for Uncontrolled M/M/1 at $k_m = 1$

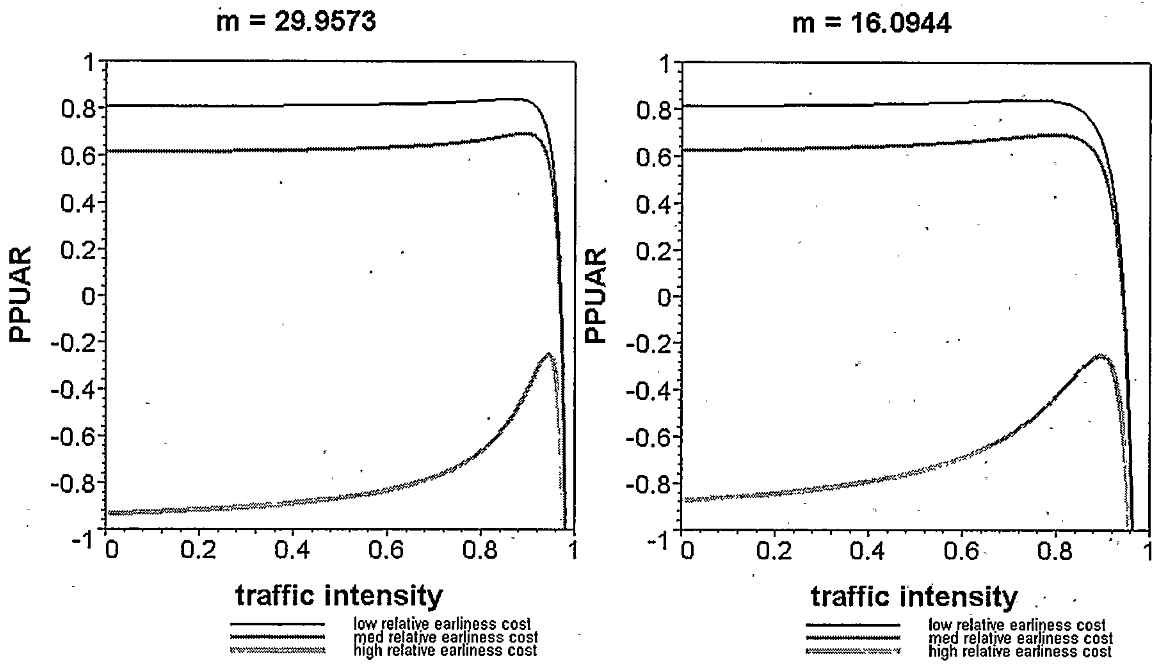


Figure 3.7: PPUAR vs. ρ for Uncontrolled M/M/1 at $k_m = 0.5$

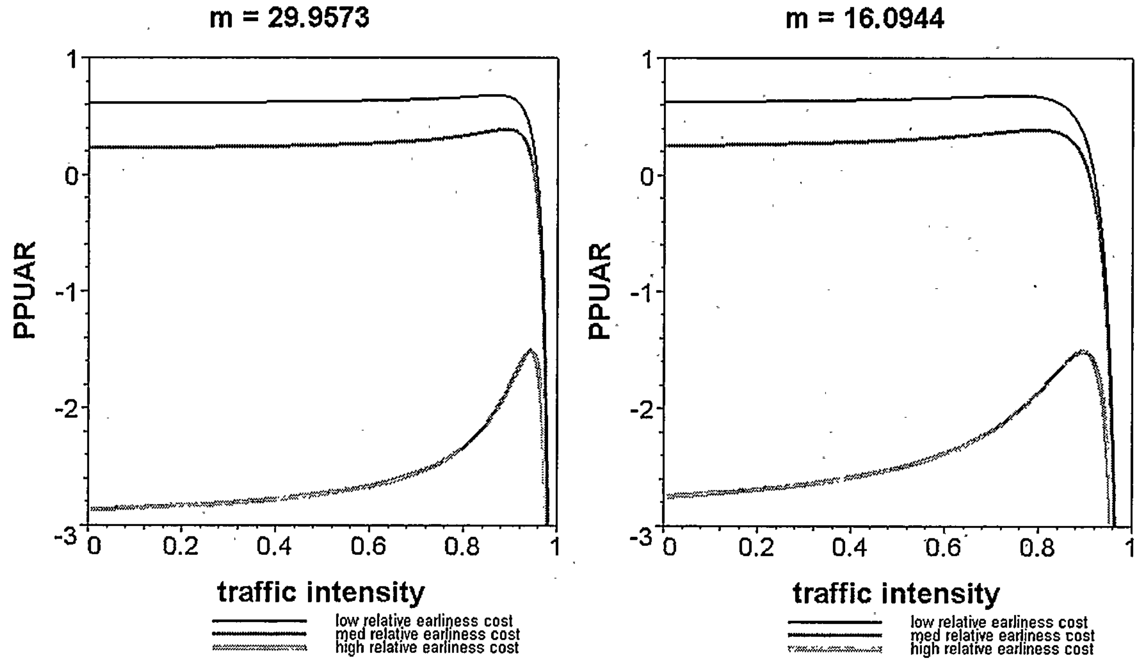


Figure 3.8: PPUAR vs. ρ for Uncontrolled M/M/1 at $k_m = 0.25$

3.4.2 Cost Breakdown

Figure 3.9 shows the cost per unit arriving revenue (CPUAR), total and broken down into earliness and tardiness components, for the lowest-cost parameter combination (m=29.9573, k_m=1, k_r=10). Figure 3.10 shows similar plots for a medium-cost parameter combination (m=16.0944, k_m=0.5, k_r=5), and Figure 3.11[7] for the highest-cost parameter combination (m=16.0944, k_m=0.25, k_r=1). These three sets of parameter combinations, summarized in Table 2, will be used extensively for analysis of control policies where simulation time constraints limit the number of parameter combinations that can be analyzed, or where space constraints require the selection of a limited number of parameter combinations to be investigated. The main results of interest from these plots can be summarized as follows:

- As anticipated from our analysis of expected tardiness, tardiness costs are negligible at low and medium traffic intensities, but increase sharply as higher traffic intensities are reached.
- As anticipated from our analysis of expected earliness, earliness costs decrease slowly with increasing traffic intensity for low traffic intensities, and decrease more rapidly to zero as traffic intensity approaches 100%.
- We see that the minimum total cost (maximum PPUAR) occurs as a tradeoff between the decreasing earliness costs, and the rapidly increasing tardiness costs.

[7] Note the difference in the y-axis scale for Figure 3.11.

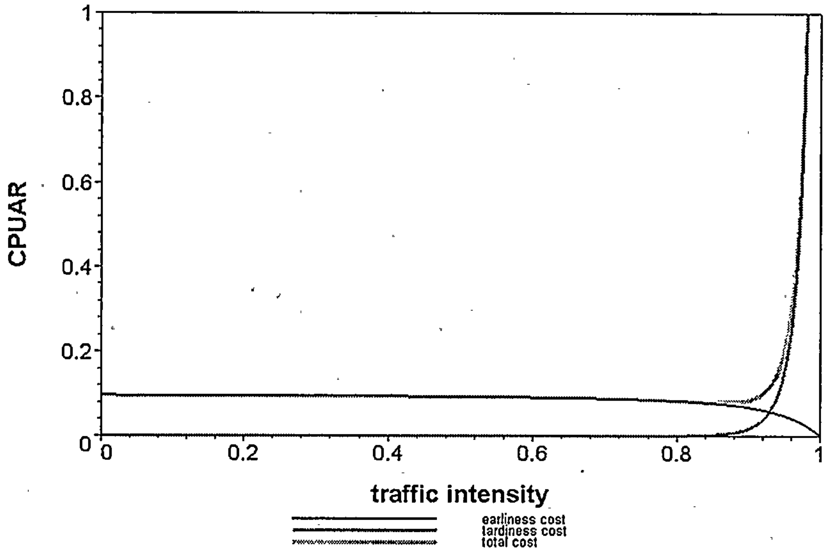


Figure 3.9: CPUAR vs. ρ for the Low Cost Parameter Combination

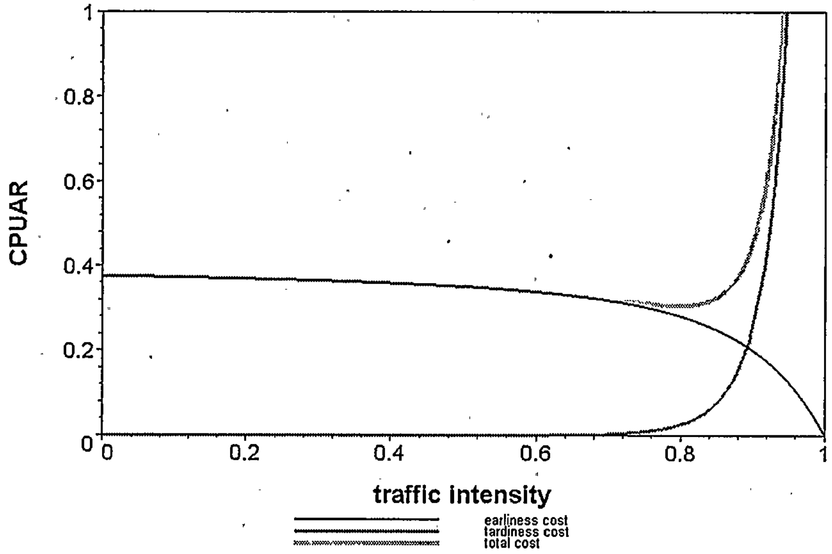


Figure 3.10: CPUAR vs. ρ for the Medium Cost Parameter Combination

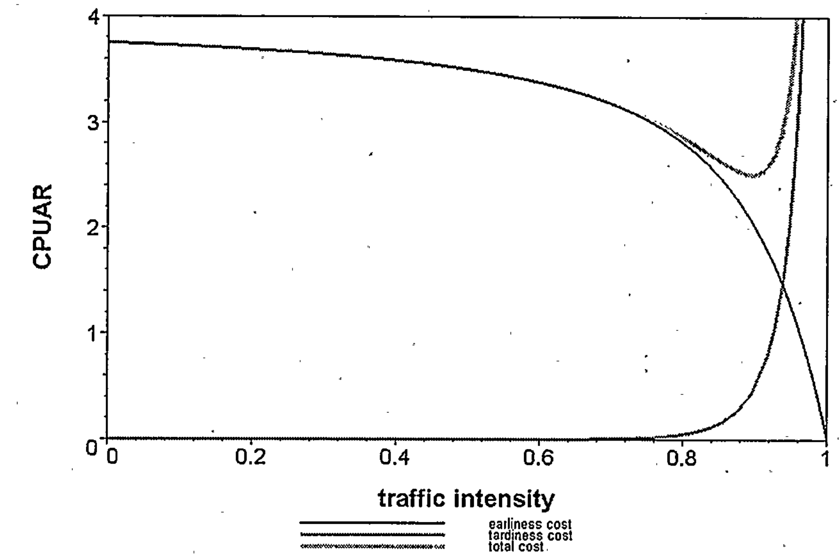


Figure 3.11: CPUAR vs. ρ for the High Cost Parameter Combination

Table 2: Cost Parameter Combinations

Parameter Combination	m	km	kr
Low Cost	30.9573	1	10
Medium Cost	16.0944	0.5	5
High Cost	16.0944	0.25	1

3.4.3 Absolute Profit Rate

Figure 3.12 shows how the absolute profit rate varies with traffic intensity when k_m=1 for all other explored combinations of parameter values. Figure 3.13 and Figure 3.14 show similar plots for k_m=0.5 and k_m=0.25 respectively. We note that for all the curves the absolute profit rate does decrease with increasing traffic intensity at higher traffic intensities. As stated previously a good control policy should yield a non-decreasing absolute profit rate as the arrival rate increases (even if PPUAR itself does decrease).

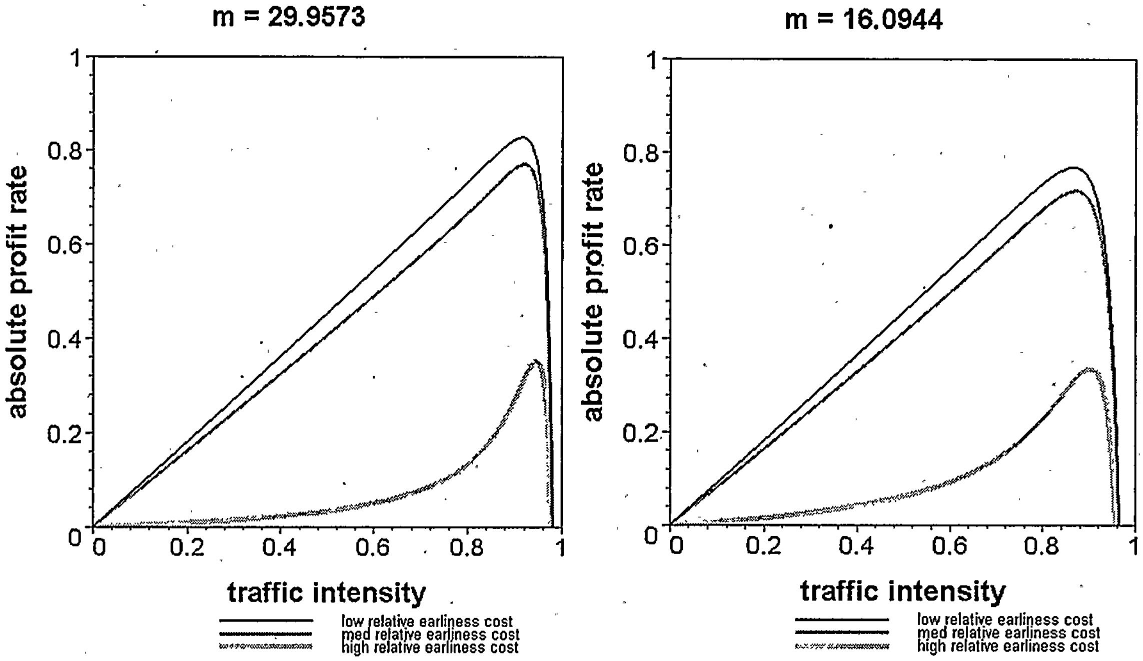


Figure 3.12: Absolute Profit Rate vs. ρ for Uncontrolled M/M/1 at $k_{\text{m}} = 1$

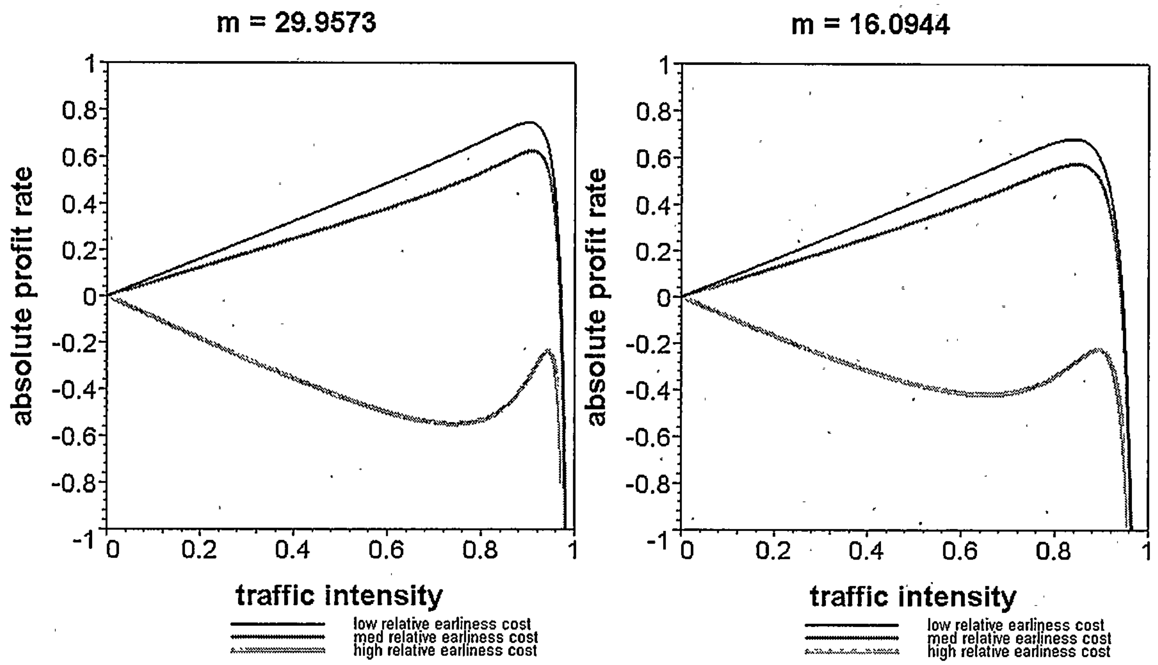


Figure 3.13: Absolute Profit Rate vs. ρ for Uncontrolled M/M/1 at $k_m = 0.5$

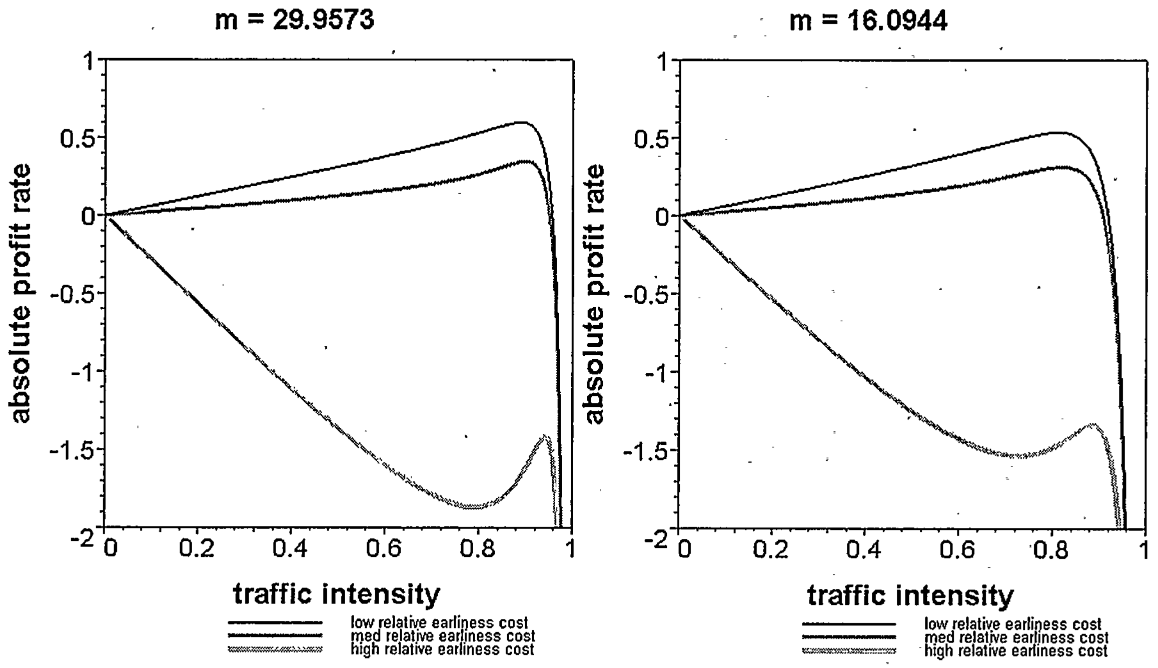


Figure 3.14: Absolute Profit Rate vs. ρ for Uncontrolled M/M/1 at $k_m = 0.25$

3.4.4 Summary of Results

We can summarize the results of the analysis of the M/M/1 system as follows:

1. When relative earliness costs are low or medium, PPUAR performance is relatively flat, increasing slowly with traffic intensity until peaking, after which it degrades rapidly as traffic intensity approaches 100%.
2. For high relative earliness cost settings, PPUAR performance increases more rapidly with traffic intensity before declining precipitously as traffic intensity approaches 100%. This yields a relatively narrow range of traffic intensity over which PPUAR is near-optimal.
3. The system cannot be operated profitably under high relative earliness costs situations when the cost magnitude is medium or high. This is due to the fact that either earliness costs or tardiness costs (or both) are always significant due to the high inherent variability of the M/M/1 system.
4. The cost magnitude factor does not affect the traffic intensity at which PPUAR is maximized.
5. The flow allowance does not affect the maximum value of PPUAR, but increasing the flow allowance increases the traffic intensity at which PPUAR is optimal. Increasing the flow allowance increases earliness costs (thereby degrading performance at lower traffic intensities), but reduces tardiness costs.

6. The absolute profit rate is not non-decreasing with increasing traffic intensity (but rather peaks and then rapidly degrades). This undesirable behaviour means that the availability of additional customers degrades profits.

These results will guide the analysis of the alternative control policies to be applied to our system.

Chapter 4

Analytically Tractable Control Policies

In this chapter we will investigate two analytically tractable control policies. The first control policy to be investigated, which will be referred to as the M/M/1 – d policy, uses a fixed release delay to emulate the order release component of the ORR mechanism. The second policy, which will be referred to as the M/M/1/N policy, uses a system work in process limit to reject arriving orders when a certain level of system congestion is present.

4.1 M/M/1 – d Analytical Derivations

4.1.1 Control System Description

We may be able to improve on system performance by introducing a fixed delay (D in natural time units, d as a multiple of mean service time) between the arrival of an order, and its release to the shop floor. All arriving jobs are held in a pre-release pool until D time units have elapsed since their arrival, and are then released to the queue. This control policy is a simple implementation of the order release functionality of the ORR mechanism, with the selective release being used to avoid excessive earliness costs. A diagram of this control policy, which we will refer to as M/M/1 – d, is shown in Figure 4.1.

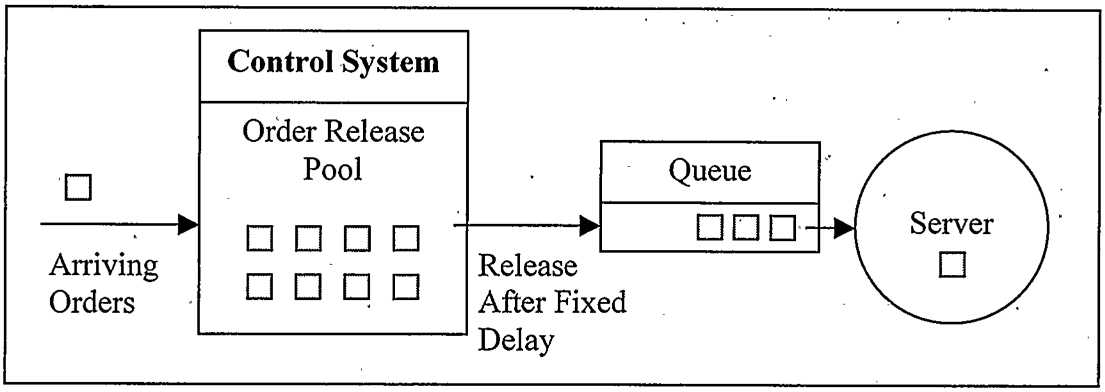


Figure 4.1: Diagram of the M/M/1 – *d* System

4.1.2 Analytical Derivations for the M/M/1 – d System

4.1.2.1 M/M/1 – *d* Basics

With this simple control policy, arrivals to the queue directly ahead of the server are still exponentially distributed (with all job arrivals simply shifted in time by the same amount). Therefore, the probability that there are n items in the queue/process component of the system is exactly the same as for the uncontrolled system, that is:

$$\pi_n = \rho^n \times (1-\rho) \tag{4.1}$$

The probability that there are n items in the pre release pool is the probability that there have been n arrivals in the last d/μ time units:

$$Pr_d(n) = \frac{e^{\lambda}\,\lambda^n}{n!} \tag{4.2}$$

Since all jobs stay in the pre-release pool for exactly d/μ time units, the probability density function of flowtime through the system is the same as for the uncontrolled M/M/1 system, but offset by d/μ time units. This yields the following expression for the flowtime probability density function (pdf), which is only valid when $t \geq d/\mu$:

$$pdf_{MM1-d}(t) = \sum_{n=0}^{\infty} \pi_n \times \frac{\mu^{(n+1)}\left(t - \frac{d}{\mu}\right)^n e^{-\mu\left(t-\frac{d}{\mu}\right)}}{n!} \tag{4.3}$$

This evaluates to:

$$pdf_{MM1-d}(t) = \mu(1-\rho)\,e^{-(\mu t-d)(1-\rho)} \tag{4.4}$$

4.1.2.2 Expected Tardiness

The expected tardiness of a job is:

$$AvgTardiness_{MM1-d} = \int_{\frac{m}{\mu}}^{\infty}(t-\frac{m}{\mu})\,pdf_{MM1-d}(t)\;dt = \int_{\frac{m}{\mu}}^{\infty}(t-\frac{m}{\mu})(1-\rho)\,\mu\,e^{-(\mu t-d)(1-\rho)}\;dt \tag{4.5}$$

This evaluates to:

$$AvgTardiness_{MM1-d} = \frac{e^{-(m-d)(1-\rho)}}{\mu(1-\rho)} \tag{4.6}$$

Note that the expected tardiness depends on the difference between the normalized flow allowance and the nomalized release delay, (*m-d*). As traffic intensity approaches 100%, the expected tardiness becomes:

$$\lim_{\rho\to 1} AvgTardiness_{MM1-d} = \lim_{\rho\to 1}\frac{e^{-(m-d)(1-\rho)}}{\mu(1-\rho)} = \infty \qquad (4.7)$$

This highlights, as expected, that the M/M/1 – *d* control policy is not capable of controlling tardiness at high traffic intensities. As traffic intensity approaches zero, the expected tardiness becomes:

$$\lim_{\rho\to 0} AvgTardiness_{MM1-d} = \lim_{\rho\to 0}\frac{e^{-(m-d)(1-\rho)}}{\mu(1-\rho)} = \frac{e^{-(m-d)}}{\mu} \qquad (4.8)$$

This expression is very small relative to $1/\mu$ when (*m-d*) is reasonably large.

Figure 4.2 shows the expected tardiness vs. traffic intensity for multiple values of *d* (0, 5, 10 and 15) when *m* is equal to 20 and μ is equal to 1. We see that at low traffic intensities, expected tardiness is negligible despite the presence of a release delay. However, the greater the release delay, the lower the traffic intensity at which expected tardiness ceases to be negligible, and the significantly earlier it becomes (infinitely) large. Thus the implementation of a release delay has a minimal effect on expected tardiness at low traffic intensities, and causes significantly worse tardiness performance at higher traffic intensities.

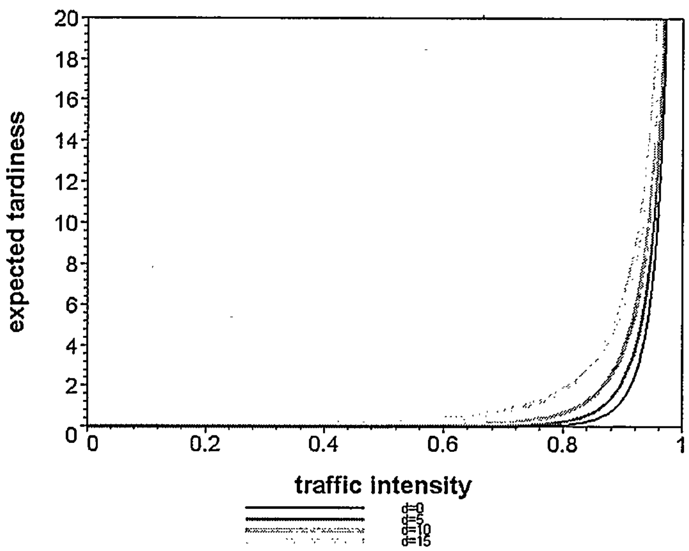


Figure 4.2: Expected Tardiness vs. ρ at Various Values of d (m=20, μ=1)

4.1.2.3 Expected Earliness

The expected earliness of a job is:

$$AvgEarliness_{MM1\text{-}d} = \int_0^{\frac{m}{\mu}} (\frac{m}{\mu} - t)\, pdf_{MM1\text{-}d}(t)\; dt = \int_0^{\frac{m}{\mu}} (\frac{m}{\mu} - t)(1-\rho)\,\mu\, e^{-(\mu t - d)(1-\rho)}\, dt \qquad (4.9)$$

This evaluates to:

$$AvgEarliness_{MM1\text{-}d} = \frac{e^{-(m-d)(1-\rho)} + (m-d)(1-\rho) - 1}{\mu(1-\rho)} \qquad (4.10)$$

Note that the expected earliness also depends only on the difference between the normalized flow allowance and the normalized release delay, (m-d). As traffic intensity approaches zero, the expected earliness approaches:

$$\lim_{\rho \to 0} AvgEarliness_{MM1-d} = \frac{e^{-(m-d)} + (m-d) - 1}{\mu} \tag{4.11}$$

Again, when (*m-d*) is reasonably large, the exponential term will become very small, thus the expected earliness becomes approximately (slightly greater than) the flow allowance, minus the release delay, minus the mean service time:

$$\lim_{\rho \to 0} AvgEarliness_{MM1-d} \approx M - D - \frac{1}{\mu} \tag{4.12}$$

As traffic intensity approaches one, the average earliness becomes zero:

$$\lim_{\rho \to 1} AvgEarliness_{MM1-d} = \lim_{\rho \to 1} \frac{e^{-(m-d)(1-\rho)} + (m-d)(1-\rho) - 1}{\mu(1-\rho)} = 0 \tag{4.13}$$

Figure 4.3 shows the expected earliness vs. traffic intensity for multiple values of *d* (0, 5, 10 and 15) when *m* is equal to 20 and μ is equal to 1. We see that the implementation of a release delay significantly reduces the earliness, although this improvement becomes less significant as traffic intensity approaches 100% (where expected earliness becomes zero).

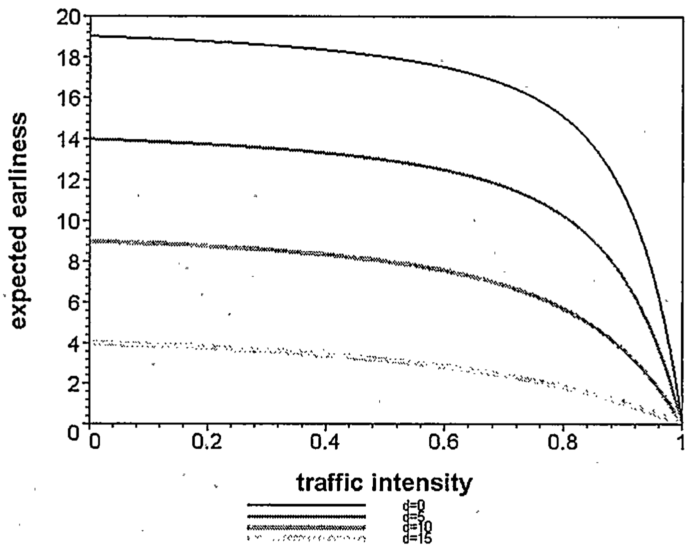


Figure 4.3: Expected Earliness vs. ρ at Various Values of d (m=20, μ=1)

4.1.2.4 Profit Per Unit Arriving Revenue

Since there are no rejected jobs, our profit per unit arriving revenue is:

$$PPUAR_{MM1-d} = 1 - \frac{1}{k_m\, m}\frac{e^{-(m-d)(1-\rho)}}{(1-\rho)} - \frac{1}{k_m\, k_r\, m}\frac{e^{-(m-d)(1-\rho)} + (m-d)(1-\rho) - 1}{(1-\rho)} \tag{4.14}$$

4.1.2.5 Optimal Release Delay

We can find the optimal release delay, d^*, by solving for the root of the derivative of $\mathrm{PPUAR}_{\mathrm{MM1\text{-}d}}$ with respect to d. The derivative of PPUAR with respect to d is:

$$\frac{d}{dd} PPUAR_{MM1-d} = \frac{1-(1+k_r)e^{-(m-d)(1-\rho)}}{k_m\, k_r\, m} \tag{4.15}$$

Setting this equal to zero and solving for d^* yields the results below. Note that since we cannot release an order prior to its arrival, d^* cannot be negative, hence the formula includes a "max" function to ensure that only valid values for d^* result:

$$d^* = \max(0, m - \frac{\ln(1+k_r)}{(1-\rho)}) \tag{4.16}$$

Since d^* is constrained to be non-negative in practice, we can see that the M/M/1 – d policy only exerts control when d^* is greater than 0, which occurs under the following condition:

$$m - \frac{\ln(1+k_r)}{(1-\rho)} > 0 \tag{4.17}$$

Note that the optimal release delay is independent of k_m. Given flow allowance, m, and relative earliness cost factor, k_r, we can find the range of traffic intensities over which this control policy is appropriate (ρ_d):

$$\rho_d < 1 - \frac{\ln(1+k_r)}{m} \tag{4.18}$$

Figure 4.4 plots the maximum traffic intensity for which the M/M/1 – d control policy is appropriate vs. m for several values of k_r. As expected, lowering the flow allowance or increasing the relative earliness cost factor (decreasing the severity of earliness costs) reduces the maximum traffic intensity for which the optimal release delay is non-zero. Note that at very low m values, the delay is always zero.

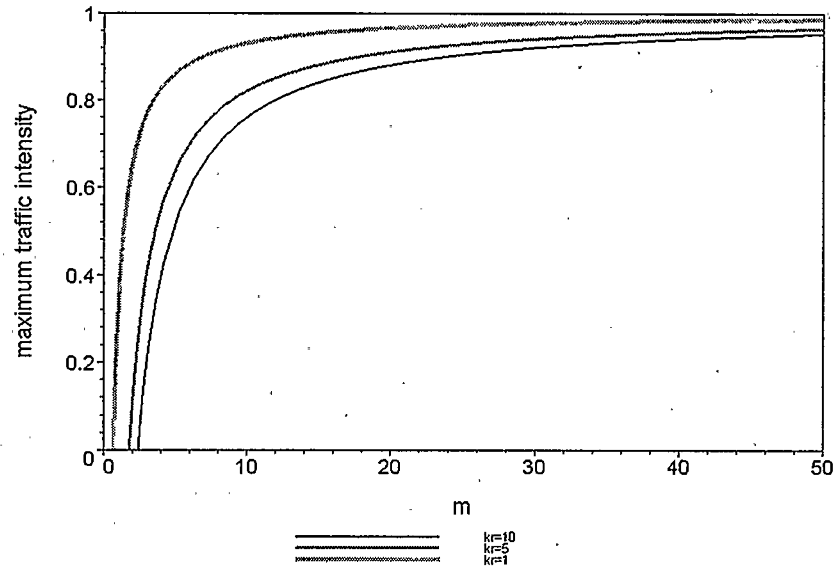


Figure 4.4: Maximum ρ Value for Which M/M/1 – d is Appropriate vs. m (for Various k_r Values)

4.1.2.6 Optimal PPUAR

Substituting the optimal release delay into our expression for PPUAR we get an expression for optimal PPUAR:

$$PPUAR^*_{MM1-d} = 1 - \frac{e^{-\left(m-\max\left(0, m-\frac{\ln(1+k_r)}{(1-\rho)}\right)\right)(1-\rho)}}{k_m m(1-\rho)} - \frac{e^{-\left(m-\max\left(0, m-\frac{\ln(1+k_r)}{(1-\rho)}\right)\right)(1-\rho)} + \left(m-\max\left(0, m-\frac{\ln(1+k_r)}{(1-\rho)}\right)\right)(1-\rho)-1}{k_m k_r m(1-\rho)} \quad (4.19)$$

Provided that we restrict out attention to those situations where the optimal release delay is non-zero, this expression can be further simplified as follows:

$$PPUAR^*_{MM1-d} = 1 - \frac{\ln(k_r+1)}{k_m k_r \, m(1-\rho)}. \quad (4.20)$$

4.2 M/M/1 – d Results

In this section we explore the performance of the M/M/1 – *d* system under our profit model. First we will explore *d**, then we will explore the performance at *d** for a full range of exogenous parameter combinations. We will also explore the sensitivity of performance to non-optimal delays.

4.2.1 *d* vs* ρ

Figure 4.5 shows *d** vs. ρ for loose and tight due dates and for low, medium and high values of k_r (recall that *d** is independent of k_m). We see that *d** starts at $m - ln(1+k_r)$, decreasing as traffic intensity increases until it becomes zero, where it remains as traffic intensity approaches 100%. As expected, increasing the severity of earliness costs increases *d** (since increasing the release delay reduces expected earliness). As noted earlier, increasing the severity of earliness costs also increases the traffic intensity at which *d** becomes zero. Decreasing the flow allowance decreases the traffic intensity at which *d** becomes zero, and generally lowers the value of *d** at all traffic intensities.

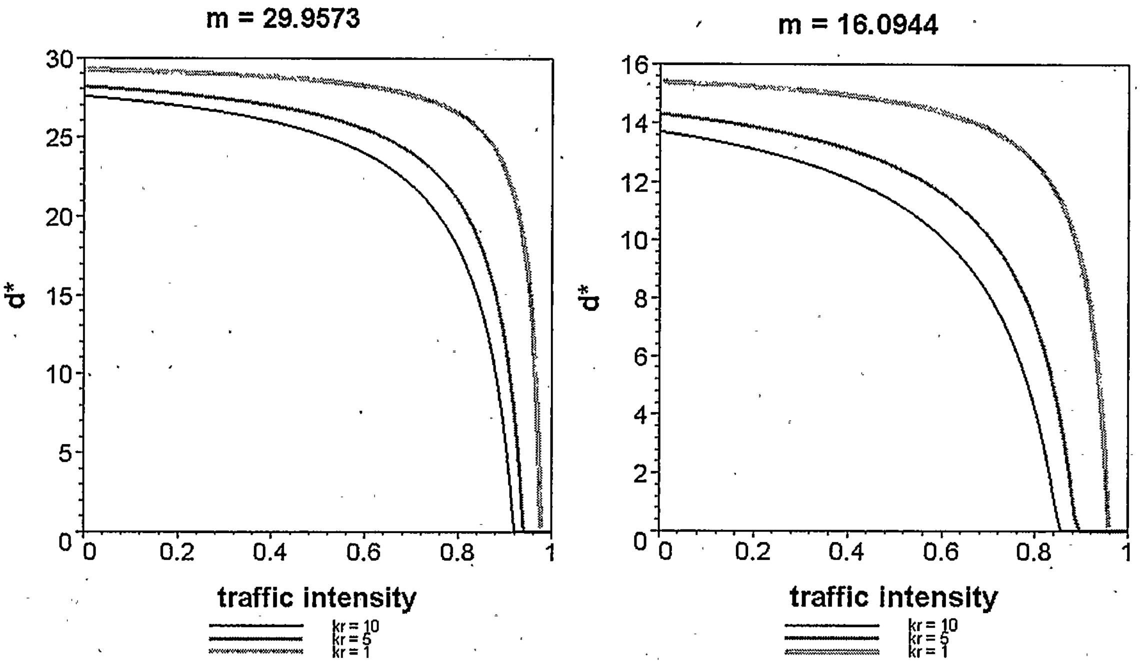


Figure 4.5: ***d** vs. *ρ***

4.2.2 PPUAR vs. ρ*

Figure 4.6 shows how PPUAR at d^* varies with traffic intensity when k_m=1, for all other explored combinations of parameter values. Figure 4.7 and Figure 4.8 show similar plots for k_m=0.5 and k_m=0.25 respectively. The main results of interest from these plots can be summarized as follows:

- Unlike for the uncontrolled system, PPUAR performance is highest at minimal traffic intensity and decreases as traffic intensity increases. As predicted analytically, we see that PPUAR decreases from a maximum when ρ is zero, instead of rising to a maximum at an intermediate ρ as it did without control

(Figure 3.6 through Figure 3.8). The difference is especially dramatic at low traffic intensities when k_r=1 and k_m=0.5 or 0.25.

- At high traffic intensity, M/M/1 – *d* behaves identically to the uncontrolled system (since *d** is zero) with PPUAR becoming rapidly negative as traffic intensity approaches 100%.
- This decline happens at lower traffic intensities for increasingly severe cost regimes.

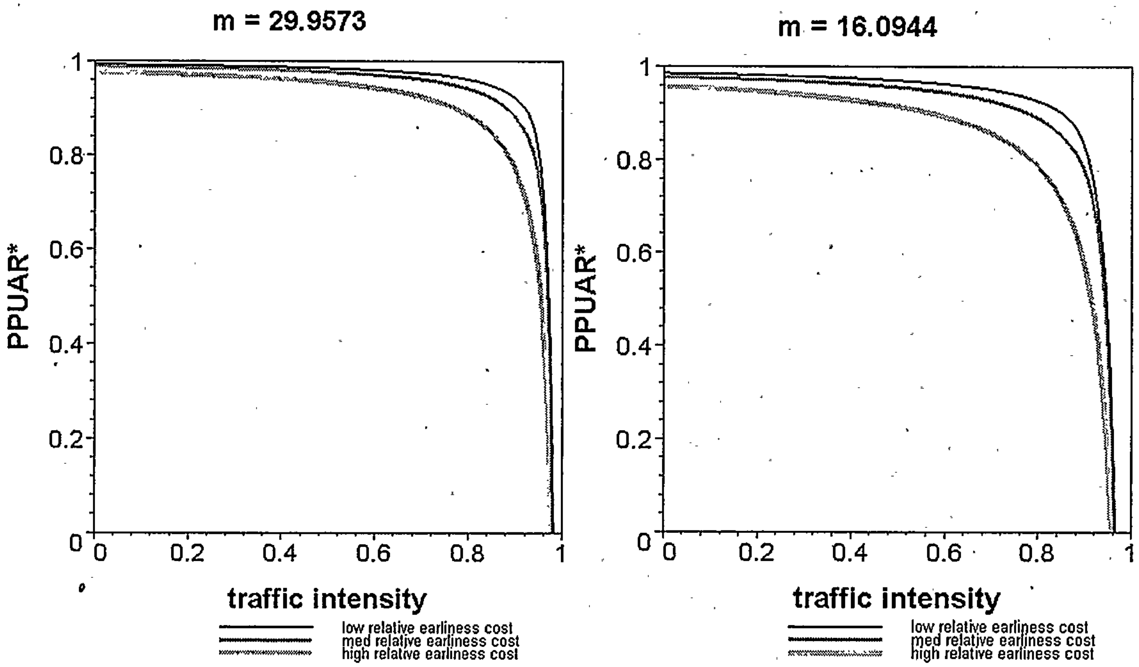


Figure 4.6: PPUAR* vs. ρ for M/M/1 – d at k_m = 1

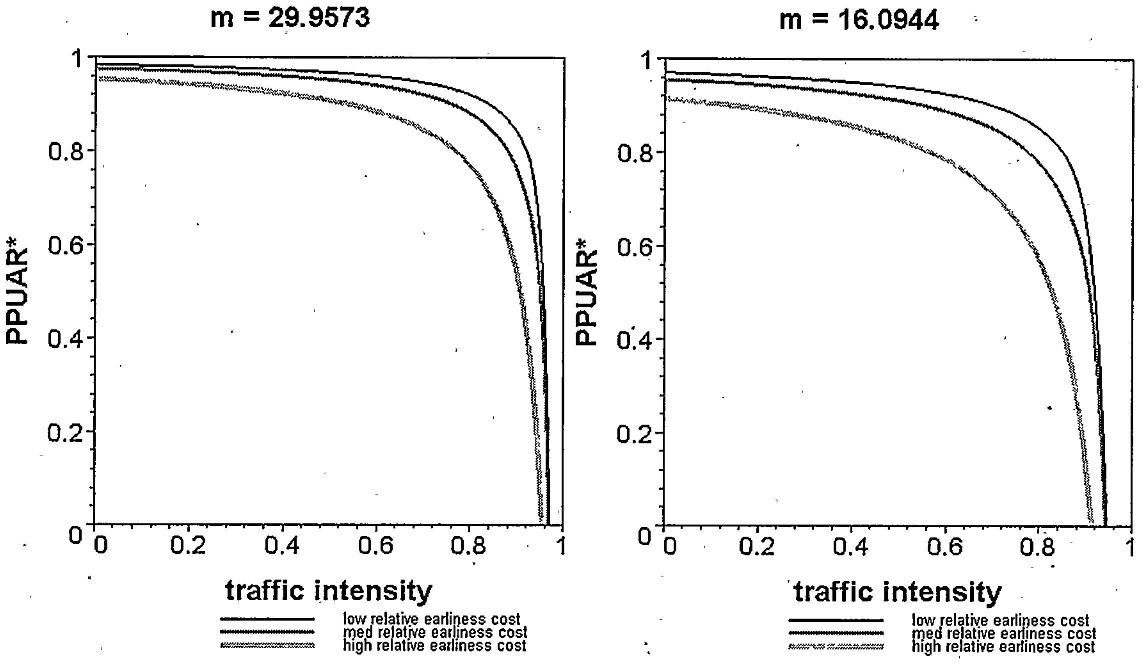


Figure 4.7: PPUAR* vs. ρ for M/M/1 – d at $k_m = 0.5$

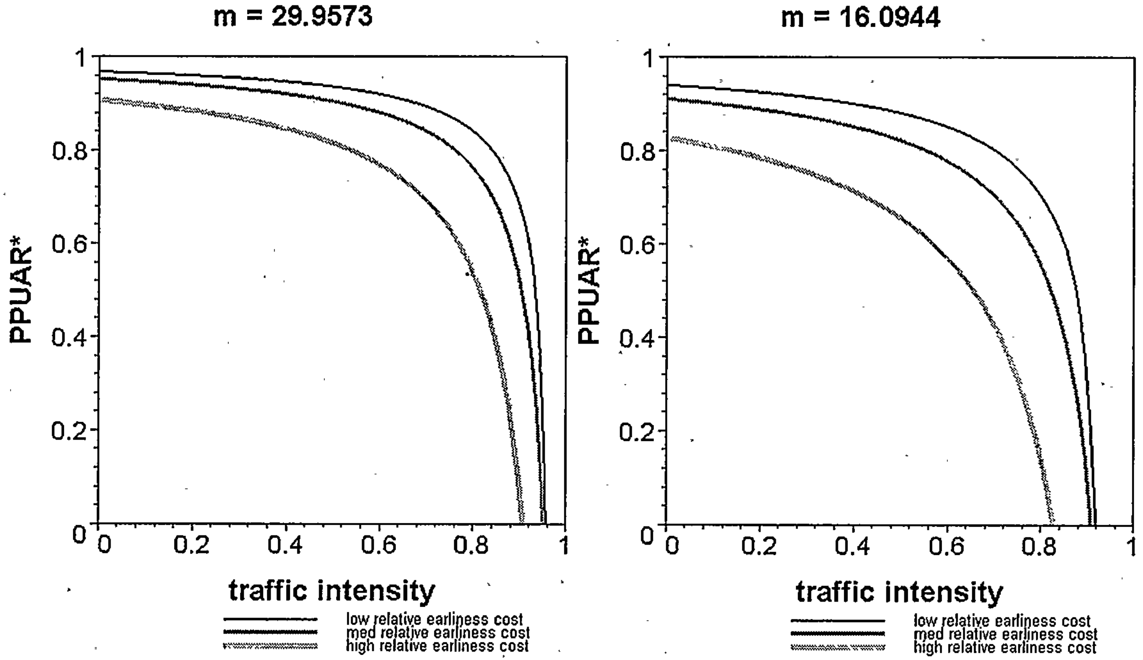


Figure 4.8: PPUAR* vs. ρ for M/M/1 – d at $k_m = 0.25$

4.2.3 Cost Breakdown

Figure 4.9, Figure 4.10 and Figure 4.11[8] show the cost per unit arriving revenue (CPUAR), total and broken down into earliness and tardiness components, for the previously examined low, medium and high cost parameter combinations respectively. There are two distinct differences between these figures, and those for the uncontrolled system (Figure 3.9 through Figure 3.11):

- Both tardiness and earliness costs increase with traffic intensity, as opposed to the uncontrolled system where earliness costs decreased with increasing traffic intensity. We do see earliness costs decreasing sharply at high traffic intensities (the discontinuity indicates that this decrease begins at the point that d^* ceases to be non-zero), however this decrease occurs at the point where tardiness costs begin to increase exponentially and dominate the total cost.
- Unlike for the uncontrolled system, tardiness costs are not zero at low traffic intensities. The M/M/1 – d policy accepts a small tardiness cost in exchange for a significant reduction in earliness cost.

[8] Note the difference in vertical axes scale for Figure 4.11.

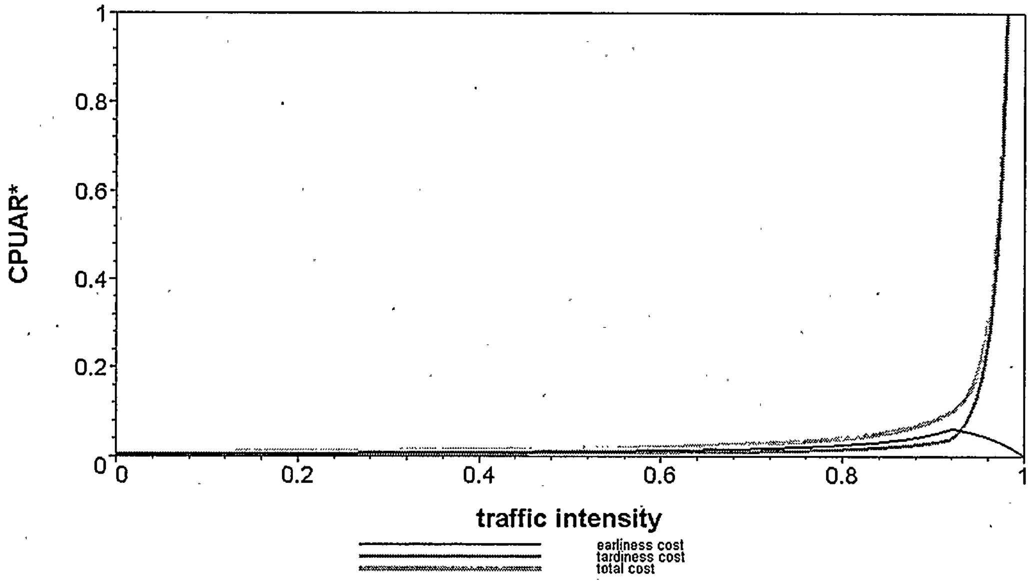


Figure 4.9: CPUAR* vs. ρ for the Low Cost Parameter Combination

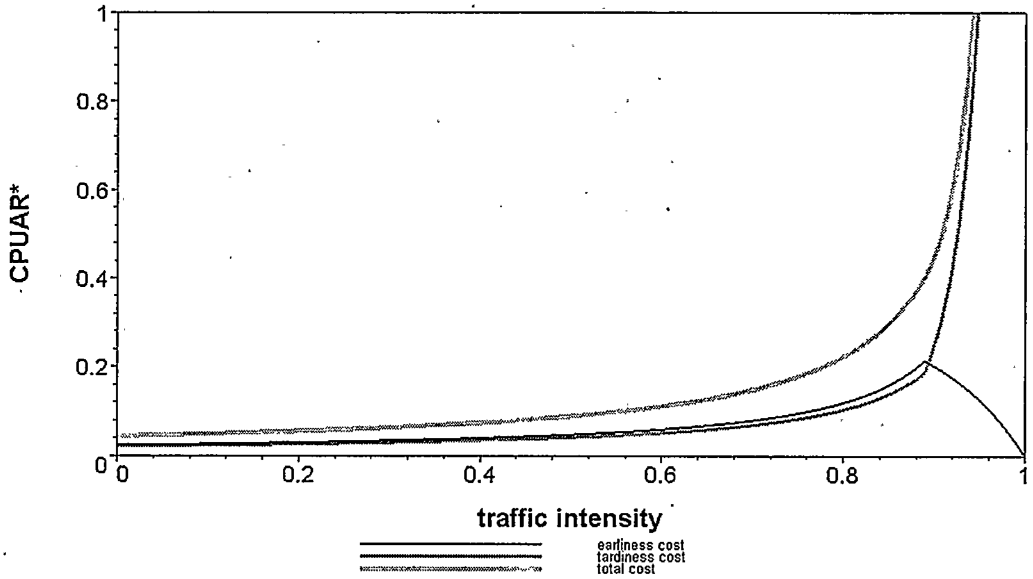


Figure 4.10: CPUAR* vs. ρ for the Medium Cost Parameter Combination

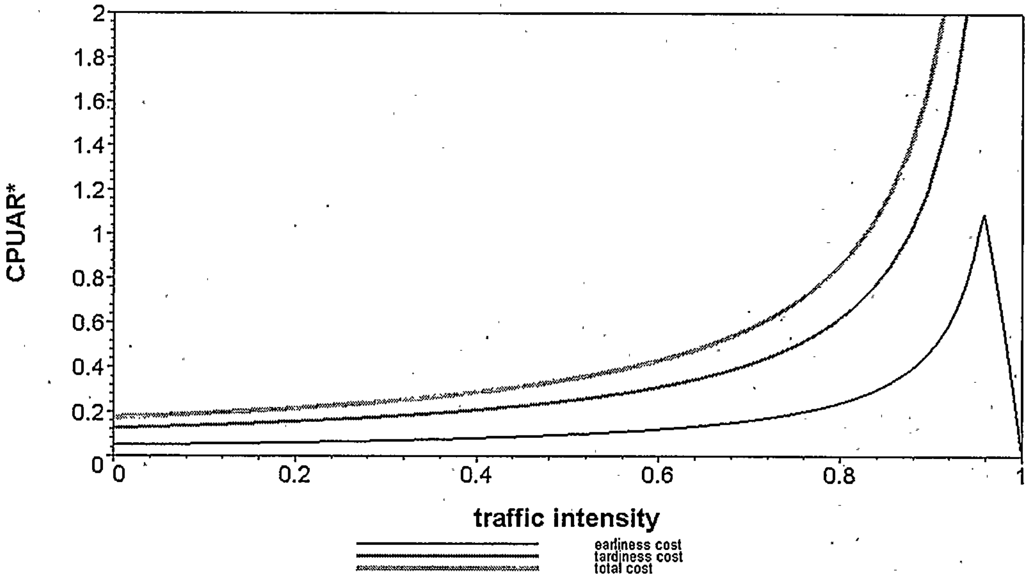


Figure 4.11: CPUAR* vs. ρ for the High Cost Parameter Combination

4.2.4 Absolute Profit Rate

Figure 4.12 shows how the absolute profit rate varies with traffic intensity when k_m=1 for all other explored combinations of parameter values. Figure 4.13 and Figure 4.14 show similar plots for k_m=0.5 and k_m=0.25 respectively. We again note that for all the curves the absolute profit rate does decrease with increasing traffic intensity at high traffic intensities. As stated previously a good control policy should yield a continuously higher absolute profit rate as the arrival rate increases (even if PPUAR itself does decrease).

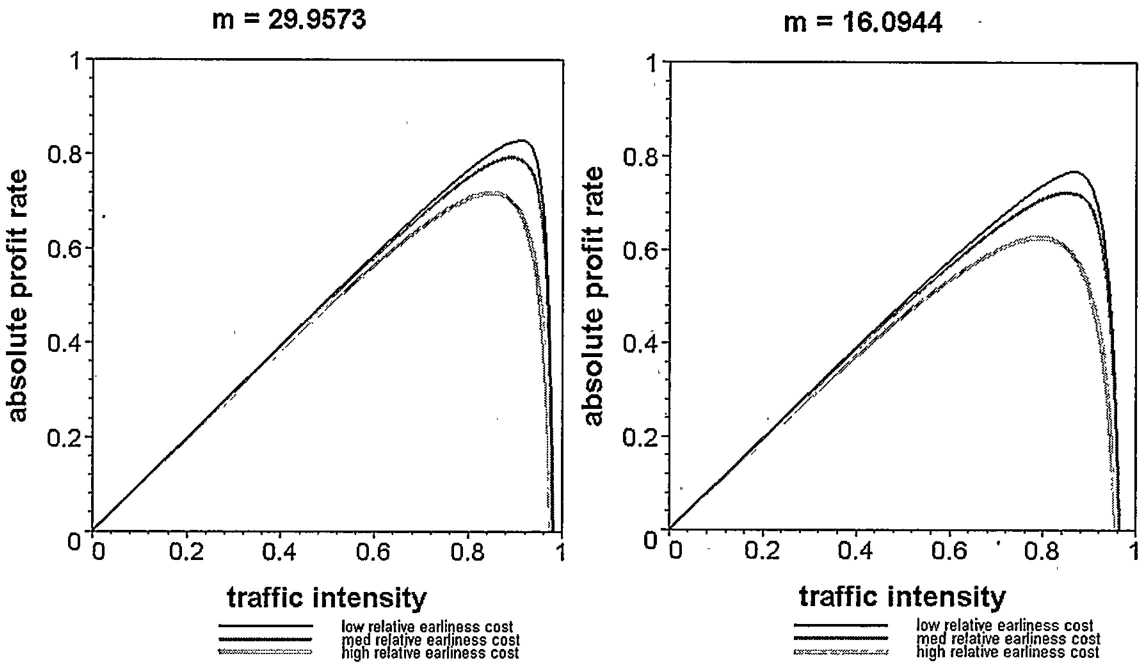


Figure 4.12: Absolute Profit Rate vs. ρ for M/M/1 – d at $k_{\mathrm{m}} = 1$

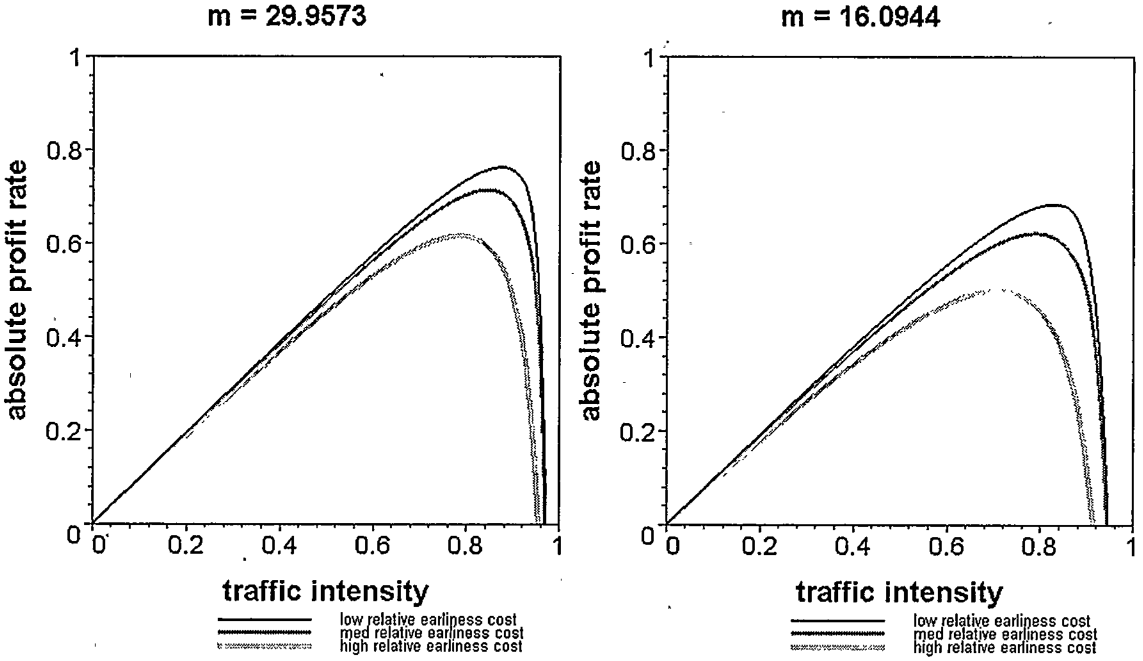


Figure 4.13: Absolute Profit Rate vs. ρ for M/M/1 – d at $k_{\mathrm{m}} = 0.5$

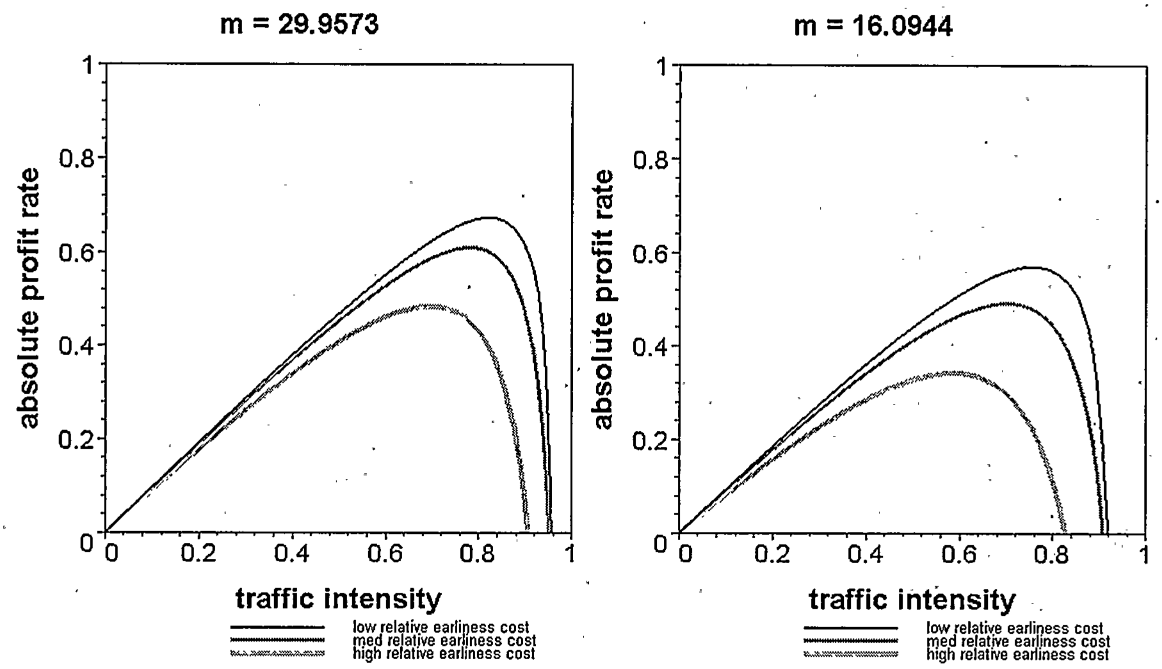


Figure 4.14: Absolute Profit Rate vs. ρ for M/M/1 – d at $k_m = 0.25$

4.2.5 Sensitivity of PPUAR Performance to Non-Optimal d

We wish to explore the sensitivity of PPUAR performance under the M/M/1 – d control policy to non-optimal values of d. Figure 4.15 shows how PPUAR performance varies with d under the medium cost parameter combination (k_m=0.5, k_r=5, m=16.0944) when traffic intensity is 75% (for these parameters, d^* is equal to 8.93). The horizontal line is the PPUAR performance for the equivalent uncontrolled M/M/1 system. We note that: (i) PPUAR performance is relatively flat around d^*; (ii) PPUAR performance is better than that for the uncontrolled system for an extremely wide range of d, from zero to just over fourteen.

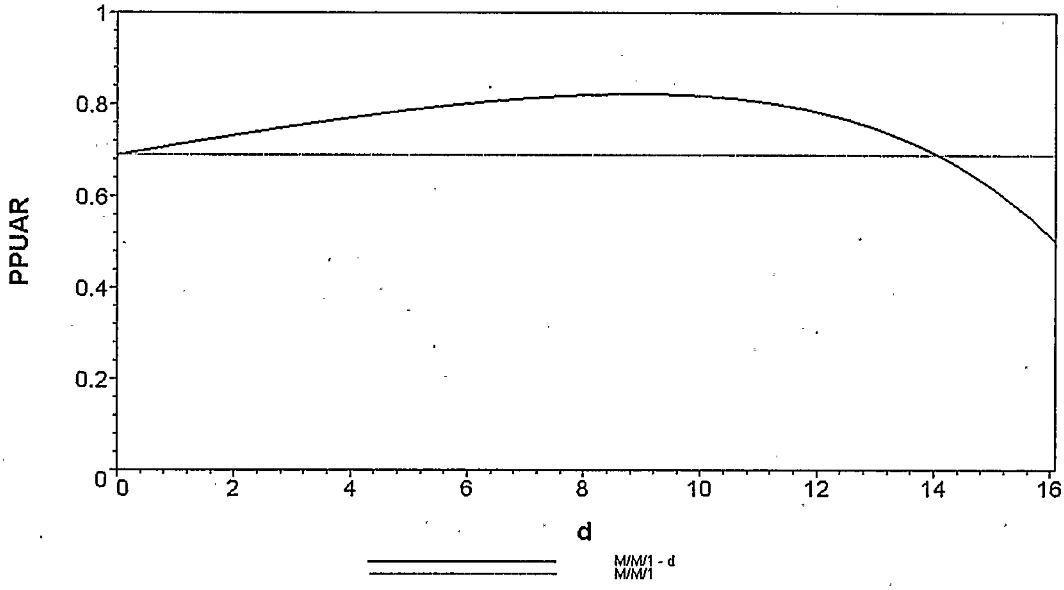


Figure 4.15: PPUAR vs. *d* for the Medium Cost Parameter Combination, ρ=75%

Figure 4.16 shows a similar figure for a higher traffic intensity of 85%, at which *d** is equal to 4.15. PPUAR performance is still flat around the *d**, but now M/M/1 – *d* performance is worse than when uncontrolled when *d* is only eight or greater. Thus as traffic intensity increases (and *d** gets smaller) it becomes more important to ensure that the release delay is not so large that it is worse than no delay at all. Erring towards a lower-than-optimal release delay will never result in performance worse than that of the uncontrolled system.

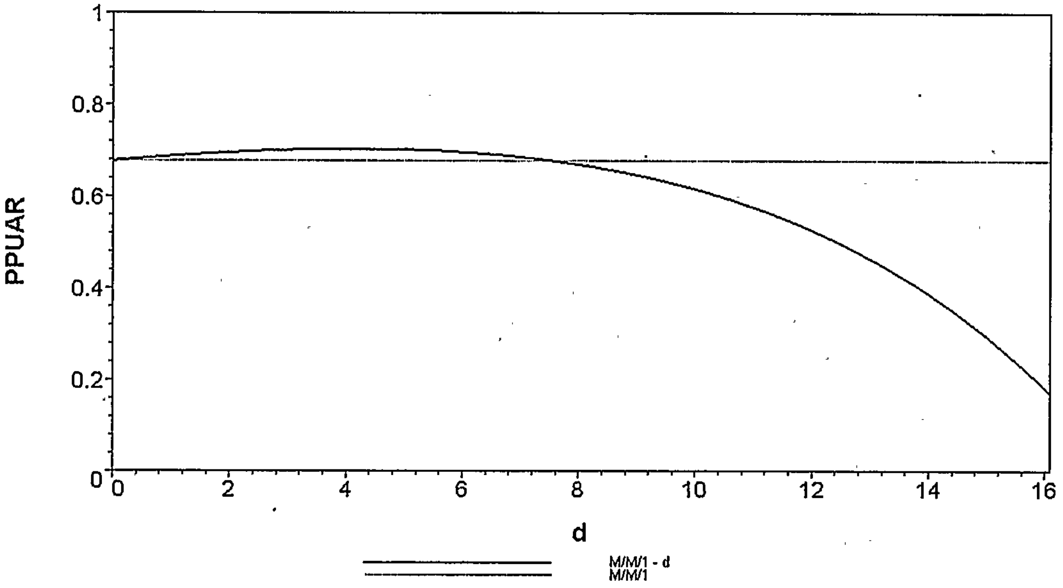


Figure 4.16: PPUAR vs. *d* for the Medium Cost Parameter Combination, ρ=85%

4.2.6 Summary of Results

We can summarize the results of the analysis of the M/M/1 – *d* system as follows:

1. We can achieve a significant improvement in PPUAR over the uncontrolled system at low and medium traffic intensities. As traffic intensity increases beyond some high value (85%-95% depending on cost parameters and flow allowance), the control policy becomes irrelevant (*d**=0) and performance is identical to that of the uncontrolled M/M/1 (i.e. rapidly plunging as traffic intensity approaches 100%).

2. Performance improvement is achieved by incurring some additional tardiness cost in exchange for greater reductions in earliness costs. Therefore, if earliness costs

are low (either because earliness is low or because the penalty for earliness is low) the M/M/1 – *d* policy will not result in significant improvement.

3. The absolute profit rate is not non-decreasing with increasing traffic intensity (but rather peaks and then rapidly degrades). This undesirable behaviour means that the availability of additional customers degrades profits.

4. PPUAR performance is relatively flat for near-optimal release delays. Under-estimating the optimal release delay will still result in better performance then when the system is uncontrolled, but overestimating may actually degrade performance.

The results for the M/M/1 – *d* analysis help address the criticism that time spent in a pre-release pool more than offsets manufacturing lead time reductions. We see that the implementation of a release delay can improve PPUAR performance despite causing an *increase* in mean tardiness over that for the uncontrolled system. This is because the delay results in earliness cost reductions, which are included in our cost model. Much of the ORR literature is concerned only with tardiness (mean tardiness or percent tardy), and examining tardiness-based measures alone would not show any positive performance results.

4.3 M/M/1/N Analytical Derivations

4.3.1 Control System Description

An alternative approach to improving system performance is to reject any jobs which arrive to find the system "full". Specifically, we can make use of a system work in process limit (*N*), such that any job which arrives to find *N* jobs already present (in queue or in service) is not permitted to enter the system. This control policy is a simple implementation of the accept/reject decision as part of the order review functionality of the ORR mechanism, with job rejection being used to avoid excessive tardiness costs. A diagram showing the role of this control policy, which we will refer to as M/M/1/*N*, is shown in Figure 4.17.

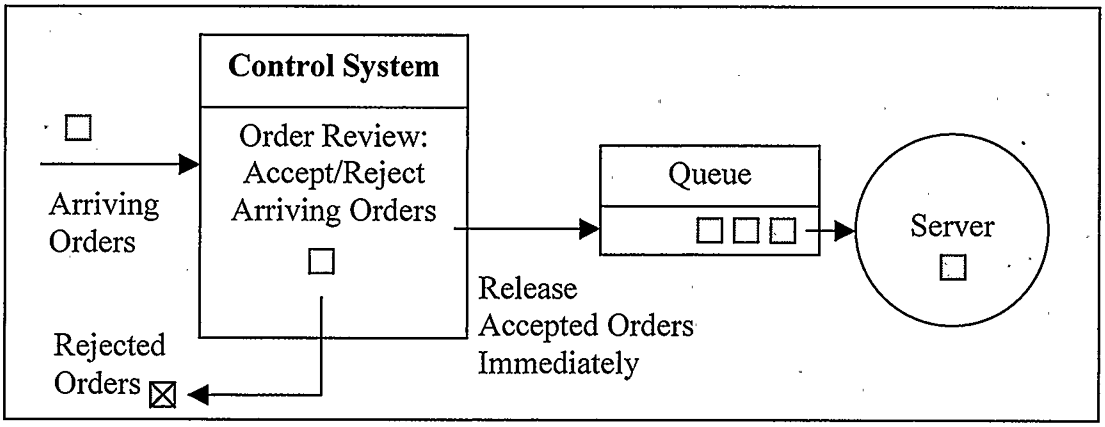

Control System
Order Review: Accept/Reject Arriving Orders
Arriving Orders
Rejected Orders
Release Accepted Orders Immediately
Queue
Server

Figure 4.17: Diagram of the M/M/1/*N* System

4.3.2 Analytical Derivations for the M/M/1/N System

4.3.2.1 Proportion of Jobs Accepted

For an M/M/1/*N* system, the probability of there being *n* jobs in the system is:

$$\pi_n = \frac{\rho^n(1-\rho)}{(1-\rho^{N+1})} \tag{4.21}$$

The proportion of jobs rejected is equal to the probability that the system contains *N* jobs:

$$\Pr(rej) = \pi_N = \frac{\rho^N(1-\rho)}{(1-\rho^{N+1})} \tag{4.22}$$

Therefore, the proportion of jobs accepted is equal to:

$$\Pr(acc) = 1 - \Pr(rej) = 1 - \frac{\rho^N(1-\rho)}{(1-\rho^{N+1})} = \frac{1-\rho^N}{1-\rho^{N+1}} \tag{4.23}$$

Figure 4.18 shows the proportion accepted vs. ρ for multiple values of *N*. We see that as *N* increases, the traffic intensity below which effectively all jobs are accepted increases.

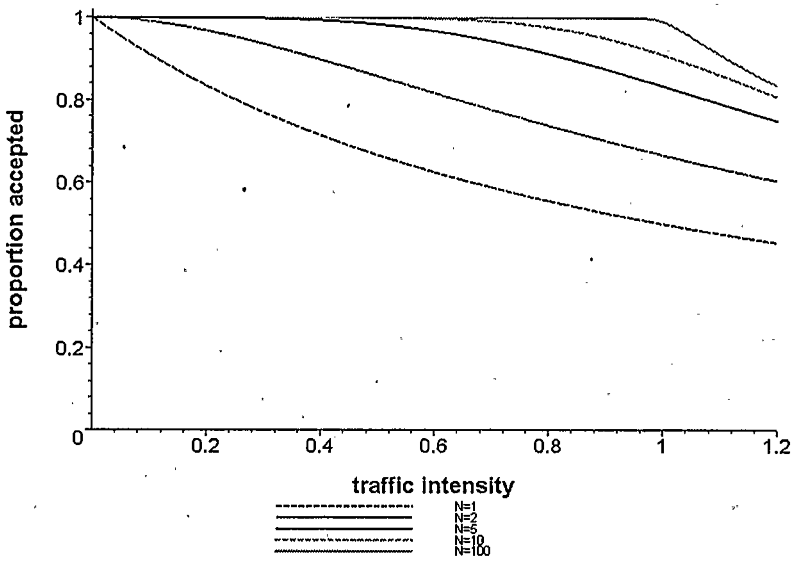


Figure 4.18: Proportion Accepted vs. ρ

4.3.2.2 Expected Tardiness

Given n jobs currently in the system at arrival, the flowtime of an arriving job is the same as for the uncontrolled M/M/1 system, and equals the sum of n+1 service times independently sampled from an exponential distribution with mean $1/\mu$ (provided that $n<N\text{-}1$, otherwise the flowtime is meaningless since the job is rejected):

$$f_{MM1N}(n,t) = \frac{\mu^{n+1}\, t^{n}\, e^{-\mu t}}{n!} \tag{4.24}$$

The expected tardiness of a job arriving when there are n jobs in the system is:

$$AvgTardiness_{MM1N}(n) = \int_{\frac{m}{\mu}}^{\infty} (t - \frac{m}{\mu})\, f_{MM1N}(n,t)\, dt = \int_{\frac{m}{\mu}}^{\infty} (t - \frac{m}{\mu}) \frac{\mu^{n+1}\, t^{n}\, e^{-\mu t}}{n!} dt \tag{4.25}$$

The integral evaluates to:

$$AvgTardiness_{MM1N}(n) = \frac{e^{-m}}{\mu}\left(\frac{m^{n+1}}{n!} - (m-n-1)\sum_{s=0}^{n}\frac{m^s}{s!}\right) \tag{4.26}$$

The average tardiness of all accepted jobs is the weighted sum of the probability of being in state n and the expected tardiness of a job arriving at state n (for all n in which jobs are accepted), divided by the probability of acceptance:

$$AvgTardiness_{MM1N} = \frac{1}{\Pr(acc)} \times \sum_{n=0}^{N-1} \pi_n \cdot AvgTardiness_{MM1N}(n) \tag{4.27}$$

$$AvgTardiness_{MM1N} = \frac{1-\rho^{N+1}}{1-\rho^{N}} \times \sum_{n=0}^{N-1} \frac{\rho^n(1-\rho)}{1-\rho^{N+1}} \frac{e^{-m}}{\mu}\left(\frac{m^{n+1}}{n!} - (m-n-1)\sum_{s=0}^{n}\frac{m^s}{s!}\right) \tag{4.28}$$

The nested summations can be removed, resulting in the expression:

$$AvgTardiness_{MM1N} =$$
$$\frac{e^{-m}}{\mu\left(1-\rho^N\right)}\left(\frac{1}{1-\rho}\sum_{n=0}^{N-1}\left(\frac{(m\rho)^n}{n!}\right) + \frac{\rho^N\left((1-\rho)(m-N)-1\right)}{(1-\rho)}\sum_{n=0}^{N-1}\left(\frac{m^n}{n!}\right) - \frac{(m\rho)^N}{(N-1)!}\right) \tag{4.29}$$

Alternatively, the expression can be expressed without the use of summations using the gamma function:

$$AvgTardiness_{MM1N} =$$
$$\frac{1}{\mu\left(1-\rho^N\right)}\left(\frac{1}{1-\rho}\frac{e^{-m(1-\rho)}\Gamma(N,m\rho)}{\Gamma(N)} + \frac{\rho^N\left((1-\rho)(m-N)-1\right)}{(1-\rho)}\frac{\Gamma(N,m)}{\Gamma(N)} - \frac{e^{-m}(m\rho)^N}{\Gamma(N)}\right) \tag{4.30}$$

In the above equation, Γ(a) is the complete gamma function and Γ(a, x) is the upper incomplete gamma function where for real values of *a*:

$$\Gamma(a) = \int_0^{\infty} t^{a-1} e^{-t}\, dt \tag{4.31}$$

$$\Gamma(a,x) = \int_x^{\infty} t^{a-1} e^{-t}\, dt \tag{4.32}$$

Using the gamma function form instead of the summation form allows for much faster and simpler computation, particularly from within spreadsheets, where implementation of the summation operator is slow and complex.

As traffic intensity approaches zero, the expected tardiness approaches:

$$\lim_{\rho \to 0} AvgTardiness_{MM1N} = \frac{e^{-m}}{\mu} \tag{4.33}$$

This is the same as for the uncontrolled system (and very small relative to the service time for reasonable values of *m*). For the M/M/1/*N* system, traffic intensity is not constrained to being less than 100%. As traffic intensity increases, the expected tardiness converges on:

$$\lim_{\rho \to \infty} AvgTardiness_{MM1N} = \frac{e^{-m} m^{N} - (m-N)\Gamma(N;m)}{\mu \Gamma(N)} \tag{4.34}$$

This is the same as the expected tardiness of a job entering when there are N-1 jobs in the system. This result is unlike the previous systems where expected tardiness increased rapidly to infinity as traffic intensity approached 100%.

Figure 4.19 shows the expected tardiness vs. ρ for multiple values of N. Similar to previous systems, we see that expected tardiness is minimal at lower traffic intensities. We see that for lower values of N, tardiness increases very slowly, even as traffic intensity increases beyond 100%. For higher values of N we see the rate of increase in expected tardiness decreasing as traffic intensity increases beyond 100% (until approaching the value defined in Equation (4.34)).

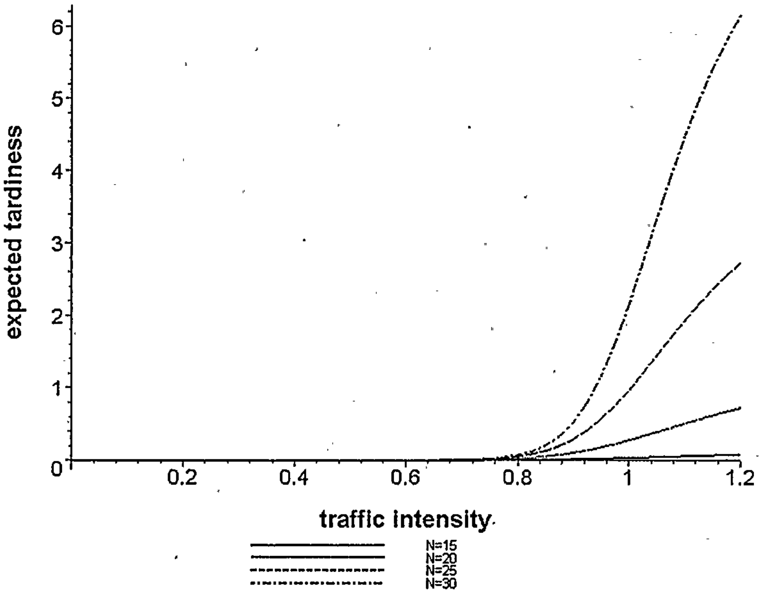


Figure 4.19: Expected Tardiness vs. ρ at Various Values of N (m=20, μ=1)

4.3.2.3 Expected Earliness

The expected earliness of a job arriving when there are n jobs in the system is:

$$AvgEarliness_{MM1N}(n) = \int_0^{\frac{m}{\mu}} (\frac{m}{\mu} - t)\, f_{MM1N}(n,t)\, dt = \int_0^{\frac{m}{\mu}} (\frac{m}{\mu} - t) \frac{\mu^{n+1}\, t^n\, e^{-\mu t}}{n!} dt \qquad (4.35)$$

This simplifies to:

$$AvgEarliness_{MM1N}(n) = \frac{1}{\mu}\left((m-n-1)\left(1 - e^{-m} \sum_{s=0}^{n} \frac{m^s}{s!}\right) + \frac{e^{-m} m^{n+1}}{n!} \right) \qquad (4.36)$$

The average earliness of all accepted jobs is the weighted sum of the probability of being in state n and the expected earliness of a job arriving at state n (for all n in which jobs are accepted), divided by the overall probability of acceptance:

$$AvgEarliness_{MM1N} = \frac{1}{\Pr(acc)} \times \sum_{n=0}^{N-1} \pi_n \cdot AvgEarliness_{MM1N}(n) \qquad (4.37)$$

$$AvgEarliness_{MM1N} = \frac{1-\rho^{N+1}}{1-\rho^N} \times \sum_{n=0}^{N-1} \frac{\rho^n (1-\rho)}{1-\rho^{N+1}} \frac{1}{\mu}\left((m-n-1)\left(1 - e^{-m} \sum_{s=0}^{n} \frac{m^s}{s!}\right) + \frac{e^{-m} m^{n+1}}{n!} \right) \qquad (4.38)$$

We can remove the nested summations and simplify as follows:

$$\begin{aligned} AvgEarliness_{MM1N} = \frac{1}{\mu\left(1-\rho^N\right)} \times (& \frac{\left(1-\rho^N\right)\left(m(1-\rho)-1\right) + \rho^N N(1-\rho)}{(1-\rho)} \\ & - \frac{e^{-m}(\rho m)^N}{(N-1)!} + \frac{e^{-m}}{(1-\rho)} \sum_{s=0}^{N-1} \frac{(\rho m)^s}{s!} + \frac{e^{-m}\rho^N \left((1-\rho)(m-N)-1\right)}{(1-\rho)} \sum_{s=0}^{N-1} \frac{m^s}{s!} \;) \end{aligned} \qquad (4.39)$$

Alternatively, this can be expressed using the gamma function for computational efficiency as follows:

$$AvgEarliness_{MM1N} = \frac{1}{\mu\left(1-\rho^{N}\right)} \times \Bigg(\frac{\left(1-\rho^{N}\right)\left(m\left(1-\rho\right)-1\right)+\rho^{N}N\left(1-\rho\right)}{\left(1-\rho\right)} \\ - \frac{e^{-m}\left(\rho m\right)^{N}}{\left(N-1\right)!} + \frac{e^{-m(1-\rho)}}{\left(1-\rho\right)}\frac{\Gamma\left(N,\rho m\right)}{\Gamma\left(N\right)} + \frac{\rho^{N}\left(\left(1-\rho\right)\left(m-N\right)-1\right)}{\left(1-\rho\right)}\frac{\Gamma\left(N,m\right)}{\Gamma\left(N\right)} \Bigg) \tag{4.40}$$

As traffic intensity approaches zero, the expected earliness becomes:

$$\lim_{\rho \to 0} AvgEarliness_{MM1N} = \frac{e^{-m}+m-1}{\mu} \tag{4.41}$$

This is the same as for the uncontrolled system, and is approximately equal to the flow allowance (*M*) less the mean service time (1/μ). As traffic intensity increases to infinity, the expected earliness becomes:

$$\lim_{\rho \to \infty} AvgEarliness_{MM1N} = \frac{1}{\mu}\left(m-N+\frac{e^{-m}m^{N}-(m-N)\Gamma\left(N,m\right)}{\Gamma\left(N\right)} \right) \tag{4.42}$$

For large values of (*m*-*N*) this is approximately equal to (*m*-*N*)/μ, and for large values of (*N*-*m*) it converges to zero.

Figure 4.20 shows expected earliness vs. ρ for multiple values of *N*. At low traffic intensities, the system work in process limit does not have a significant impact, and expected earliness is similar to that for an uncontrolled system. As traffic intensity increases we see that the lower the system work limit, the greater the earliness. We see that as traffic intensity exceeds 100%, the rate of decrease in expected earliness slows (until it approaches the value defined in Equation (4.42)).

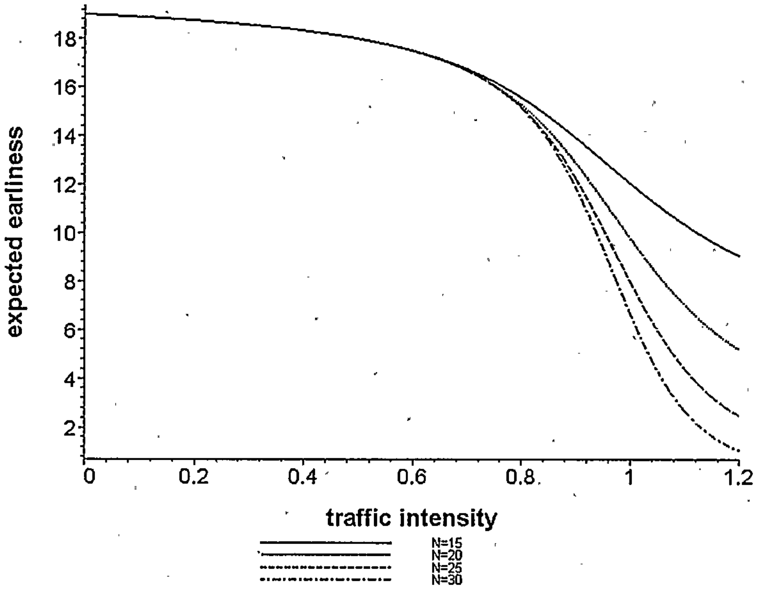


Figure 4.20: Expected Earliness vs. ρ at Various Values of N (m=20, μ=1)

4.3.2.4 The Special Case, ρ=1

The analytical work done previously assumes that the traffic intensity is not equal to 100%, but under the M/M/1/N control policy, traffic intensity is allowed to equal and exceed unity. The following analysis is for the special case when ρ=1, denoted by the subscript 'sp'.

When traffic intensity is equal to 100%, the probability of there being n jobs in the system is:

$$\pi_{n\,sp} = \frac{1}{N+1} \tag{4.43}$$

The proportion of jobs rejected is equal to the probability that the system contains N jobs:

$$\Pr_{sp}(rej) = \pi_{N\,sp} = \frac{1}{N+1} \tag{4.44}$$

Therefore, the proportion of jobs accepted is equal to:

$$\Pr_{sp}(acc) = 1 - \Pr_{sp}(rej) = \frac{N}{N+1} \tag{4.45}$$

The expected tardiness of a job arriving when there are n jobs in the system is still equal to:

$$AvgTardiness_{MM1N}(n) = \frac{e^{-m}}{\mu}\left(\frac{m^{n+1}}{n!} - (m-n-1)\sum_{s=0}^{n}\frac{m^s}{s!}\right) \tag{4.46}$$

The average tardiness of all accepted jobs for the special case is the weighted sum of the probability of being in state n and the expected tardiness of a job arriving at state n (for all n in which jobs are accepted), divided by the probability of acceptance, all for the special case:

$$AvgTardiness_{MM1N\,sp} = \frac{1}{\Pr_{sp}(acc)} \times \sum_{n=0}^{N-1} \pi_{n\,sp} \cdot AvgTardiness_{MM1N}(n) \tag{4.47}$$

$$AvgTardiness_{MM1N\,sp} = \frac{N+1}{N} \times \sum_{n=0}^{N-1} \frac{1}{N+1}\frac{e^{-m}}{\mu}\left(\frac{m^{n+1}}{n!} - (m-n-1)\sum_{s=0}^{n}\frac{m^s}{s!}\right) \tag{4.48}$$

The nested summations can be removed, resulting in the expression:

$$AvgTardiness_{MM1N\,sp} = \frac{e^{-m}}{2N}\left(\frac{m^N(1-m+N)}{(N-1)!} + \left(m^2+N-2mN+N^2\right)\sum_{n=0}^{N-1}\frac{m^n}{n!}\right) \quad (4.49)$$

Alternatively, this can be expressed using the gamma function for computational efficiency as follows:

$$AvgTardiness_{MM1N\,sp} = \frac{\left(e^{-m}m^N(1-m+N)+\left(m^2+N-2mN+N^2\right)\Gamma(N,m)\right)}{2\Gamma(N+1)} \quad (4.50)$$

The expected earliness of a job arriving when there are *n* jobs in the system is still equal to:

$$AvgEarliness_{MM1N}(n) = \frac{1}{\mu}\left((m-n-1)\left(1-e^{-m}\sum_{s=0}^{n}\frac{m^s}{s!}\right)+\frac{e^{-m}m^{n+1}}{n!}\right) \quad (4.51)$$

The average earliness of all accepted jobs for the special case is the weighted sum of the probability of being in state *n* and the expected earliness of a job arriving at state *n* (for all *n* in which jobs are accepted), divided by the overall probability of acceptance, all for the special case:

$$AvgEarliness_{MM1N\,sp} = \frac{1}{\Pr_{sp}(acc)}\times\sum_{n=0}^{N-1}\pi_{n\,sp}\cdot AvgEarliness_{MM1N}(n) \quad (4.52)$$

$$AvgEarliness_{MM1N\,sp} = \frac{N+1}{N}\times\sum_{n=0}^{N-1}\frac{1}{N+1}\frac{1}{\mu}\left((m-n-1)\left(1-e^{-m}\sum_{s=0}^{n}\frac{m^s}{s!}\right)+\frac{e^{-m}m^{n+1}}{n!}\right) \quad (4.53)$$

The nested summations can be removed, resulting in the expression:

$$AvgEarliness_{MM1N\,sp} = \frac{e^{-m}}{2N}\times\left(\frac{m^N\left(1-m+N\right)}{\left(N-1\right)!}+\left(2m-N-1\right)+\left(m^2+N-2mN+N^2\right)\sum_{n=0}^{N-1}\frac{m^n}{n!}\right) \quad (4.54)$$

Alternatively, this can be expressed using the gamma function as follows:

$$AvgTardiness_{MM1N\,sp} = \frac{e^{-m}m^N\left(1-m+N\right)+\left(2m-N-1\right)\Gamma\left(N+1\right)+\left(m^2+N-2mN+N^2\right)\Gamma\left(N,m\right)}{2\Gamma\left(N+1\right)} \quad (4.55)$$

The above results can be used to find all the remaining measures of interest (i.e. PPUAR) for the special case.

4.3.2.5 Profit Per Unit Arriving Revenue

For the M/M/1/*N* system, the profit per unit arriving revenue is:

$$PPUAR_{MM1N} = \Pr\left(acc\right)\times\left(1-\frac{\mu}{k_m\,m}AvgTardiness_{MM1N}-\frac{\mu}{k_m\,k_r\,m}AvgEarliness_{MM1N}\right) \quad (4.56)$$

Using our results from the previous sections, this can be expressed as the following function of our exogenous variables and our control policy parameters:

$$PPUAR_{MM1N=}\frac{1}{\left(1-\rho^{N+1}\right)}\times \quad (4.57)$$

$$\left(\left(1-\rho^N\right)-\frac{\left(1-\rho^N\right)\left(m\left(1-\rho\right)-1\right)+\rho^N N\left(1-\rho\right)}{k_m k_r m\left(1-\rho\right)}-\frac{\left(1+k_r\right)e^{-m}}{k_m k_r m}\times\left(\frac{1}{\left(1-\rho\right)}\sum_{n=0}^{N-1}\frac{\left(m\rho\right)^n}{n!}+\frac{\rho^N\left(\left(1-\rho\right)\left(m-N\right)-1\right)}{\left(1-\rho\right)}\sum_{n=0}^{N-1}\frac{m^n}{n!}-\frac{\left(m\rho\right)^N}{\left(N-1\right)!}\right)\right)$$

Alternatively, this can be expressed using the gamma function instead of the summation operator for computational efficiency as follows:

$$PPUAR_{MM1N} = \frac{1}{\left(1-\rho^{N+1}\right)} \times \left(\left(1-\rho^{N}\right) - \frac{\left(1-\rho^{N}\right)\left(m(1-\rho)-1\right)+\rho^{N}N(1-\rho)}{k_m k_r m(1-\rho)} - \frac{(1+k_r)}{k_m k_r m} \times \left(\frac{e^{-m(1-\rho)}}{(1-\rho)} \frac{\Gamma(N,m\rho)}{\Gamma(N)} + \frac{\rho^{N}\left((1-\rho)(m-N)-1\right)}{(1-\rho)} \frac{\Gamma(N,m)}{\Gamma(N)} - \frac{e^{-m}(m\rho)^{N}}{\Gamma(N)} \right) \right) \tag{4.58}$$

4.3.2.6 Optimal Work Limit

Figure 4.21 shows PPUAR as a function of *N* for three values of ρ (70%, 80%, 90%) when m=16.0944, k_m=0.5, and k_r=5. We see that when traffic intensity is 70%, PPUAR performance rapidly converges to a maximum as *N* increases. At this traffic intensity, *N** is infinite, and optimal performance is equal to that of the uncontrolled system. For traffic intensities of 80% and 90% we observe that PPUAR performance rapidly increases with *N*, and then peaks at a finite *N** before converging on the lower M/M/1 performance value as *N* increases to infinity. Note that improvement over M/M/1 at optimal *N** is much larger for ρ=90% than for ρ=80%. Also note that while under no control PPUAR is higher at 80% than at 90% traffic intensity, the reverse is true under optimal M/M/1/*N* control.

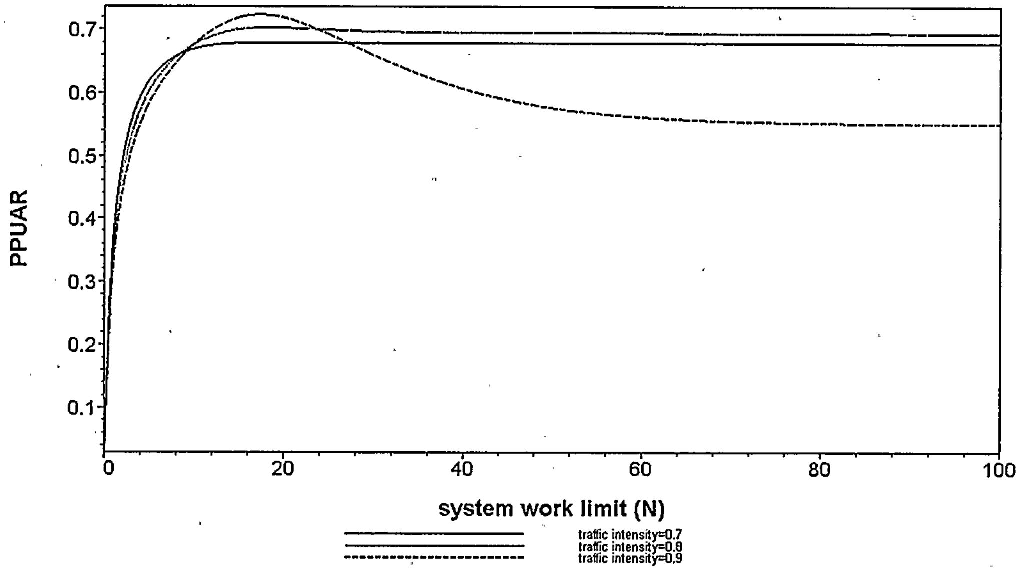


Figure 4.21: PPUAR vs. *N* (m=16.0944, k_m=0.5, k_r=5)

Unfortunately we are unable to find the optimal work limit (*N**) using analytical means. While we can take the derivative of the gamma function form of the PPUAR with respect to *N* (with *N* as a continuous variable), we are unable to solve analytically for the root of the derivative. Furthermore, even if we were able to do so, we would have to numerically compare the values of PPUAR for *N* rounded up and down to the two nearest integer solutions. Consequently, to determine *N**, we implemented a simple exhaustive search algorithm which finds the optimal *N* value by calculating PPUAR exhaustively for all *N* values in a specified range, and returning the *N* value that maximizes PPUAR performance. Algorithm results were checked against graphical and numerical results for a variety of test cases, and were found to be correct for all tested cases. Because of the integer-based search, computational speed is adequate for our purposes, and a maximum

N value of 99 was found to reasonably represent a system work limit of infinity (i.e. the absence of a system work limit).

4.3.2.7 Optimal Work Limit When PPUAR is Negative

Under some combinations of exogenous parameters the M/M/1/*N* control policy is not able to generate a positive PPUAR value (i.e. earliness and tardiness costs result in the system losing money for every job accepted). Under these conditions, there are two possible approaches: (i) setting the system work limit to zero – effectively shutting down the system – and "maximizing" PPUAR at zero; (ii) using the system work limit that results in the best (negative) PPUAR, maximizing performance under the assumption that the system must stay open.

We will choose the first option – rejecting all jobs when PPUAR is negative. – as this is the truly optimal decision under the M/M/1/*N* control policy.

4.4 M/M/1/N Results

In this section we explore the performance of the M/M/1/*N* system under our profit model. First we will explore *N**, then we will explore the performance at *N** for a full range of earliness and tardiness cost factor combinations over a wide range of traffic intensities. Additionally, we will explore the sensitivity of the M/M/1/*N* control policy to non-optimal choices of *N*.

4.4.1 N^* vs. ρ

Figure 4.22 shows N^* vs. ρ for all combinations of low, medium and high relative earliness costs and loose and tight due dates when k_m=1. Figure 4.23 and Figure 4.24 show similar plots for k_m=0.5 and k_m=0.25 respectively. Note that when there is no N^* delineated on the graph, N^* is effectively infinite. The main results of interest from these figures can be summarized as follows:

- For most parameter combinations, N^* is infinite at low traffic intensities. As traffic intensity increases beyond a certain point, the optimal work limit becomes finite, and then gradually decreases as traffic intensity increases towards, and past, 100%.
- Decreasing the flow allowance significantly reduces the traffic intensity at which N^* becomes finite.
- Increasing the severity of earliness costs (i.e. reducing the k_r) results in a marginal increase in the traffic intensity at which N^* becomes finite.
- When the relative earliness cost is high, and the cost magnitude is medium or high, we see that N^* is equal to zero at low traffic intensities. For these conditions the system is not able to operate profitably under this control policy, and we make use of the ability to "close down" the system to halt losses.
- The greater the severity of earliness cost (i.e. the lower k_r), the greater the optimal system work limit. This is because allowing more jobs into the system decreases expected earliness.

- Similarly, increasing the flow allowance also increases N^* so as to reduce the impact of the corresponding increase in expected earliness.

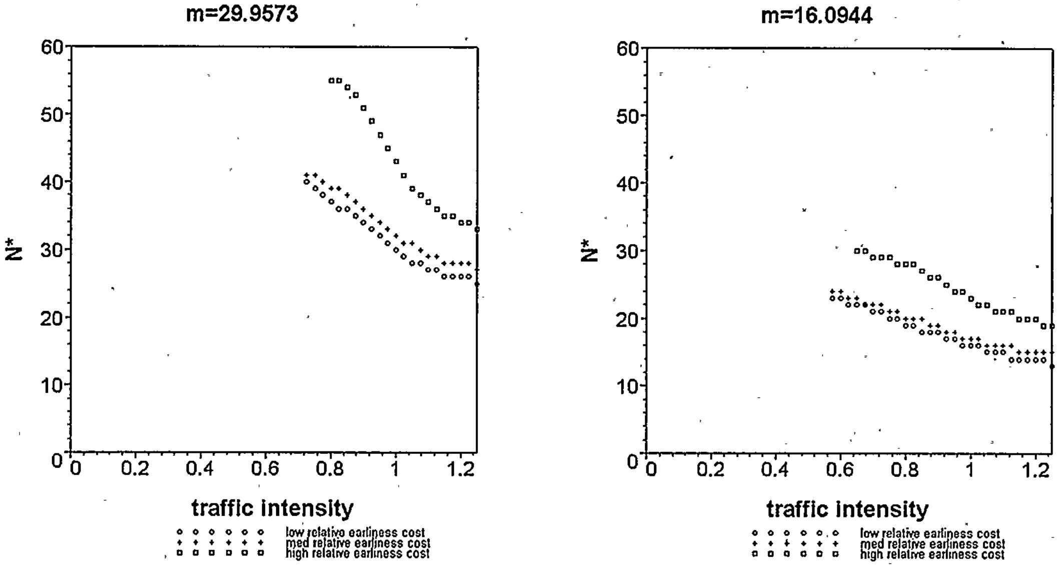


Figure 4.22: N^* vs. ρ for M/M/1/N at $k_m = 1$

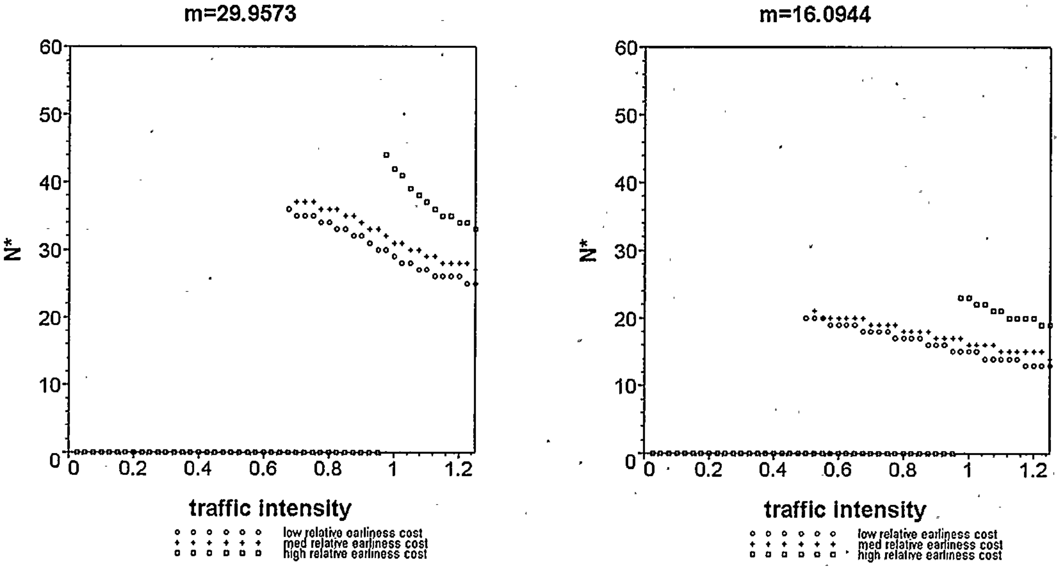


Figure 4.23: N^* vs. ρ for M/M/1/N at $k_m = 0.5$

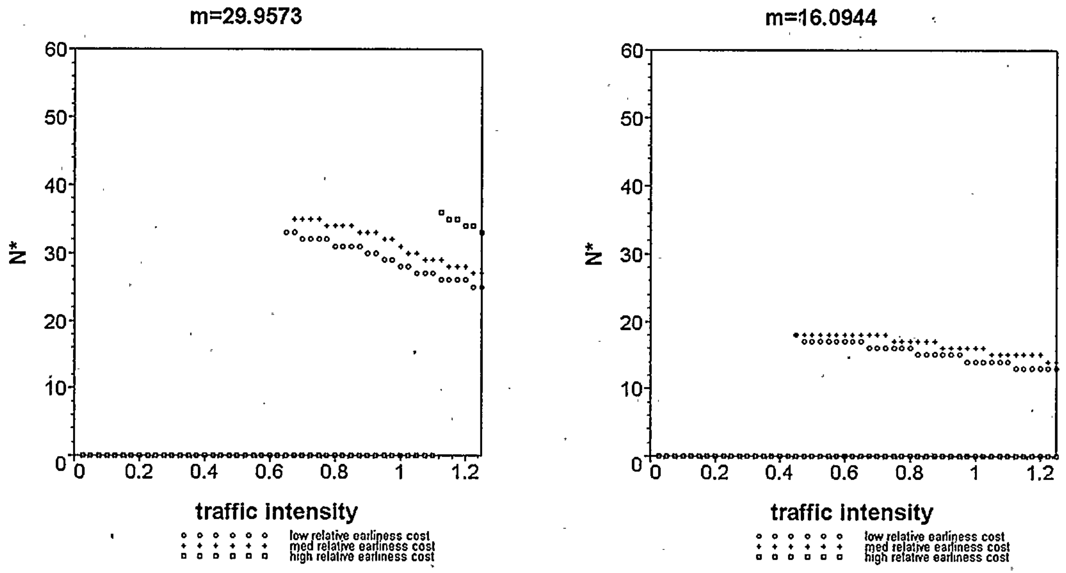


Figure 4.24: N^* **vs.** ρ **for M/M/1/N at** $k_{\mathrm{m}} = 0.25$

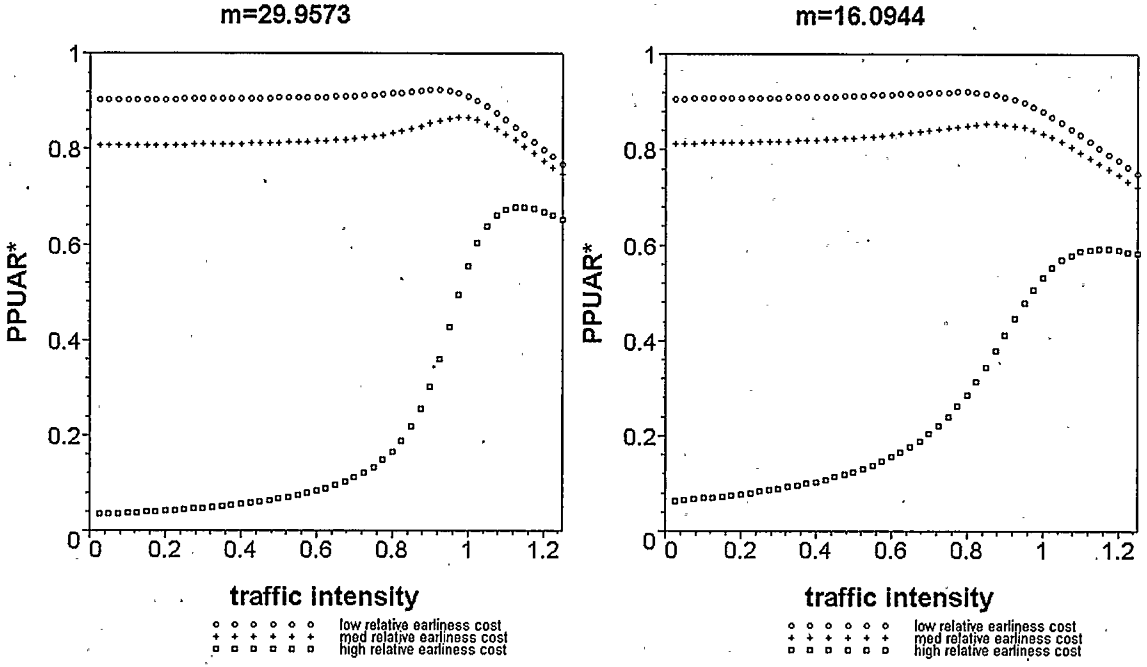
4.4.2 PPUAR vs. ρ*

Figure 4.25 shows how PPUAR at optimal N varies with traffic intensity when k_m=1, for all other explored combinations of parameter values. Figure 4.26 and Figure 4.27 show similar plots for k_m=0.5 and k_m=0.25 respectively. The main results of interest from these figures can be summarized as follows:

- Performance under low traffic intensity is identical to that for the uncontrolled M/M/1 system where N^* is infinite. Where N^* is zero, performance at low traffic intensities is better than uncontrolled (i.e. the system has been shut down to prevent losses).
- At high traffic intensities, performance is significantly better than for the uncontrolled M/M/1 system. Instead of PPUAR rapidly plunging as traffic

intensity approaches 100%, PPUAR slowly declines as traffic intensity increases past 100%.

- Under medium cost magnitude and high relative earliness cost, *N** does not become non-zero until traffic intensity is very close to 100%. Under high cost magnitude and high relative earliness cost, *N** does not become non-zero until traffic intensity is well over 100%[9].


m=29.9573

m=16.0944

Figure 4.25: PPUAR* vs. ρ for M/M/1/*N* at $k_m = 1$

[9] For a tight flow allowance, a positive PPUAR cannot be achieved until traffic intensity exceeds 134% (outside of the range plotted).

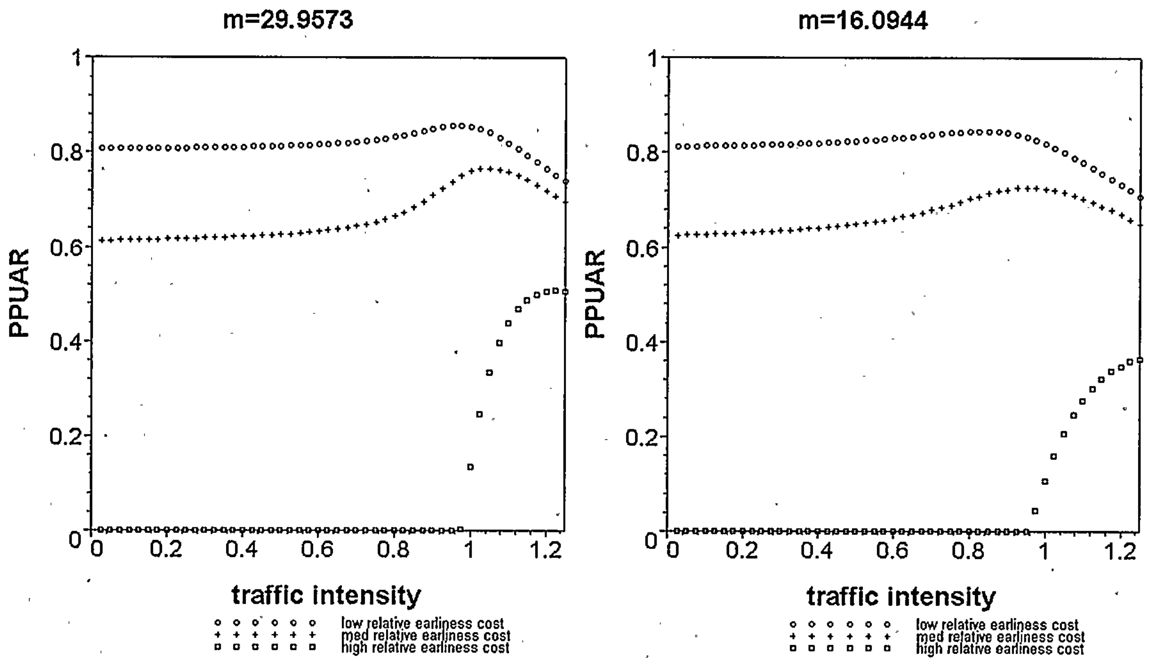


Figure 4.26: PPUAR* vs. ρ for M/M/1/N at $k_m = 0.5$

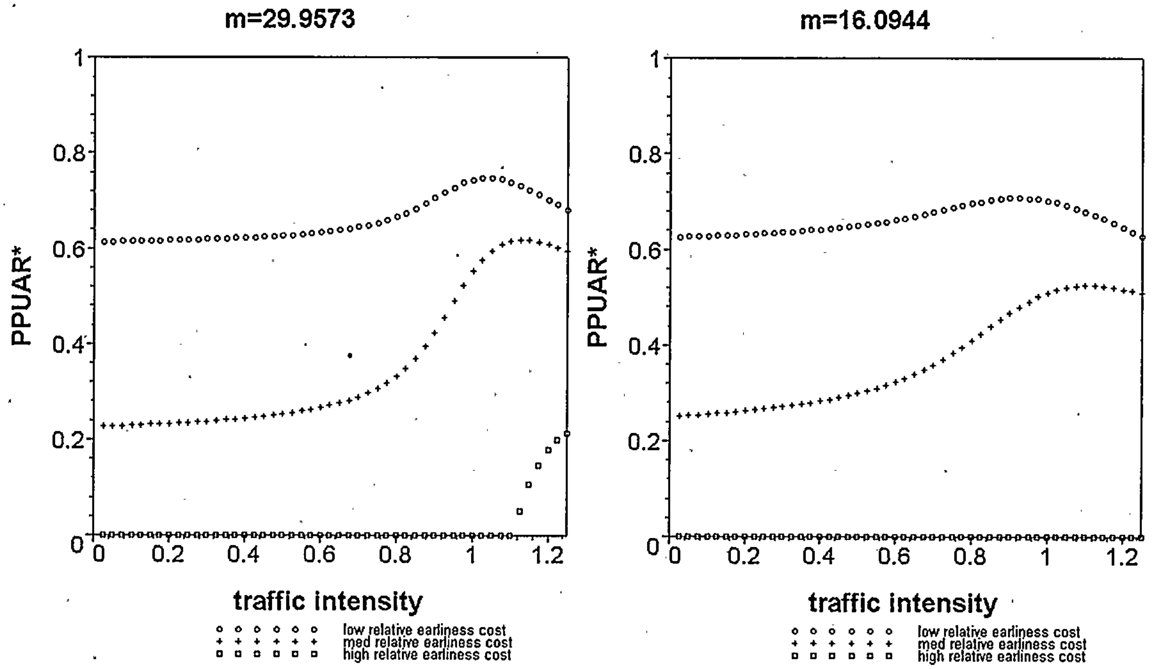


Figure 4.27: PPUAR* vs. ρ for M/M/1/N at $k_m = 0.25$

4.4.3 Cost Breakdown

Figure 4.28 shows the optimal cost per unit arriving revenue (CPUAR*) for the low cost parameter combination, in total and broken up into its three components: (i) rejection CPUAR; (ii) earliness CPUAR; (iii) tardiness CPUAR. Figure 4.29 shows a similar plot for the medium cost parameter combination. Recall that for the high cost combination N^* is zero for the range of traffic intensities investigated, and CPUAR is therefore 100% (and entirely due to the rejection component). The main results of interest from these figures can be summarized as follows:

- For the low and medium cost scenarios, all cost at low traffic intensities is due to earliness.
- Tardiness costs are well controlled by the M/M/1/N control policy, and are never the dominant cost.
- At high traffic intensities, tardiness costs form a "sawtooth" pattern – repeatedly falling suddenly after rising slowly. This occurs because N^* is an integer and changes suddenly.
- Unlike for the uncontrolled and M/M/1 – d control policies, earliness costs continue to be a significant factor even at high traffic intensities (approaching and exceeding 100%).
- Rejection costs climb rapidly as traffic intensity increases past 100%.

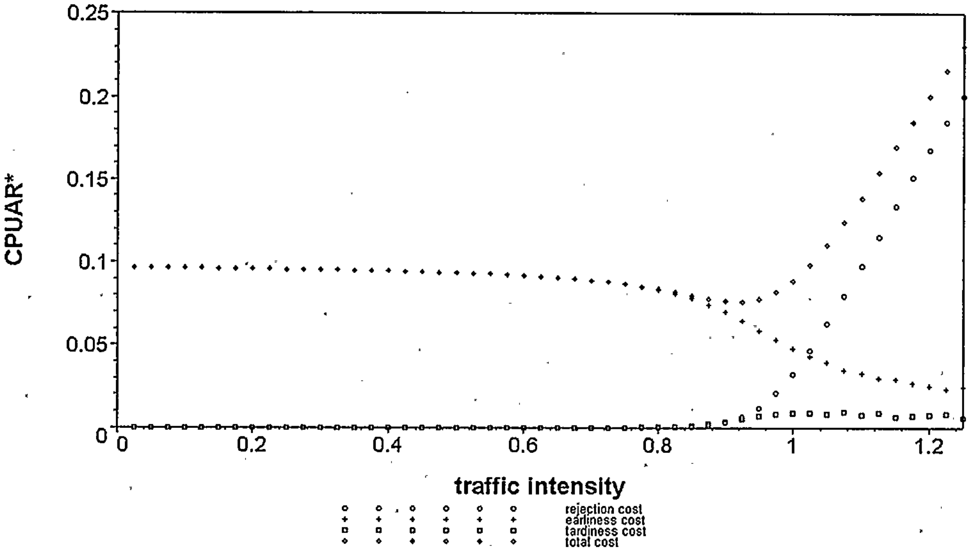


Figure 4.28: CPUAR* vs. ρ for the Low Cost Parameter Combination

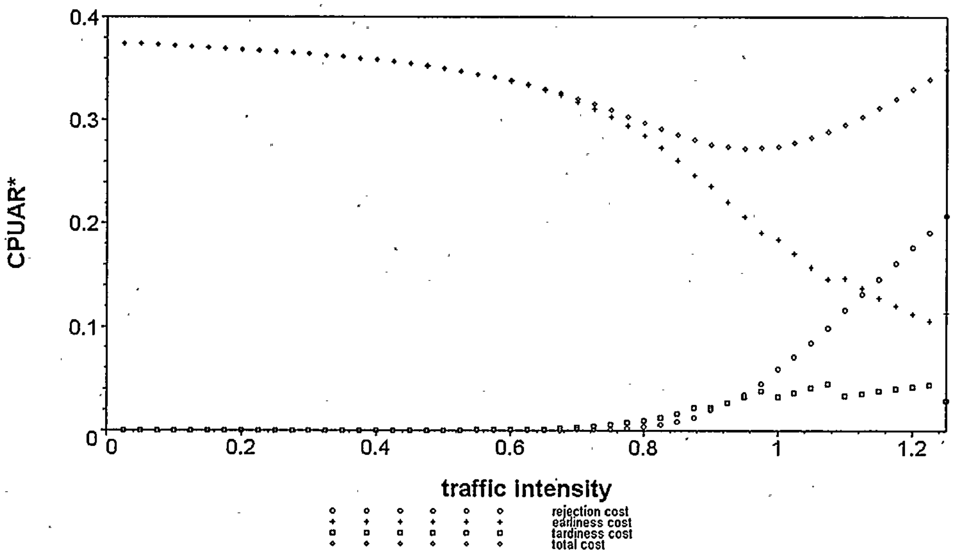


Figure 4.29: CPUAR* vs. ρ for the Medium Cost Parameter Combination

4.4.4 Absolute Profit Rate

Figure 4.30 shows how the absolute profit rate varies with traffic intensity when k_m=1 for all other explored combinations of parameter values. Figure 4.31 and Figure 4.32 show similar plots for k_m=0.5 and k_m=0.25 respectively. Nowhere on any of the curves does the absolute profit rate decrease with increasing traffic intensity. Note that this occurs despite the fact that PPUAR* itself decreases at higher traffic intensities.

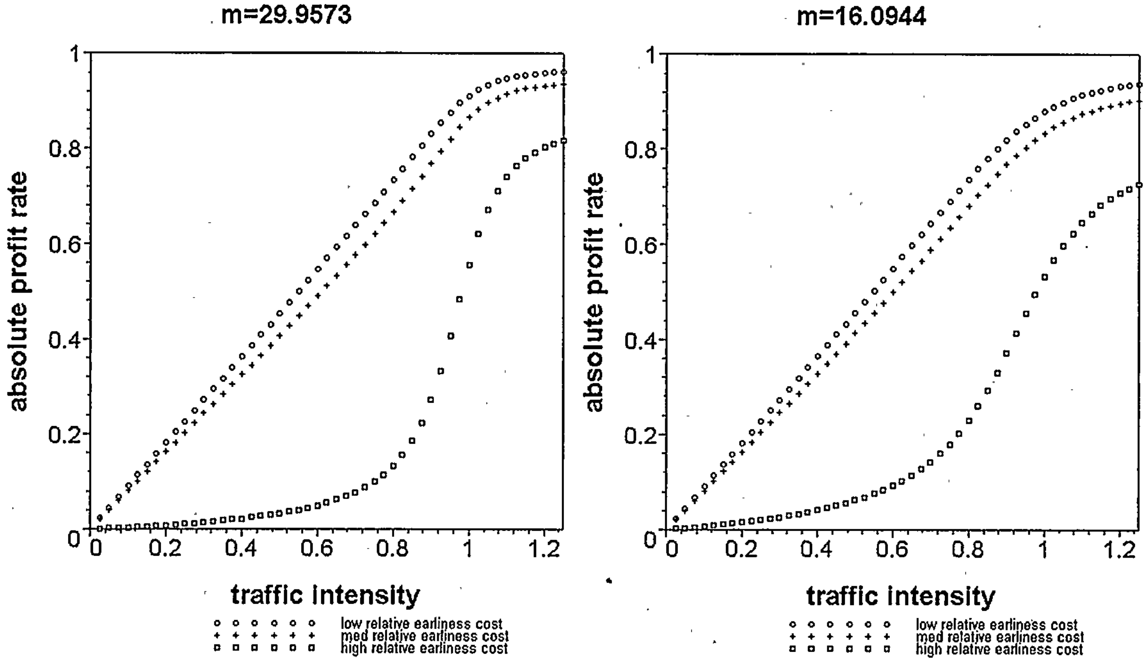


Figure 4.30: Absolute Profit Rate vs. ρ for M/M/1/N at k_m = 1

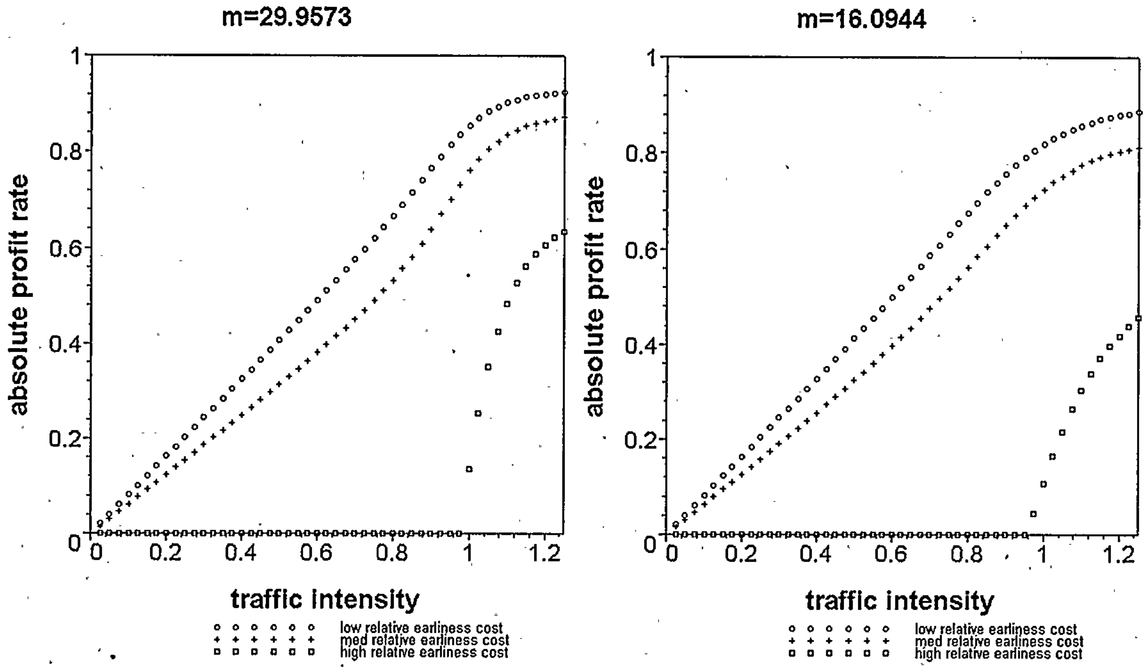


Figure 4.31: Absolute Profit Rate vs. ρ for M/M/1/N at $k_{\mathrm{m}} = 0.5$

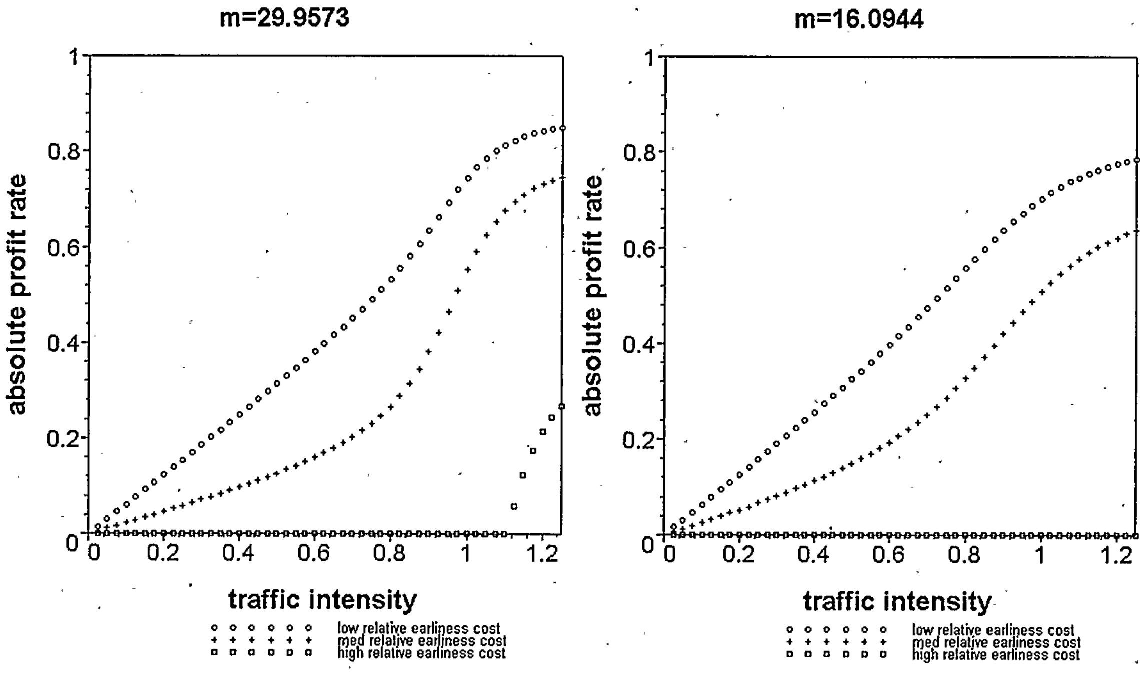


Figure 4.32: Absolute Profit Rate vs. ρ for M/M/1/N at $k_{\mathrm{m}} = 0.25$

4.4.5 Sensitivity of PPUAR Performance to Non-Optimal N

We wish to explore the sensitivity of PPUAR performance under the M/M/1/*N* control policy to non-optimal values of *N*. Figure 4.33 shows how PPUAR performance varies with *N* under the medium cost parameter combination when traffic intensity is 85% (for these parameters, *N** is equal to 18), with the horizontal line illustrating PPUAR performance without any control. Figure 4.34 shows similar results for a traffic intensity of 90% (*N** is 17).We observe that:

- PPUAR performance is reasonably flat around *N**.
- Overestimating *N** cannot result in PPUAR performance worse than that for the uncontrolled system.
- Underestimating *N** can result in PPUAR performance worse than that for the uncontrolled system. The smaller the maximum potential improvement of the M/M/1/*N* policy over the uncontrolled M/M/1, the greater the risk of using a too-low *N* that results in degraded, instead of improved, performance.

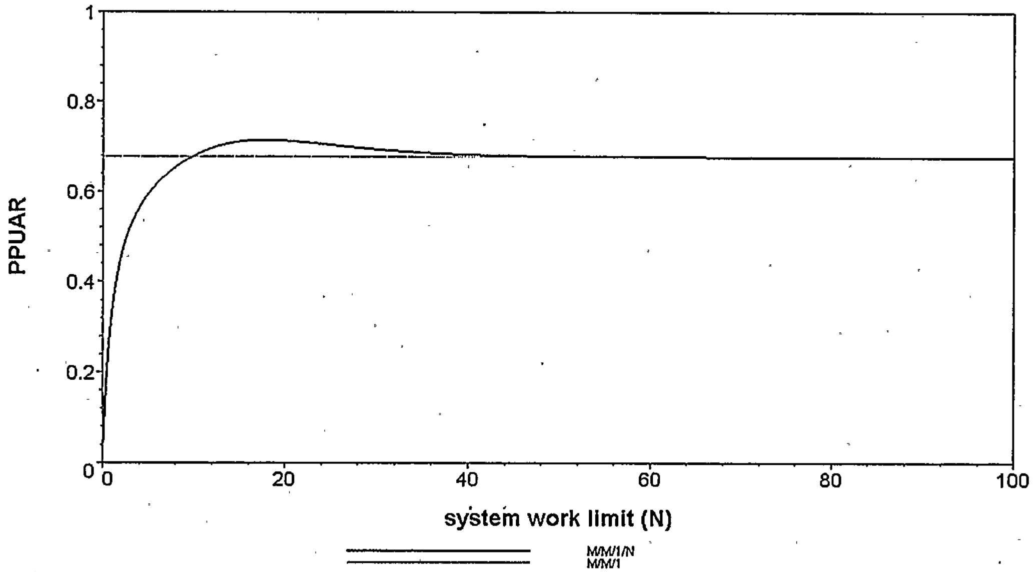


Figure 4.33: PPUAR vs. *N* Under Medium Cost Parameters, ρ=85%

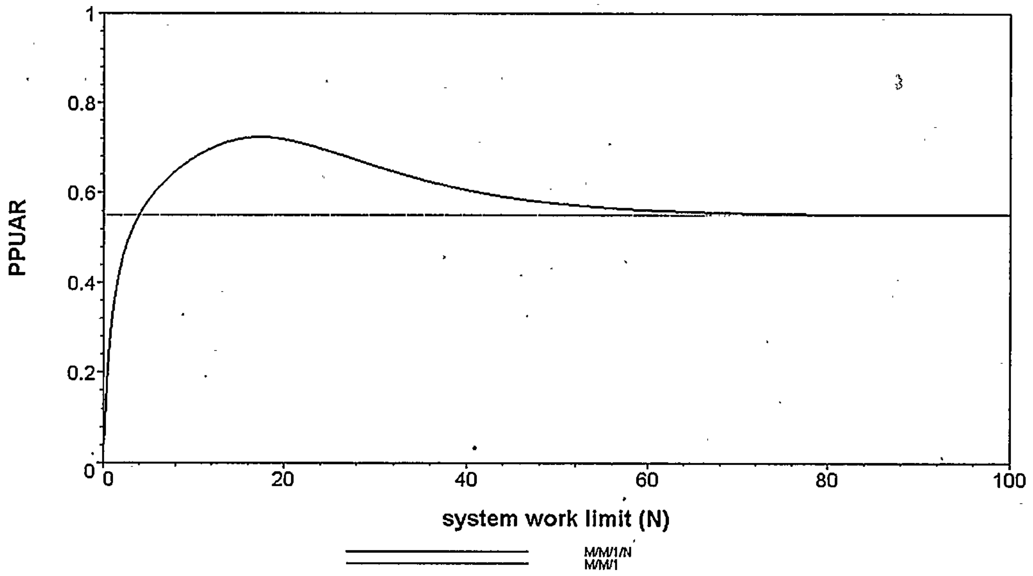


Figure 4.34: PPUAR vs. *N* Under Medium Cost Parameters, ρ=90%

4.4.6 Summary of Results

We can summarize the results of the analysis of the M/M/1/*N* system as follows:

1. Where uncontrolled PPUAR is positive, the M/M/1/*N* control policy is not effective at low and medium traffic intensities where earliness costs dominate. Under these conditions *N** is infinite and performance is identical to that for the uncontrolled M/M/1. However, when high earliness costs result in negative PPUAR performance at low traffic intensities under no control, an *N** value of zero can be used to minimize losses by "shutting down" the system.

2. As traffic intensity increases, at some point *N** becomes finite, and PPUAR exceeds that for the uncontrolled system. While uncontrolled PPUAR drops precipitously as traffic intensity approaches 100%, under M/M/1/*N* control PPUAR remains stable.

3. When present, performance improvement is achieved by incurring rejection costs in exchange for a greater reduction in tardiness costs. As a side-product of lowering system congestion, earliness costs will also increase when rejection costs are present. This is obviously only beneficial when high tardiness costs are present, which, as we saw for the uncontrolled M/M/1 system, occurs only at high traffic intensities.

4. The absolute profit rate is non-decreasing with increasing traffic intensity even where PPUAR is decreasing. This is desirable because it means that additional customers can only increase profits.

5. PPUAR performance is reasonably flat around N^*, and it is impossible for a higher-than-optimal N to result in worse performance than that for the uncontrolled system[10]. It is possible for a too-low N to cause degraded performance when compared with that for no control.

The results for the M/M/1/N analysis show that the accept/reject decision addresses situations with high tardiness costs, which the release delay cannot effectively control. We see that the when the system work limit is in effect, high tardiness costs are significantly lowered in exchange for a increase in earliness costs, and lost potential revenues due to rejection. Selective rejection is incapable of improving performance where earliness costs are dominant. Subsequent control policies will explore the use of the accept/reject decision in conjunction with a release delay.

[10] Except for the special case when N^* is equal to zero.

Chapter 5

Experimental Control Policies

We have so far determined that use of a fixed release delay is effective in improving PPUAR performance at low traffic intensities, and the use of a work limit is effective in improving PPUAR performance at higher traffic intensities. However, used independently, neither the release delay nor the system work limit alone is effective over the entire range of traffic intensities.

In this chapter we will investigate two control policies that make simultaneous use of order release and accept/reject mechanisms. These polices are not analytically tractable, and will therefore be explored using discrete-event simulation. The first control policy to be investigated, which will be referred to as the M/M/1/$N-d$ policy, uses both a fixed release delay and a system work in process limit. The second control policy, which will be referred to as the M/M/1$N-d-RL$ policy, is similar to the M/M/1/$N-d$ system, but allows early release from the order release pool when a certain pool size is exceeded and the server is idle.

5.1 M/M/1/N – d Description and Results

5.1.1 Control System Description

Here we model a complete ORR mechanism, implementing an order review component through a system work limit (N), and an order release component through a fixed delay ($D=d/\mu$) in a pre-release pool for all accepted jobs (i.e. this policy involves two control parameters, d and N). Note that the limit on system work in process now includes both released jobs and those in the pre-release pool. This policy is not explored analytically, but is implemented using the simulation model described in Appendix B. A diagram showing the role of this control policy, which we will refer to as M/M/1/$N – d$, is shown in Figure 5.1.

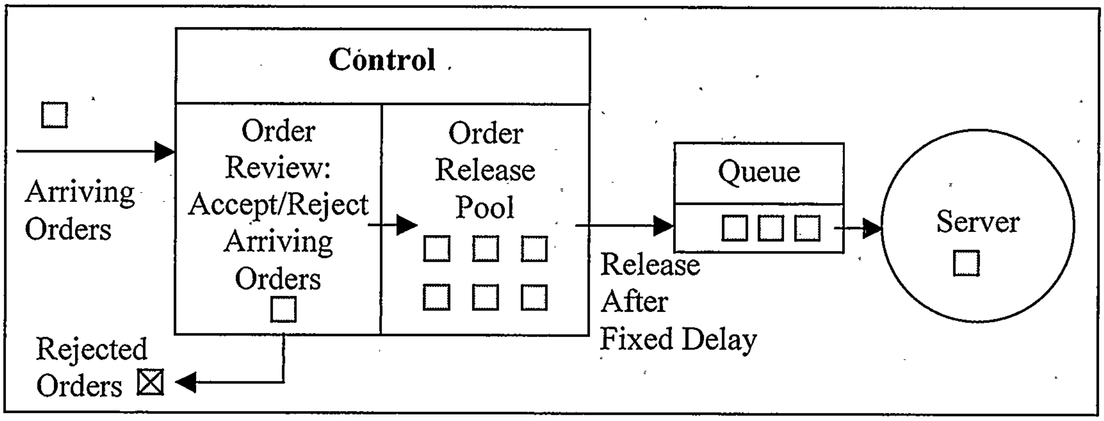

Control
Arriving Orders
Order Review: Accept/Reject Arriving Orders
Order Release Pool
Queue
Server
Release After Fixed Delay
Rejected Orders ☒

Figure 5.1: Diagram of the M/M/1/*N – d* System

In the following sections we explore the performance of the M/M/1/$N-d$ control policy under our cost model when optimal control parameters are used. We wish to determine if the addition of a work limit results in performance improvement under conditions where M/M/1 – d appears to perform well. Additionally, we wish to determine if addition of a release delay results in performance improvement under conditions where M/M/1/N appears to perform well. We also wish to examine the sensitivity of PPUAR performance to deviation in the values of the control parameters from their optimal values.

5.1.2 Optimal Control Parameters (N^*, d^*)

5.1.2.1 Finding the Optimal Control Parameters

Because of the time-intensive nature of simulation (in comparison to analytical math models) finding the optimal set of control parameters for a specific set of values of the exogenous parameters (ρ, m, k_{m}, k_{r}) is not a straightforward task. The optimum-seeking simulation tool *OptQuest for Arena* was used to find the optimal set of control parameters (N^*, d^*). *OptQuest for Arena* is a tool which makes use of meta-heuristics (including tabu-search and genetic algorithms) in order to seek the optimum set of control parameters given an objective function (based on model outputs), linear constraints on the control parameters, and restrictions on specified output measures. *OptQuest* makes use of an existing *Arena* model by:

1. Feeding a potential set of control parameters into the Arena model by changing the value of the model variables.

2. Prompting Arena to run the model for the specified number of replications, and recording the average of the output measures of interest for the replications.
3. Analyzing the results of the simulation and using its heuristic search procedures to generate a new set of potential control parameters.
4. Repeating this process as many times as it is allotted, with the goal of finding the set of valid control parameters that maximizes (or minimizes) the objective function and satisfies all restrictions on output measures.

For a more detailed description of *OptQuest for Arena* see Rockwell Software (2000), Kelton *et al.* (2002), and Rogers (2002).

In order to reduce the amount of analysis, we explore only the low, medium, and high cost parameter combinations discussed previously in Section 3.4.2. Additionally, we will explore a smaller number of traffic intensity values for each parameter combination.

The optimal control parameter set at each traffic intensity for each cost parameter combination was found in two stages: (i) a preliminary broad *OptQuest* search of low precision; (ii) a second more focused search of higher precision.

For the preliminary broad search, control parameters were permitted to vary over their entire valid range (0 to *m* for *d*, 1 to 99 for *N*). Although the release delay *d* is a continuous-valued parameter, a step size of 0.1 was used. With the mean service time set to 1 minute[11], a single short simulation run length of 200 days was used in addition to a 10-day warm up period. *OptQuest* was allowed 100 simulation runs to find the optimal set of control parameters at each explored traffic intensity for each exogenous parameter combination. The results of the preliminary broad search were used to guide the second, more focused, search.

For the second, more focused, search a longer single-replication of 1000 days plus a warm-up period of 50 days was used. The control parameters were allowed to vary +/- two steps from the optimal values generated in the preliminary search (for N each step is 1, for d each step is 0.1). *OptQuest* was allowed to search this tighter parameter space exhaustively. If one or both of the optimal control parameters were found to lie on the edge of the searched spaced (i.e. the highest or lowest permitted value), the space was expanded in the appropriate direction, and again exhaustively searched. The optimal control parameter set resulting from the second search was then used to generate the optimal performance results as presented below in Section 5.1.3.

5.1.2.2 Optimal Control Parameters (N^*, d^*) vs. ρ

Figure 5.2 shows N^* and d^* vs. ρ for the low cost parameter combination under the M/M/1/$N - d$ control policy. Figure 5.3 and Figure 5.4 show similar plots for the medium and high cost parameter combinations respectively. N^* values are delineated on the left-

[11] This results in 1440 expected arrivals per day at 100% traffic intensity.

hand vertical axis, and d^* values are on the right-hand vertical axis. Note that where there is no N^* delineated on the graph, N^* is effectively infinite. The main results of interests from these figures can be summarized as follows:

- d^* decreases with increasing traffic intensity. For the low cost parameter combination, d^*decreases significantly (from 26.7 to 2) as traffic intensity goes from 25% to 99%. As the cost parameter combination becomes more severe, the d^* vs. ρ curve flattens considerably. For the high cost parameter combination, d^* only decreases from 15.2 to 14.2 as traffic intensity goes from 25% to 99%.
- N^* also decreases with traffic intensity. For both the low and high cost parameter combinations, N^* is effectively infinite at traffic intensities at and below 50%[12]. For the medium cost parameter combination N^* is finite at traffic intensities at and below 50%, however, the improvement in PPUAR over that of M/M/1 – d system for these traffic intensities is extremely small (and smaller than the 95% confidence interval of the simulated PPUAR value), and we suspect that the true N^* is in fact infinite.
- For the medium and high cost parameter combinations, N^* is extremely flat at traffic intensities above 75%. N^* decreases with increasing cost severity.

[12] Because of the low resolution along the ρ-axis, we do not know where between 50% and 75% N^* becomes finite.

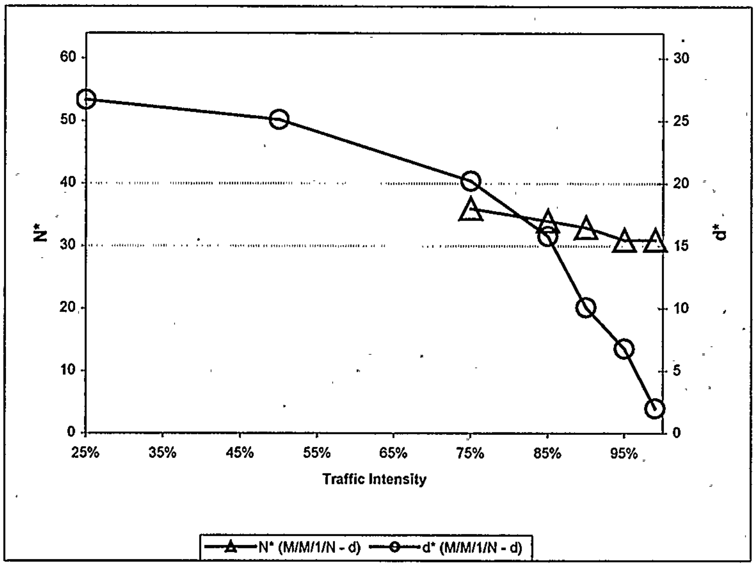


Figure 5.2: (*N, *d**) vs. *ρ* for the Low Cost Parameter Combination**

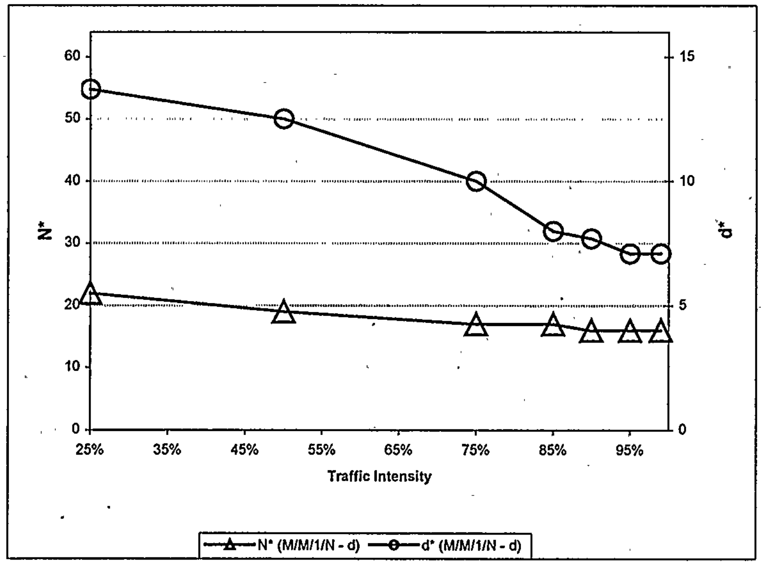


Figure 5.3: (*N, *d**) vs. *ρ* for the Medium Cost Parameter Combination**

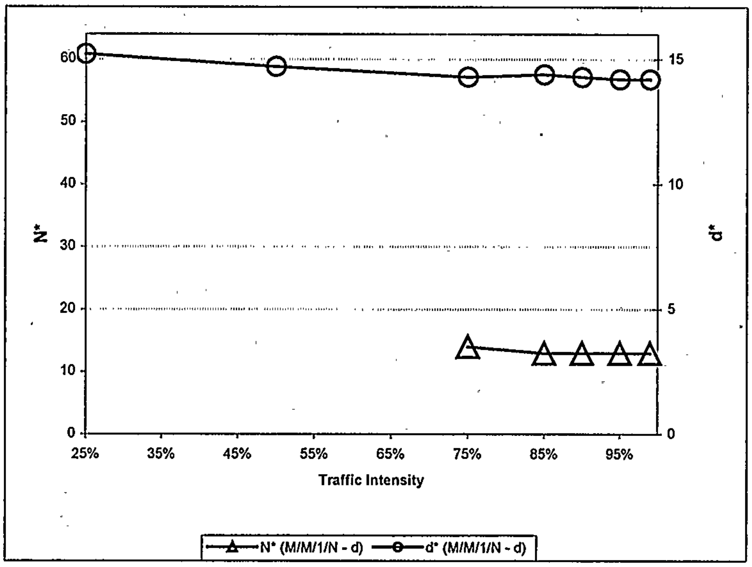


Figure 5.4: (N^*, d^*) vs. ρ for the High Cost Parameter Combination

It is of interest to compare the N^* and d^* under the M/M/1/$N-d$ policy to N^* under the M/M/1/N policy and d^* under the M/M/1 $-d$ policy. Figure 5.5, Figure 5.6, and Figure 5.7 add the N^* for the M/M/1/N control policy and d^* for the M/M/1 $-d$ control policy to Figure 5.2, Figure 5.3, and Figure 5.4 respectively. The main results of interest from these figures can be summarized as follows:

- At low traffic intensities, the d^* for the M/M/1/$N-d$ policy is the same as for the M/M/1 $-d$ system. However, d^* for the M/M/1/$N-d$ system does not rapidly plunge to zero as traffic intensity increases as it does for the M/M/1 $-d$ policy.

- Where N^* is non-zero under the M/M/1/N policy[13], N^* under the M/M/1/$N-d$ policy is slightly lower than for the M/M/1/N policy, with the shapes of the N^* vs. ρ curves being very similar over all traffic intensities.
- N^* was zero under all traffic intensities for the high-cost parameter combination under M/M/1/N control (because earliness costs could not be controlled, the system was "shut down"). However, under M/M/1/$N-d$ control, N^* is non-zero indicating that the system can be operated profitably under all traffic intensities and under all cost parameter combination (which we will see in the next section).

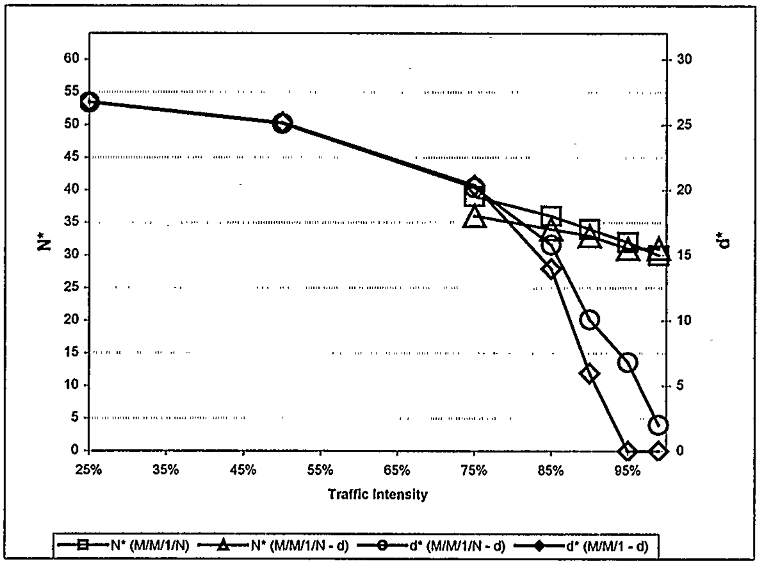


Figure 5.5: Optimal Control Parameters for M/M/1/$N-d$, M/M/1/N and M/M/1 – d vs. ρ for the Low Cost Parameter Combination

[13] And assuming that performance for the finite N^* values at ρ=25% and ρ=50% is effectively equivalent to that for infinite N.

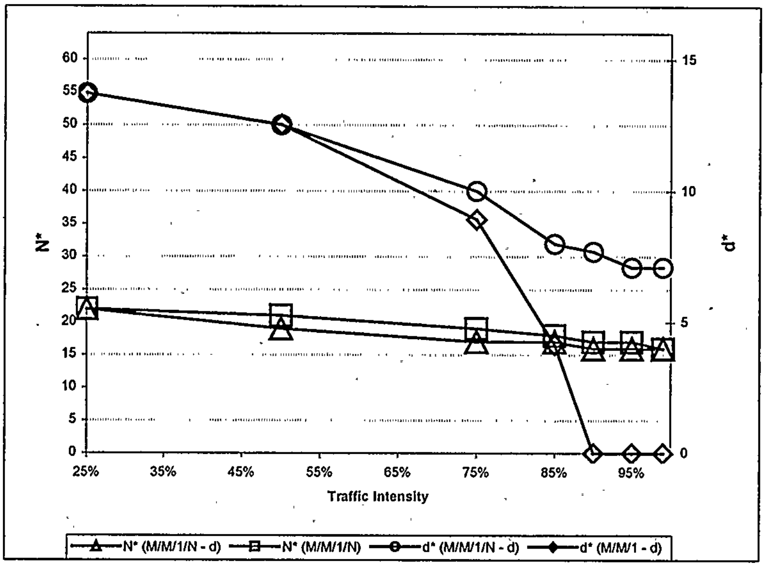


Figure 5.6: Optimal Control Parameters for M/M/1/*N* – *d*, M/M/1/*N* and M/M/1 – *d* vs. ρ for the Medium Cost Parameter Combination

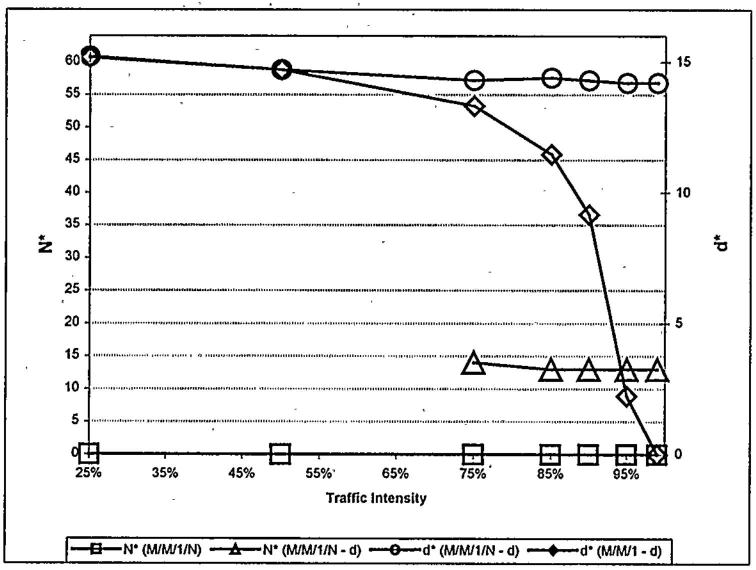


Figure 5.7: Optimal Control Parameters for M/M/1/*N* – *d*, M/M/1/*N* and M/M/1 – *d* vs. ρ for the High Cost Parameter Combination

5.1.3 PPUAR vs. ρ*

Once the optimal control parameter sets were found for each cost parameter combination at each traffic intensity investigated, a set of longer multiple-replication simulations was used to generate more precise performance results. *Arena*'s *Process Analyzer*[14] (PAN) batch simulation tool was used to run a 50-replication experiment, with each replication having a run length of 1000 days (plus a warm-up period of 50 days), at each traffic intensity/cost parameter combination. These multiple-replication experiments yielded very high confidence in the final PPUAR values (a 95% confidence interval half-width between 0.0014% and 0.0452% depending on the exogenous parameters). Confidence intervals are tight enough that error bars are not usefully visible on any of the following graphs.

Figure 5.8 shows PPUAR* vs. ρ for the M/M/1/$N - d$ policy under the low cost parameter combination. The optimal PPUAR for the M/M/1/N and M/M/1 $- d$ policy are also included on the graph. Figure 5.9 and Figure 5.10 show similar plots for the medium and high cost parameter combinations respectively. The main results of interests from these figures can be summarized as follows:

- Under M/M/1/$N - d$ control, optimal PPUAR decreases as traffic intensity increases from zero, but only slowly (not precipitously).

[14] PAN is a tool used to manage the evaluation of many input parameter alternatives. PAN presents input parameters and corresponding output measures in a sortable table, and allows for the batch execution of simulation experiments. For more information on PAN see Section 5.8.5 of Kelton *et al.*(2002).

- For low and high cost parameter combinations, at low traffic intensities, optimal PPUAR is the same for the M/M/1/$N-d$ policy as for the M/M/1 $-d$ policy (we saw in the previous section that the d* values are the same and that N* is effectively infinite). Under the medium cost parameter combination, when ρ=25% or ρ=50% , N* under the M/M/1/$N-d$ policy is finite, whereas it is infinite under M/M/1 $-d$ control. However, there is a negligibly small improvement over the M/M/1 $-d$ policy at these low traffic intensities (i.e. 0.0012% improvement in PPUAR at 50% traffic intensity). Because this improvement is significantly smaller than the 95% confidence half-width of PPUAR under M/M/1/$N-d$ control, performance is equivalent to that for infinite N.
- For the low cost parameter combinations, at high traffic intensities, optimal PPUAR under the M/M/1/$N-d$ policy converges towards the optimal PPUAR for the M/M/1/N policy.
- The improvement in performance of the M/M/1/$N-d$ policy over the better of the M/M/1 $-d$ and the M/M/1/N policy is greater as costs increase. Additionally, the range of traffic intensities over which performance is better increases in width as costs increase. It is interesting to note that this range of traffic intensities where performance improves significantly (75% to 95%) is also the range where most real manufacturing systems likely wish to operate.

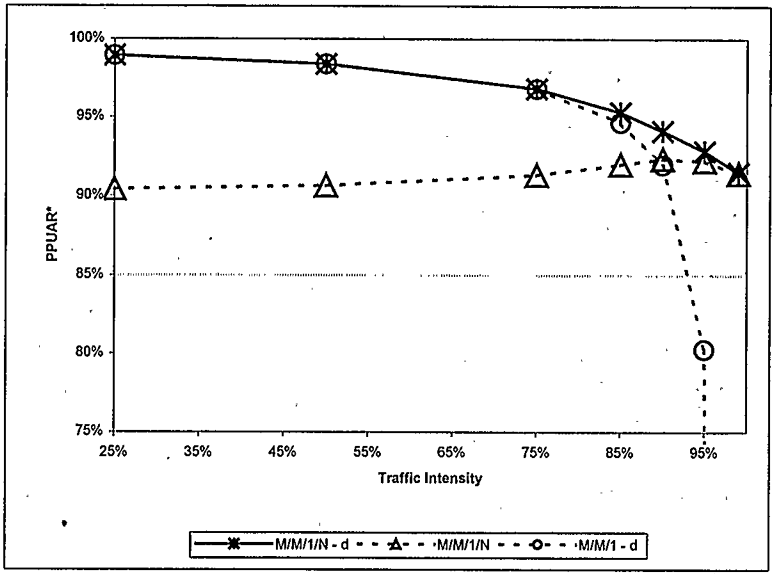


Figure 5.8: PPUAR* for M/M/1/*N* – *d*, M/M/1/*N* and M/M/1 – *d* Under the Low Cost Parameter Combination

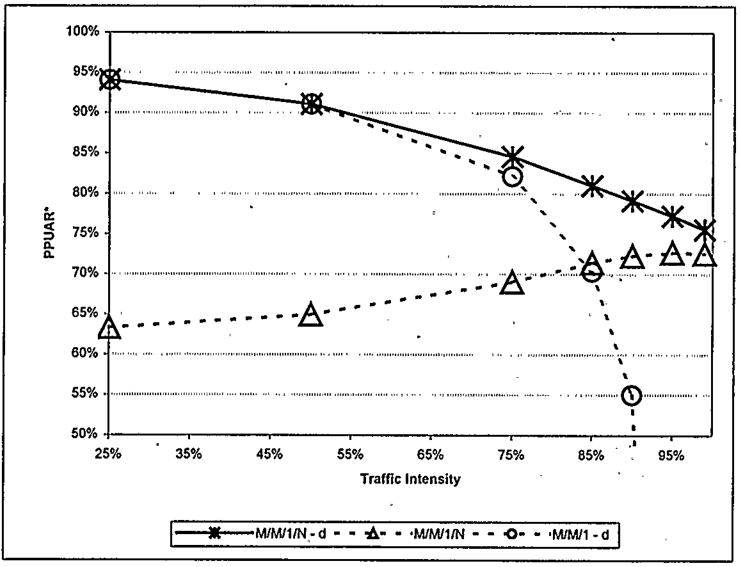


Figure 5.9: PPUAR* for M/M/1/*N* – *d*, M/M/1/*N* and M/M/1 – *d* Under the Medium Cost Parameter Combination

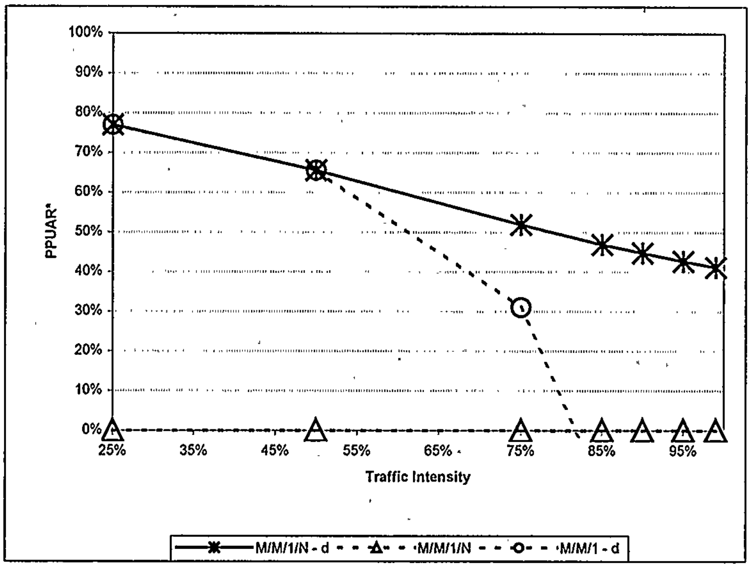


Figure 5.10: PPUAR* for M/M/1/$N-d$, M/M/1/N and M/M/1 – d Under the High Cost Parameter Combination

5.1.4 Cost Breakdown

Figure 5.11 shows the optimal cost per unit arriving revenue for the low cost parameter combination under M/M/1/$N-d$ control, in total and broken up into its three components. Figure 5.12 and Figure 5.13 show similar plots for the medium and high cost parameter combinations. The main results of interest from these figures can be summarized as follows:

- Rejection costs are zero at low traffic intensities (where N* is infinite), and increase (once N* becomes finite) with traffic intensity.
- At low traffic intensities, earliness costs increase with traffic intensity. For the medium and high cost parameter combinations, earliness costs level off and

slightly decrease as traffic intensity exceeds 75%. For the low cost parameter combination, the earliness cost continues to increase as traffic intensity increases to 100% (although it appears that it will level off somewhere above 100%).

- The behaviour of the tardiness cost vs. ρ curves is very interesting. For all cost parameter combinations, tardiness costs increase with traffic intensity for a while, and then dip. For the low and medium cost parameter combinations, the tardiness costs then rise again. This is caused by the discrete changes of N^* (because it is an integer) and is similar to the sawtooth pattern seen under M/M/1/N control (because of the lower resolution of the graph we see a u-shaped dip instead of a sharp plunge and slow rise).

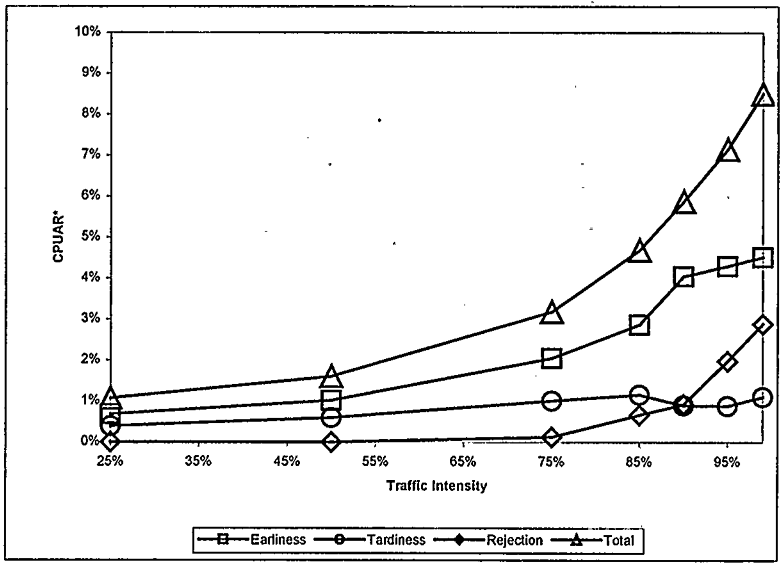


Figure 5.11: CPUAR* vs. ρ for the Low Cost Parameter Combination

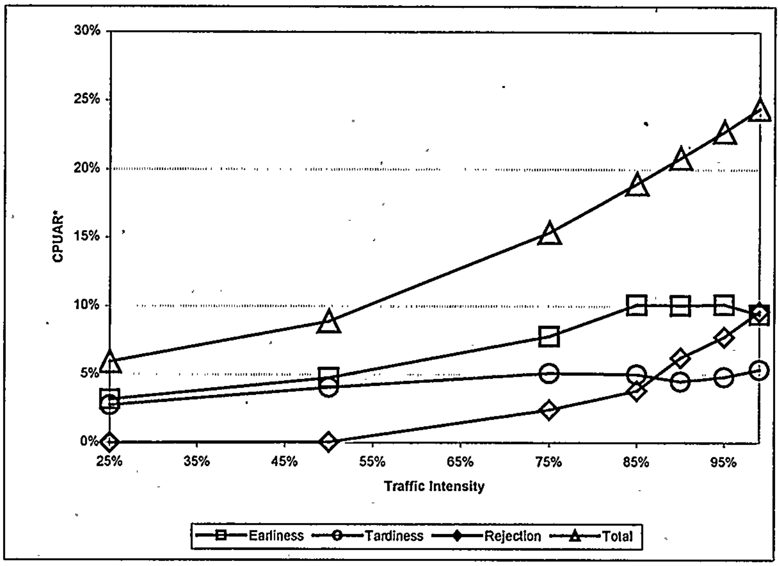


Figure 5.12: CPUAR* vs. ρ for the Medium Cost Parameter Combination

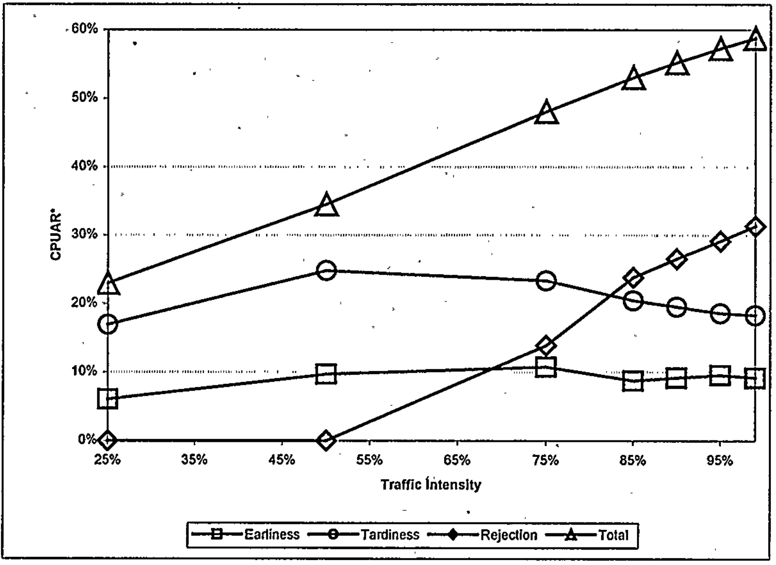


Figure 5.13: CPUAR* vs. ρ for the High Cost Parameter Combination

5.1.5 Absolute Profit Rate

Figure 5.14 shows the absolute profit rate vs. traffic intensity under optimal M/M/1/$N-d$ control for the low, medium and high cost parameter combinations. We see that in all cases the absolute profit rate is non-decreasing with increasing traffic intensity.

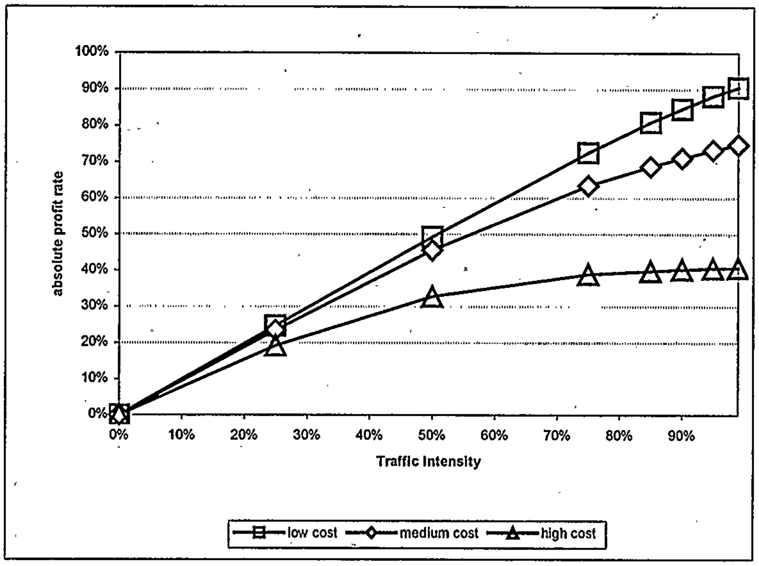


Figure 5.14: Absolute Profit Rate vs. ρ Under M/M/1/$N-d$ Control for Low, Medium and High Cost Parameter Combinations

5.1.6 Sensitivity of PPUAR Performance to Non-Optimal Control Parameters

Figure 5.15 shows the PPUAR values for a small range of near-optimal control parameters ($N^*=16$, $d^*=7.7$) at a traffic intensity of 90% for the medium cost parameter combination. The main results of interest from this graph can be summarized as follows:

- Over-estimating or underestimating *N** by even as little as one can result in significant degradation in performance when compared to the optimal *N**.
- Under this particular condition, it is better to overestimate *N** than to underestimate it. This likely does not hold true under all conditions.
- PPUAR is reasonably insensitive to non-optimal *d**, although it should be noted that the increment of change in *d* (one-fifth) is significantly smaller than the increment of change of *N* (one).

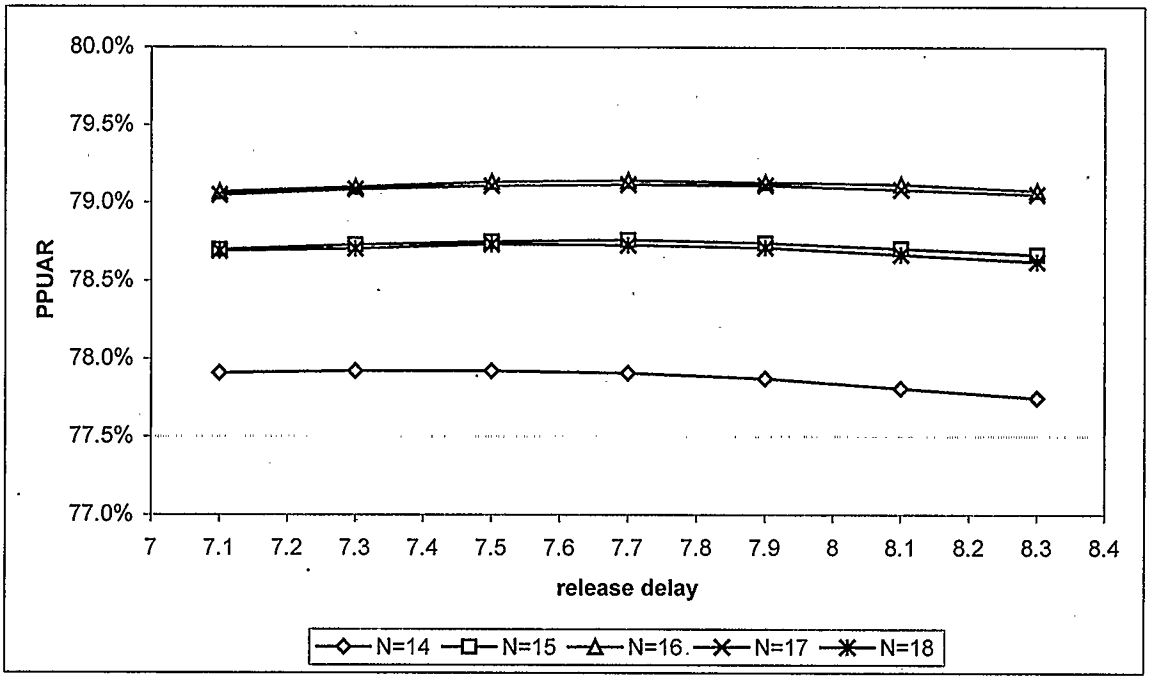


Figure 5.15: Sensitivity of PPUAR to Non-Optimal (*N* ,*d*) When ρ=90% for the Medium Cost Parameter Combination

5.1.7 Summary of Results

We can summarize the results of the analysis of the M/M/1/$N - d$ control policy as follows:

- At very low traffic intensities N^* is effectively infinite, and d^* is identical to that under the M/M/1 – d control policy (with PPUAR performance likewise identical).
- At very high traffic intensities (possible well above 100% depending on the cost parameter combination) d^* converges on some finite value, and N^* is identical to that under the M/M/1/N control policy (with PPUAR performance likewise identical).
- There is an intermediate range of traffic intensities where d^* is non-zero and N^* is finite. Under these conditions N* will be lower than that under M/M/1/N control, and d^* will be higher than that under M/M/1 – d control, with PPUAR performance exceeding that of both the simpler policies.
- The control policy is capable of operating profitably (with a positive PPUAR) at even the high cost parameter combination (without shutting down the system by setting N^* to zero).
- Performance is relatively insensitive to a slightly non-optimal choice of d, but is sensitive to even a slightly non-optimal N.

5.2 M/M/1/N – d – RL Description and Results

5.2.1 Control System Description

Here we model a control policy similar to the M/M/1/$N-d$ policy described above, but with a single enhancement. In an attempt to improve the performance of the order release mechanism we allow the item at the head of the pre-release pool to be released early under two conditions: (i) the system is idle (the server is not busy and there are no jobs in the queue); (ii) the number of jobs in the pre-release pool is greater than or equal to a specified release limit (*RL*). Note that these conditions need only be checked at job acceptance and at job departure from the server. This revised order release policy, which involves three control parameters (*N*, *d* and *RL*), is capable of dealing with congestion by eliminating inserted idle time under busy conditions. This policy is not explored analytically, but is implemented using the simulation model described in Appendix B.

In the following sections we explore the performance of the M/M/1/$N-d-RL$ control policy under our cost model when optimal control parameters are used. We wish to determine if the addition of the early release mechanism results in improvement in performance over the M/M/1/$N-d$ policy. We also wish to examine the sensitivity of PPUAR performance to deviation in the control parameters from their optimal values.

5.2.2 *Optimal Control Parameters (N^*, d^*, RL^*)*

5.2.2.1 Finding the Optimal Control Parameters

We again use *OptQuest* to find the optimal control parameters for a range of traffic intensities under the low, medium, and high cost parameter combinations discussed previously. The optimal control parameter set at each traffic intensity for each cost parameter combination was again found in two stages: (i) a preliminary broad search of low precision; (ii) a second more focused search of higher precision.

For the preliminary broad search, control parameters were permitted to vary over their entire valid range (0 to m for d, 1 to 99 for N, 1 to N+1 for RL[15]). Again, although the release delay d is a continuous-valued parameter, a step size of 0.1 was used. A single short simulation run length of 200 days was used in addition to a 10-day warm up period. *OptQuest* was allowed 300 simulation runs to find the optimal set of control parameter values at each explored traffic intensity, for each exogenous parameter combination. The results of the preliminary broad search were used to guide the second, more focused, search.

For the second, more focused, search a longer single-replication of 1000 days plus a warm-up period of 50 days was used. The control parameters were allowed to vary +/- two steps from the optimal values generated in the preliminary search (for N and RL each

[15] When $RL^*=N+1$, early release is not possible, and the M/M/1/$N-d-RL$ policy is identical to the M/M/1/$N-d$ policy with the same N and d values.

step is 1, for d each step is 0.1). *OptQuest* was allowed to search this tighter parameter space exhaustively. If any of the optimal control parameters was found to lie on the edge of the searched spaced (i.e. the highest or lowest permitted value), the space was expanded in the appropriate direction, and again exhaustively searched. The optimal control parameter set resulting from the second search was then used to generate the optimal performance results presented below in Section 5.2.3.

5.2.2.2 Optimal Control Parameters (N^*, d^*, RL^*) vs. ρ

Figure 5.16 shows N^*, d^*, and RL^* vs. ρ for the low cost parameter combination under the M/M/1/$N-d-RL$ control policy. Figure 5.17 shows a similar plot for the medium cost parameter combination. Figure 5.18 and Figure 5.19 compare N^* and d^* under M/M/1/$N-d$ and M/M/1/$N-d-RL$ control for the low and medium cost parameter combinations respectively. No plots are presented for the high cost parameter combination since results were found to be identical to the M/M/1/$N-d$ policy, with N^* and d^* the same, and RL^* equal to N^*+1. The main results of interest from these figures can be summarized as follows:

- N^* does not change when an early release limit is added to the M/M/1/$N-d$ policy, except at ρ=25% and ρ=50% for the medium cost parameter combination, where, as already noted performance at N^* is effectively no different than that for $N=\infty$.

- The addition of the early release limit does significantly increase the optimal release delay for the majority of traffic intensities. This longer release delay is possible because the system has a means of "expediting" under high-load conditions. Under very low traffic intensities (25%) the early release limit does not increase *d**, and for the low cost parameter combination, the release does not significantly increase *d** at very high (99%) traffic intensities either.
- The release limit itself is very flat at traffic intensities below 75%, and decreases as traffic intensity increases beyond that point.
- The M/M/1/$N-d-RL$ policy identical to the M/M/1/$N-d$ policy for the high cost parameter scenario.

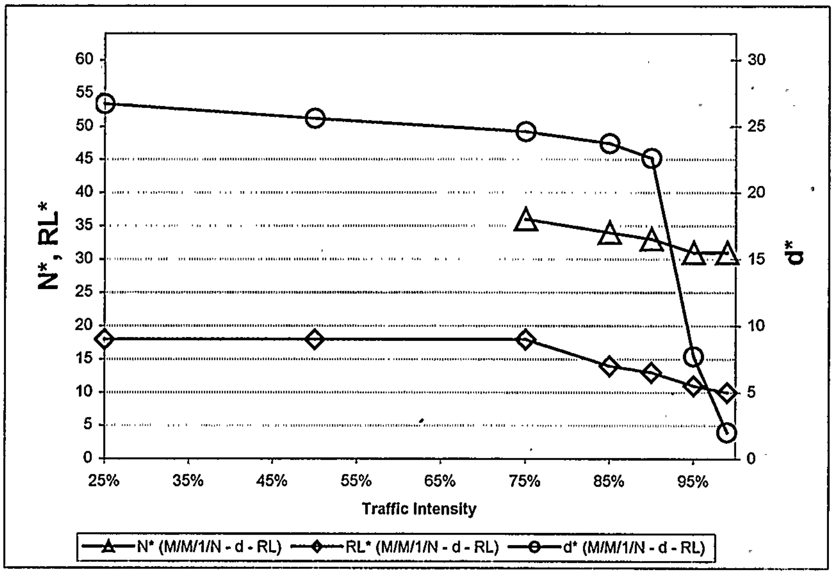


Figure 5.16: (*N, *d**, *RL**) vs. ρ for the Low Cost Parameter Combination**

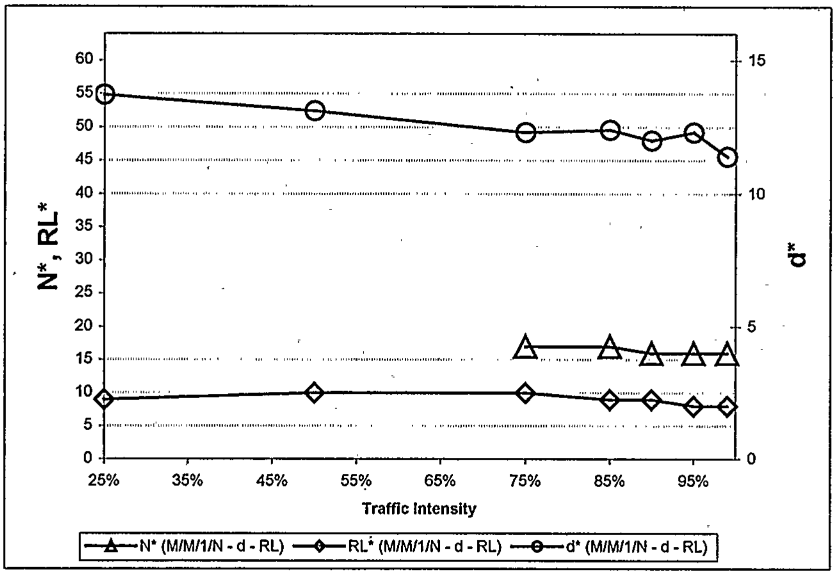


Figure 5.17: (*N, *d**, *RL**) vs. ρ for the Medium Cost Parameter Combination**

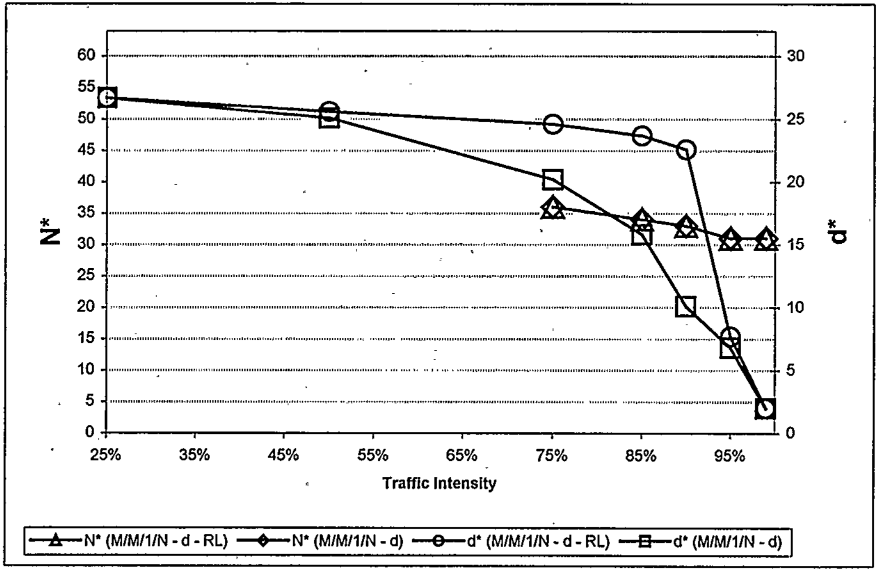


Figure 5.18: (*N, *d**) for M/M/1/*N* – *d* and for M/M/1/*N* – *d* – *RL* vs. ρ for the Low Cost Parameter Combination**

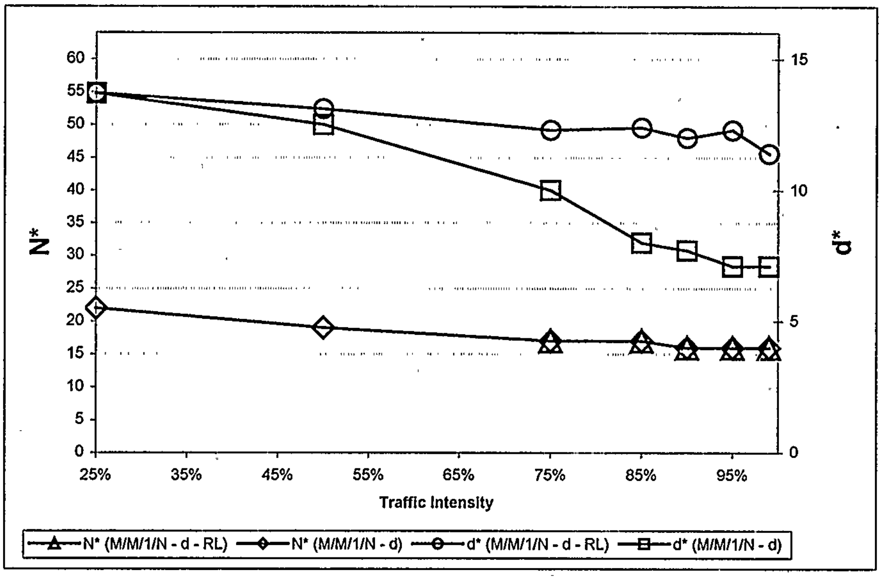


Figure 5.19: (N*, d*) for M/M/1/$N-d$ and for M/M/1/$N-d-RL$ vs. ρ for the Medium Cost Parameter Combination

5.2.3 PPUAR vs. ρ*

Arena's *Process Analyzer* was again used to run 50-replication experiments at the optimal control parameters for each investigated combination of exogenous parameters (with each replication having a run length of 1000 days plus a 50-daywarm up period). These multiple-replication experiments also yielded very high confidence in the final PPUAR* values, such that error bars indicating the 95% confidence interval would not be visible on any of the following graphs.

Figure 5.20 shows PPUAR* vs. ρ for the M/M/1/$N-d-RL$ and M/M/1/$N-d$ policies under the low cost parameter combination. Figure 5.21 shows a similar plot for the medium cost parameter combination. Recall that for the high cost parameter combination, the release limit is not of benefit, and the M/M/1/$N-d-RL$ policy is not capable of besting the M/M/1/$N-d$ policy. The main results of interest from these figures can be summarized as follows:

- For the low cost parameter combination, the addition of the early release limit improves PPUAR* at intermediate traffic intensities. The maximum improvement is 0.48% at a traffic intensity of 75%. At low and high traffic intensity there is no difference in performance over the M/M/1/$N-d$ policy.

- For the medium cost parameter combination, the addition of the early release limit improves PPUAR* at all but the lowest traffic intensity values tested. The maximum improvement is 1.73% at a traffic intensity of 75%, but the improvement remains above 1% for all traffic intensities above this.

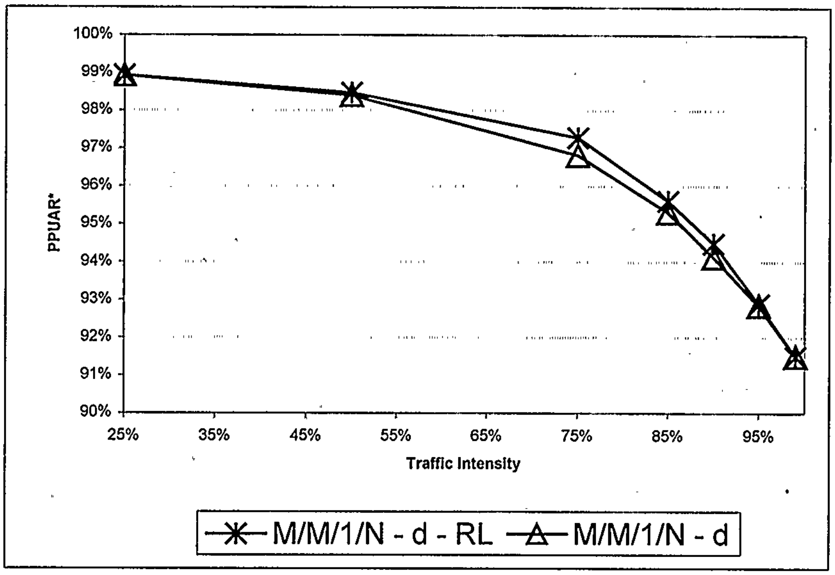


Figure 5.20: PPUAR* for M/M/1/*N* – *d* – *RL* and M/M/1/*N* - *d* for the Low Cost Parameter Combination

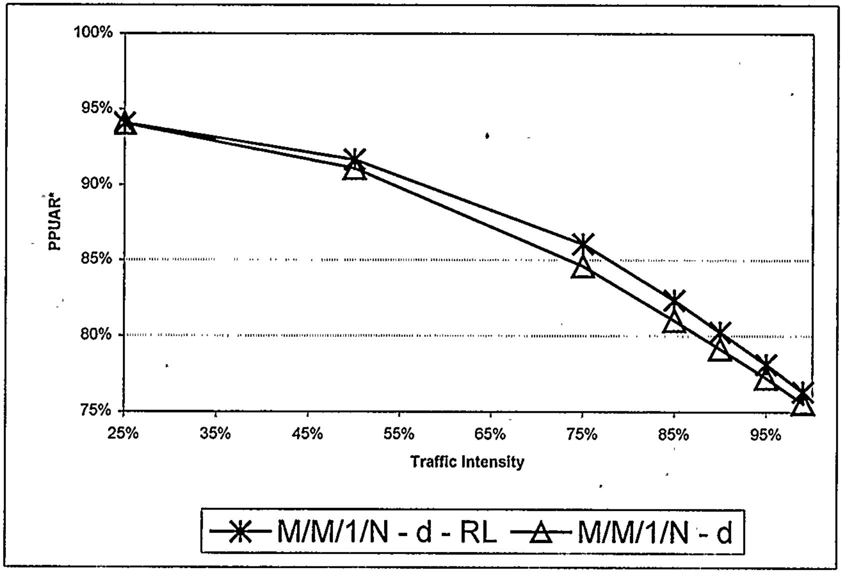


Figure 5.21: PPUAR* for M/M/1/*N* – *d* – *RL* and M/M/1/*N* - *d* for the Medium Cost Parameter Combination

5.2.4 Cost Breakdown

Figure 5.22 shows the optimal cost per unit arriving revenue for the low cost parameter combination under M/M/1/$N-d-RL$ control, in total and broken up into its three components. Figure 5.23 show a similar plot for the medium cost parameter combination. The main results of interest from these figures can be summarized as follows:

- We see when comparing Figure 5.23 to Figure 5.12, that under the medium cost parameter combination, the addition of the release limit improves performance by reducing the earliness costs. Note that tardiness costs actually increase slightly..
- However, when comparing Figure 5.22 to Figure 5.11 (low cost parameter combination), we see that the performance improvements resulting form the addition of the early release limit are sometimes due to lowering earliness costs (with a slight increase in tardiness costs), and sometimes due to lowering tardiness costs (with a slight increase in earliness costs). This is likely a result of the discrete nature of N and RL.

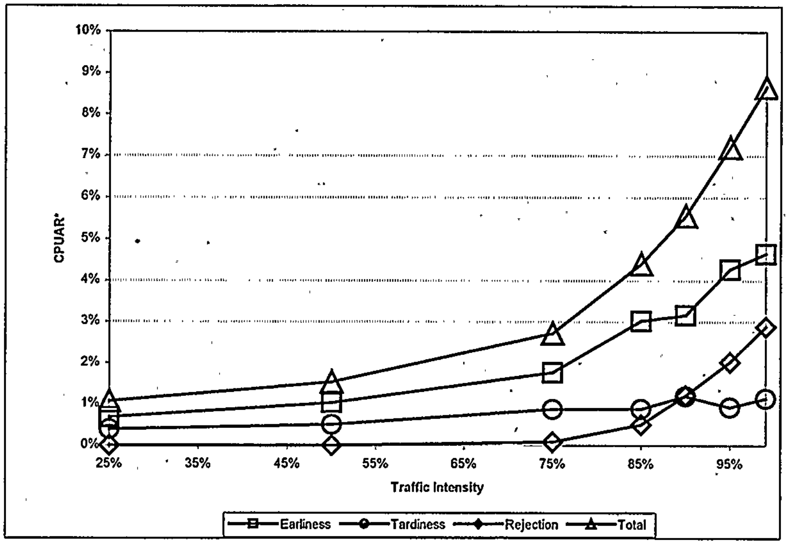


Figure 5.22: CPUAR* vs. ρ for the Low Cost Parameter Combination

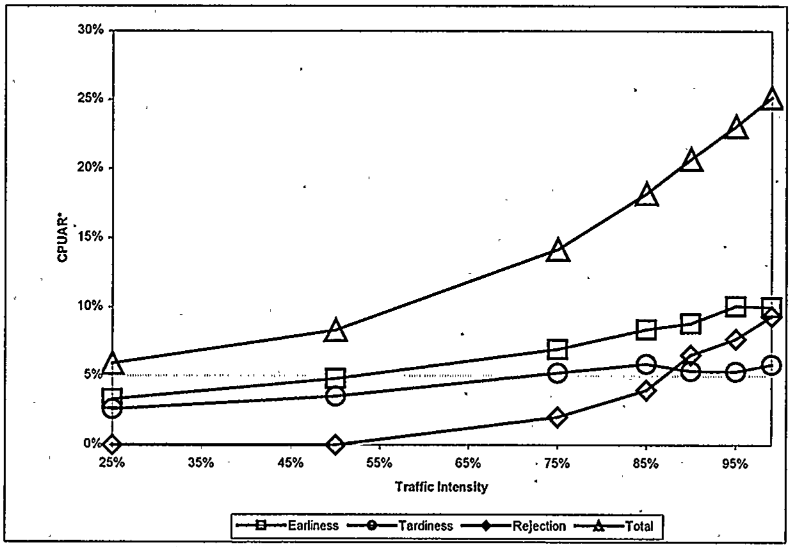


Figure 5.23: CPUAR* vs. ρ for the Medium Cost Parameter Combination

5.2.5 Absolute Profit Rate

Figure 5.24 shows the absolute profit rate vs. traffic intensity under optimal M/M/1/$N-d-RL$ control for the low and medium cost parameter combinations. We see that in both cases the absolute profit rate is non-decreasing with increasing traffic intensity (as expected since PPUAR* for the M/M/1/$N-d-RL$ policy is at least as great as for the M/M/1/$N-d$ policy).

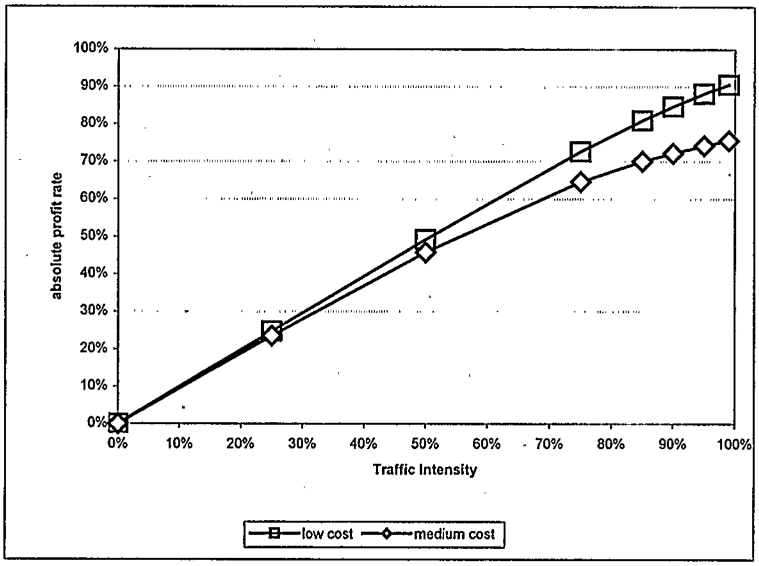


Figure 5.24: Absolute Profit Rate vs. ρ Under M/M/1/$N-d-RL$ Control for Low and Medium Cost Parameter Combinations

5.2.6 Sensitivity of PPUAR Performance to Non-Optimal RL

Figure 5.25 shows a plot of PPUAR vs. RL at ρ=90% for the medium cost parameter combination when N* and d* are optimal (N*=16, d*=12, RL*=9). The main results of interest from this figure can be summarized as follows:

- Overestimating or underestimating *RL** by as little a one or two can result in significant degradation in performance when compared to that with the optimal early release limit. Note that optimal PPUAR under M/M/1/$N-d$ control is 79.15%, and misestimating *RL** by plus or minus two still results in improved performance over that of the M/M/1/$N-d$ policy.

- Under this particular set of exogenous parameters, it is slightly better to underestimate RL* by a reasonable number (less than six) than to overestimate by the same amount, but this does likely not hold true under all conditions.

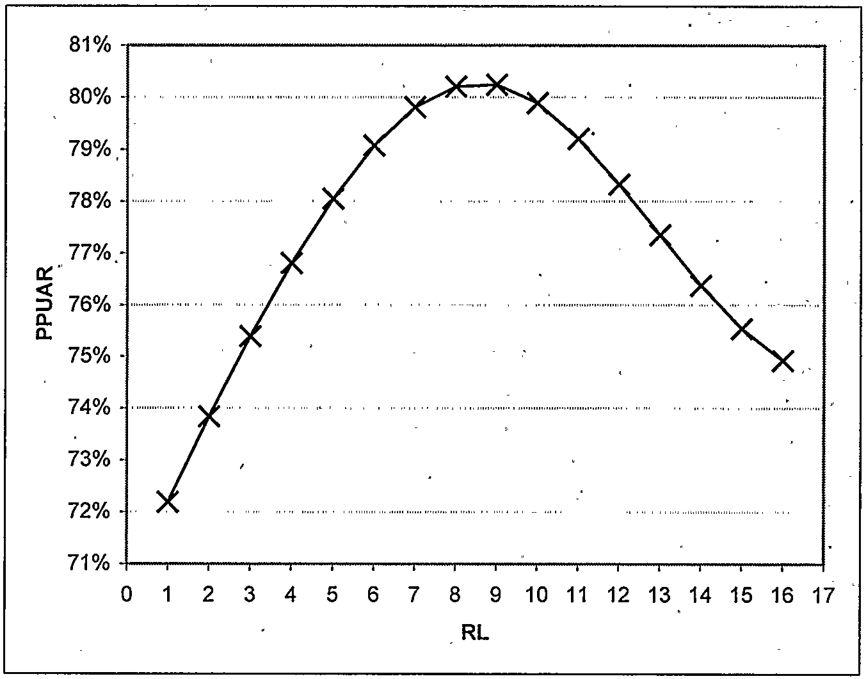


Figure 5.25: PPUAR vs. RL at ρ=90% for the Medium Cost Parameter Combination (*N*=16, *d*=12)

5.2.7 Summary of Results

We can summarize the results of the analysis of the M/M/1/$N-d-RL$ control policy as follows:

- The M/M/1/$N-d-RL$ control policy is not capable of improving performance over that of the M/M/1/$N-d$ policy under the high cost parameter combination.
- The M/M/1/$N-d-RL$ control policy is capable of improving performance over that of the M/M/1/$N-d$ policy under low and medium cost parameter combinations, with improvements being greater under the medium cost parameter scenario. For both cost parameter combinations, improvement is greatest at a traffic intensity of 75%.
- The addition of the release limit does not change the value of N^* (when compared to the M/M/1/$N-d$ control policy).
- Over the majority of traffic intensities, the addition of the early release limit does increase d^* (when compared to that for the M/M/1/$N-d$ control policy) .

Chapter 6

Conclusions and Future Research

This chapter summarizes the main results of the analytical and experimental work reported in this thesis, as well as highlighting the primary original contributions of the research. It concludes with a list of potential directions for future research that extend the work reported in this thesis.

6.1 Summarizing the Main Results From the Research

This section summarizes the main results obtained from this research, organized by control policy.

6.1.1 Uncontrolled Behaviour of the M/M/1 System

The analytical and numerical results for the uncontrolled system, found in Section 3.4, have provided insight into the clear need for control of earliness and tardiness costs under certain conditions. We have seen that without input control, earliness costs can be very significant at low traffic intensities, while tardiness costs escalate out of control as the traffic intensity approaches 100% for any value of cost parameters k_m and k_r, and any flow allowance.

6.1.2 The M/M/1 – d Control Policy

The analytical and numerical results for the M/M/1 – *d* control policy, found in Section 4.2, have provided insight into the use of an order release mechanism in isolation. Under conditions where earliness costs dominate for the uncontrolled system, the use of a fixed release delay can improve performance by significantly lowering expected earliness in exchange for a smaller increase in expected tardiness. However, earliness costs are only significant at low traffic intensities, and thus the M/M/1 – *d* policy is not effective at higher traffic intensities. Interestingly, the optimal release delay is independent of the cost magnitude factor, and decreases with increasing traffic intensity while increasing with increasing relative earliness costs.

6.1.3 The M/M/1/N Control Policy

The analytical and numerical results for the M/M/1/*N* control policy, found in Section 4.4, have provided insight into the accept/reject decision in isolation as part of the order review mechanism. Under conditions where tardiness costs dominate for the uncontrolled system, the use of a system work in process limit can improve performance by significantly lowering expected tardiness in exchange for a smaller increase in rejection costs. However, tardiness costs only dominate at high traffic intensities, and thus the M/M/1/*N* policy is not effective at lower traffic intensities. The ability to reject jobs allows traffic intensities to increase beyond 100% and also allows the system to be "shut down" under conditions where profitability is not possible.

6.1.4 The M/M/1/$N - d$ Control Policy

The experimental results for the M/M/1/$N - d$ control policy, found in Section 5.1, have provided insight into the combined use of judicious rejection and of delayed order release. The combined order release and rejection mechanisms allow for better performance than that of the best of the individual mechanisms in isolation, particularly in the intermediate range of traffic intensities in which most real manufacturing systems might operate. The combination of a release delay and judicious rejection allows the system to operate profitably at any traffic intensity, even under the most aggressive cost parameter combination. Implementing a release delay with a rejection mechanism results in a longer optimal delay than that when no rejection is permitted, and a smaller optimal system work in process limit than when no release delay is permitted.

6.1.5 The M/M/1/$N - d - RL$ Control Policy

The experimental results for the M/M/1/$N - d - RL$ control policy, found in Section 5.2, have provided insight into how an enhanced order release mechanism can result in further improvements in performance. The specific conditional early release mechanism embedded in this control policy is able to avoid the excessive tardiness costs that would otherwise be associated with an increase in the release delay, by allowing jobs to be released from the pre-release pool early under exceptionally high-load conditions.

6.2 Applicability of This Research in Practice

The major consideration in determining the extent to which the results of this research are applicable to manufacturing systems in general is the extremely simple nature of the M/M/1 system analyzed. The typical make-to-order manufacturing system to which an ORR mechanism is being applied is likely be significantly more complex in many ways, including:

- Multiple servers, possible of differing type (with a wide variety of possible shop floor configurations).
- Multiple categories of product (with differing service rates, arrival rates, flow allowances and routings).
- Occurrences of uncertain events (such as breakdowns).

The simple system modelled is not directly representative of any real manufacturing system, but input control concepts explored are still valuable scaled up and adapted for more complex systems.

While the precise results cannot be generalized to more complex real systems, it should be universally true that for any manufacturing system where both earliness and tardiness costs exist, intelligent order release can be used to reduce earliness costs and judicious rejection used to reduce tardiness costs. These generic insights can help guide the development of input control mechanisms for complex real-world manufacturing systems.

6.3 *Original Contributions*

The present research includes several original contributions in the area of input control:

1. The first contribution of this research is the evaluation of ORR concepts under a cost model capturing both earliness and tardiness costs. The existing research is primarily focused only on minimizing the expected tardiness and/or the number of tardy jobs, with earliness costs ignored. The present research uses a parametric cost model that includes both earliness and tardiness costs, and permits a wide range of cost schemes to be investigated. The ability of the order release mechanism to control earliness costs has largely been ignored by the existing research, and helps to counter the criticism that ORR is unable to reduce tardiness.

2. Another contribution of this research is the development of analytical results for the M/M/1 – *d* and the M/M/1/*N* control policies. Analytical representations of the operational, cost and profit measures allow for the easy exploration of policy performance under a wide range of exogenous parameters. Additionally, the analytical representations help provide general insight into the effects of the exogenous and control parameters on system performance.

3. This research has shown that order review and judicious acceptance/rejection, when used in combination, can be effective in controlling both earliness and tardiness costs. In real-life manufacturing systems, the complexity of the control policies will be greater (likely in proportion to the complexity of the

manufacturing system), but the function will remain primarily the same: (i) keeping jobs off the manufacturing floor as long as possible to prevent congestion and early completion; (ii) not accepting additional jobs when doing so will be detrimental to the on-time completion of jobs already accepted.

6.4 Future Research

There is a great deal of opportunity for future research related to the topics explored in this thesis. Some potential directions for further research are summarized in the following subsections.

6.5.1 Further Enhanced Control Algorithms for the Existing System

All control systems analyzed in this thesis were relatively simple in that they did not use information available about the jobs accepted by the system. It is expected that there are well-performing control algorithms that make use of the due dates of jobs waiting for release. We have made some initial investigations into some more complex algorithms (integrating discrete-event simulation with non-linear optimization) that make use of information about the jobs in the system, and have found that they may result in performance improvements over the M/M/1/$N-d-RL$ policy under certain conditions.

6.5.2 Increasing the Complexity of the System

The M/M/1 system was chosen for analysis because of its easily-analyzed nature and because of the insight it offers into more complex systems. There is significant insight to be gained by applying the control systems explored to more complex systems. One such system is a 2-stage manufacturing system where there is the opportunity for an additional delay between the two stages. The ability to delay between stages allows for a greater degree of control over earliness, but for multi-stage systems, the cost of work-in-process must also be considered. Most workload control literature focuses on job shops with 5-10 servers. Full integration of this work with the existing body of workload control research requires applying the cost model and control systems analyzed to the systems studied by other researchers.

6.5.2.1 Non-Exponential Interarrival and Service Times

Exponential interarrival and service times were used because they facilitate analytical solutions to the uncontrolled and simple control scenarios. However, exponential times are often inappropriate, particularly for service time distributions, which tend to have far less variation. Alternatively, disruptions to the service or arrival processes (such as machine breakdowns or occasional batch arrivals) may be added as additional sources of variability. Combining a delayed release strategy with exception or extreme-condition based early release rules may prove extremely valuable under these conditions. An alternative approach is to investigate the situation where the actual service time is random, but known as soon as the job arrives at the system.

6.5.2.2 Modifications to the Cost/Profit Model

There are a number of modifications that could be made to the cost/profit model that may be worthy of future investigation. One such change is the addition of an additional rejection cost, where rejected jobs would see a cost contribution beyond lost revenue. Another possible change is the capping of tardiness costs at either the maximum job revenue, or some other value. Capping tardiness costs may more realistically represent contract-specified tardiness penalties. Other potential modifications to the cost model include quadratic earliness and tardiness costs, and the use of delivery windows (extended periods of time over which neither earliness nor tardiness costs accrue).

6.5.2.3 Multiple Job Classes

While the present research is only concerned with a single job class, we have also done some investigation into multiple-class systems, where some jobs may have tighter flow allowances but are charged a price premium. The control systems investigated could be altered to accommodate multiple classes, and further work in this area is warranted. The use of multiple flow-allowance based job classes could be used to balance the tradeoffs between the competitive advantages of lowering quoted lead times, and the flexibility of maintaining a release delay.

References

Azizolglu, M., and Webster, S., 1997, "Scheduling about an unrestricted common due window with arbitrary earliness/tardiness penalty rates", IIE Transactions, Vol. 29, pp. 1001-1006.

Bechte, W., 1988, "Theory and practice of load-oriented manufacturing control", International Journal of Production Research, Vol. 26, No. 3, pp. 375-395.

Bergamaschi, D., Cisgolini, R., Perona, M., and Portioli, A., 1997, "Order review and release strategies in a job shop environment: a review and a classification", International Journal of Production Research, Vol. 35, No. 2, pp. 399-420.

Baker, K.R., 1984, "Sequencing rules and due-date assignments in a job shop", Management Science, Vol. 30, pp. 1093-1104.

Bertrand, J.M.W., 1983a, "The use of workload information to control job lateness in controlled and uncontrolled release production systems", Journal of Management, Vol. 3, No. 2, pp. 79-93.

Bertrand, J.W.M., 1983b, "The effect of workload dependent due-dates on shop performance", Management Science, Vol. 29, No. 7, pp.799-816.

Çigolini, R., and Portioli-Staudacher, A., 2002, "An experimental investigation on workload limiting methods within ORR policies in a job shop environment", Production Planning and Control, Vol. 13, No. 7, pp. 602-613.

Crabill, T.B., Gross, D., and Magazine, M.J., 1977, "A classified bibliography of research on optimal design and control of queues", Operations Research, Vol. 25, No. 2, pp. 219-232.

Dessouky, M., Kijowski, B., Verma, S., 1999, "Simultaneous batching and scheduling for chemical processing with earliness and tardiness penalties", Production and Operations Management, Vol. 8, No. 4, pp. 433-444.

Elhafsi, M., 2002, "Optimal leadtime planning in serial production systems with earliness and tardiness costs", IIE Transactions, Vol. 34, pp. 233-243

Enns, S.T., 1995, "An economic approach to job shop performance analysis", International Journal of Production Economics", Vol. 38, pp. 117-131.

Fredendall, L.D., and Melnyk, S.A., 1995, "Assessing the impact of reducing demand variance through improved planning on the performance of a dual resource constrained job shop", International Journal of Production Research, Vol. 33, No. 5, pp.1521-1543.

Gaalman, G., and Perona, M., 2002, "(Editorial) Workload control in job shops: an introduction to the special issue", Production Planning and Control, Vol. 13, No. 7, pp. 565-567.

Jensen, J.B., Philipoom, P.R., Malhotra, M.K., 1995, "Evaluation of scheduling rules with commensurate customer priorities in job shops," Journal of Operations Management, Vol. 13, pp. 213-228.

Kelton, W.D., Sadowski, R.P., and Sadowski, D.A., 2002, "Simulation with Arena", 2nd Edition, New York: McGraw-Hill.

Klienrock, L, 1975, "Queueing Systems, Volume I: Theory", John Wiley and Sons.

Lippman, S.A., and Ross, S., 1971, "The streetwalker's dilemma: a job shop model", SIAM Journal of Applied Mathematics, Vol. 20, pp. 336-344.

Melnyk, S.A., and Carter, P.L., 1987, "Production activity control: a practical guide", Homewood, IL: Dow Jones-Irwin.

Melnyk, S.A., Denzler, D.R., and Fredendall, L.D., 1992, "Variance control vs. dispatching efficiency", Production and Inventory Management Journal, Third Quarter, 1992.

Melnyk, S.A., and Ragatz, G.L., 1988, "Order review/release and its impact on the shop floor", Production and Inventory Management Journal, Second Quarter, pp. 13-17.

Melnyk, S.A., and Ragatz, G.L., 1989, “Order review: research issues and perspectives”, International Journal of Production Research, Vol. 27, No. 7, pp. 1081-1096.

Miller, B.I., 1969, “A queuing reward system with several customer classes”, Management Science, Vol. 16, pp. 234-245.

Nandi, A., 2000, “Input control strategies for make-to-order manufacturing systems via order acceptance/rejection”, Ph.D. thesis, Department of Mechanical and Manufacturing Engineering, The University of Calgary, Calgary, Alberta, Canada.

Papadopoulos, H.T., Heavey, C., and Browne, J., 1993, “Queueing Theory in Manufacturing Systems Analysis and Design”, Boundary Row, London, Chapman and Hall.

Philipoom, P.R., and Fry, T.D., 1992, “Capacity-based order review/release strategies to improve manufacturing performance”, International Journal of Production Research, Vol. 30, No. 11, pp. 2559-2577.

Rockwell Software, 2000, “OptQuest for Arena User Guide”, Rockwell Software Inc.

Rogers, P., 2002, “Optimum-seeking simulation in the design and control of manufacturing systems: experience with OptQuest for Arena”, Proceedings of the 2002 Winter Simulation Conference, edited Yücesan, E., Chen, C.-H., Snowdon, J.L. and Charnes, J.M., San Diego.

Rogers, P., and Segal, Y., 2003, "Input control via rejecting orders in a make-to-order manufacturing system", Proceedings of the Industrial Engineering Research Conference, Portland.

Scott, M., 1969, "A queuing process with some discrimination", Management Science, Vol. 16, pp. 227-233.

Scott, M., 1970, "Queuing with control on the arrival of certain types of customers", CORS Journal, Vol. 8, pp. 75-86.

ten Kate, H.A., 1994, "Towards a better understanding of order acceptance", International Journal of Production Economics, Vol. 37, pp. 139-152.

Wester, F.A.W., Wijngaard, J., and Zijm, W.H.M., 1992, "Order acceptance strategies in a production-to-order environment with setup times and due dates", International Journal of Production Research, Vol. 30, No. 6, pp. 1313-1326.

Wight, O., 1970, "Input/output control a real handle on lead time", Production and Inventory Management, Third Quarter, pp. 9-31.

Wisner, J.D., 1995, "A review of the order release policy research", International Journal of Operations and Production Management, Vol. 15, pp. 25-40.

Appendix A

Glossary of Acronyms, Symbols, and Special Terms

This appendix briefly defines the acronyms, symbols, and special terms used in this thesis.

* A '*' following a control parameter indicates that the parameter is at the value that optimizes PPUAR given the exogenous parameters. A '*' following an output measure indicates that the measure is that for optimal values of all control parameters.

AvgEarliness The expected earliness of a job accepted into the system.

AvgTardiness The expected tardiness of a job accepted into the system.

C_E The slope of the earliness cost function, with dimensions currency per unit time.

C_T The slope of the tardiness cost function, with dimensions currency per unit time.

CPUAR The cost per unit arriving revenue, equal to the difference between 100% and the PPUAR. This is further decomposed into rejection, earliness and tardiness components.

D, d The delay between when a job is accepted by the system, and when it is released to the shop floor, with D expressed in natural time units, and d expressed as a multiple of the mean service time.

$\Gamma()$ The gamma function. $\Gamma(a)$ is the complete gamma function, and is equal to $(a\text{-}1)!$ when a is an integer. $\Gamma(a, x)$ is the upper incomplete gamma function.

I_{E}, i_{E} The critical earliness interval, with I_E expressed in natural time units, and i_E expressed as a multiple of the mean service time.

I_T, i_T The critical tardiness interval, with I_T expressed in natural time units, and i_T expressed as a multiple of the mean service time.

k_m The cost magnitude factor.

k_r The relative earliness cost factor.

λ The arrival rate. The mean interarrival time is $1/\lambda$.

μ The service rate. The mean service time is $1/\mu$.

M, m The flow allowance, with M expressed in natural time units, and m expressed as a multiple of the mean service

	time.
M/M/1	Denotes the uncontrolled system.
M/M/1 $-d$	Denotes the control policy making use of a fixed release delay.
M/M/1/N	Denotes the control policy making use of a system work in process limit.
M/M/1/$N-d$	Denotes the control policy making use of both a fixed release delay and a system work in process limit.
M/M/1/$N-d-RL$	Denotes the control policy making use of a system work in process limit, and a release delay with the capability for early release when idle time occurs and the number of jobs in the system exceeds a release limit.
MLT	Manufacturing lead time.
MTO	Make-to-order.
MTS	Make-to-stock.
N	The system work in process limit.
ORR	Order review and release.

π_n	The probability of there being *n* jobs in the system.
PAC	Production activity control.
PAN	Process Analyzer, *Arena*'s batch simulation utility.
pdf	Probability distribution function.
PPAJ	Profit per arriving job (in units of currency).
PPUAR	Profit per unit arriving job revenue, as a percentage.
PR	Profit rate, with dimensions currency per unit time.
Pr(acc)	The probability of an arriving job being accepted by the system.
Pr(rej)	The probability of an arriving job being rejected by the system.
PropTard	The proportion of accepted jobs that will be tardy.
ρ	The traffic intensity, equal to λ/μ.
RL	The early release limit.
sp	Indicates the special case, $\rho = 1$.

TWK Total work content, the sum of a job's expected service time over all operations on its routing.

WIP Work in process, the total numbers of uncompleted jobs in the manufacturing system.

MLT Manufacturing lead time, the time between when an order is released it the floor, and when it is due to be completed.

WLC Workload control.

Appendix B

Description of The Simulation Model

This appendix provides a detailed description of the simulation model used to generate the experimental results reported in Chapter 5. The simulation model code can be found in Appendix C.

B.1 Model Overview

The simulation model was created in *Arena*'s high-level graphical modelling interface, and can be found in the file "*MM1 ORR Testbed.doe*". This file presents the model logic in flowchart form, as well as a graphical representation of the system for debugging purposes. The "*MM1 ORR Testbed.doe*" file can be used to generate *.exp* and *.mod* files containing only the underlying logic code included in Appendix C. For more information on the *Arena* simulation package see Kelton *et al.* (2002).

B.1.1 Model Features

The model was constructed for robustness, flexibility, and ease of use. Whenever possible, hard coding of values and formulae was avoided in favour of variables and user expressions which can be easily and centrally viewed and modified. The model makes use of *common random numbers* as a variance reduction technique. Interarrival and processing times are sampled from dedicated streams ensuring that arrival and processing patterns are synchronized across alternative control parameters (and systems) and environmental conditions. The model also includes many debugging aids including an animation of the system with a "digital dashboard" of key performance measures. Additionally, the model collects many performance measures that are not required in the context of this thesis, but may be useful for debugging, or for extending the model. In general, the extensibility of the model was a major development objective, and the model is capable of being used to pursue many of the future research opportunities identified in Chapter 6.

B.1.2 Model Logic

The following section contains a brief summary of the model logic. Complete model logic code can be found in Appendix C. The organization presented replicates the flow of jobs through the model.

The first segment of logic is devoted to the generation of jobs. Jobs are generated with appropriately sampled interarrival time, and certain statistics are collected. In order to facilitate the synchronization of the *common random number* streams, processing times are also sampled at this point.

The next segment of logic involves the accept/reject decision, which is made based on the state of the system and the value of the appropriate control parameter. Rejected jobs proceed to statistic-collection logic before exiting the system. Accepted jobs enter the pre-release pool logic.

The pre-release pool logic begins with the collection of arrival statistics for arriving jobs, and the recording of several job attributes for later use. Arriving jobs then trigger a check for early release conditions, and are sent to the pre-release pool. Their release from the pre-release pool is then scheduled.

When a scheduled or early release occurs, the released job undergoes some data collection and statistics recording, and proceeds to the first-come-first-serve queue of a single-resource server. The job waits in queue until being processed by the server according to its pre-sampled processing time. After processing, the job triggers another check for early release conditions, and proceeds to statistics collection logic before exiting the system.

The early release logic checks if early release conditions are met and triggers the release of the first job in the pre-release pool when appropriate. If a job has been released early, its scheduled release is ignored.

B.2 Functional Model Description

The following functional model description describes the variables used to configure the model, and the output statistics required to interpret model results.

B.2.1 User-Controllable Variables

The following variables may be altered by the user to control the exogenous parameters, the control policy, and the control parameters under which the system operates. Model variables not included here are used internally by the model.

Exogenous paremeters:

vkm	The cost magnitude parameter, *km*.
vkr	The relative earliness cost parameter, *kr*.
vm	The normalized flow allowance, *m*.
vMIAT	The mean interarrival time ($1/\lambda$).
vMPT	The mean processing time ($1/\mu$).

Control parameters:

vd	The normalized fixed release delay (*d*).
vN	The system work limit (*N*).
vRL	The early release limit (*RL*).

Advanced parameters:

vIncomingCheck	Binary flag that determines if early-release conditions are checked upon job arrival in addition to being checked at every service completion. All experiments were performed with vIncomingCheck set to 1 (true).
vInterarrivalStream	The random number stream used to generate interarrival times.
vServiceStream	The random number stream used to generate processing times.

B.2.2 Output Statistics

The following are the primary output statistics produced by the simulation model. Additional outputs are present primarily for debugging purposes. For multiple-replication experiments, *Arena* will provide mean, min, max and 95% confidence half-width information for the following statistics (except where noted):

AvgEarliness — The average earliness of accepted jobs.

AvgTardiness — The average tardiness of accepted jobs.

Percent Accepted — The percentage of arriving jobs accepted by the system.

%ReleasedEarly — The percentage of accepted jobs released early (when the early release limit control policy is in force).

Calculated PPUAR — The profit per unit arriving revenue given the cost parameters in effect, as a percentage (calculated based on mean earliness, mean tardiness, and percent accepted).

Mean Tallied PPUAR — The profit per unit arriving revenue given the cost parameters in effect, as a fraction (tallied for each arriving job).

Actual IAT — The actual mean interarrival time generated.

PPUAR HW — This statistic is implemented because *Arena*'s scenario analysis tool (*Process Analyzer*) does not provide confidence interval information for output statistics. For multiple-replication runs, the "Average" value of PPUAR

HW as listed in the Output Summary is the 95% half-width of Calculated PPUAR (in percent) at the end of the second-last replication. For individual replications, this statistic should be ignored. When not using PAN, the half-width of "Calculated PPUAR" should be used instead.

B.3 Configuring the Model

This section describes the process of configuring the model to explore a particular set of exogenous parameters under particular control parameter values for a particular control policy.

B.3.1 Exogenous Parameters

To configure the exogenous parameters, vm, vkm and vkr should be set to the values of m, k_m and k_r respectively. In order to configure the traffic intensity, vMPT should be set equal to 1, and vMIAT set to $1/\rho$.

B.3.2 Control System and Control Parameters

By setting certain control parameters to zero or effectively infinity (i.e. 999999), we can enable each of the control policies to be explored:

1. Uncontrolled M/M/1

   Set vd to zero, and set vN and vRL both effectively to infinity (999999).

2. M/M/1 – *d*

   Set vd to the value of *d* being investigated, and set vN and vRL both effectively to infinity (999999).

3. M/M/1/*N*

   Set vN to the value of *N* being investigated, and set vd to zero and vRL effectively to infinity (999999).

4. M/M/1/*N* – *d*

   Set vd and vN to the values of *d* and *N* being investigated, and set vRL effectively to infinity (999999).

5. M/M/1/*N* – *d* – *RL*

   Set vd, vN and vRL to the values of *d*, *N*, and *RL* being investigated.

B.3.3 Run Setup

Configuring the run set up requires specifying the number of simulation replications, the length of each simulation replication, and the warm-up period for each replication (after which statistics are cleared). Additionally the number of hours in each day, and the base time units of the model must be specified. Note that we have operated the model assuming 24-hour days with a base time unit of minutes. Therefore, setting vMPT to 1 (minute) will result in $24 \times 60 = 1440$ expected arrivals per day at a traffic intensity of 100%.

B.4 Validation and Precision Issues

This section will discuss model verification and validation issues, as well as simulation precision. Additionally, the method used to find optimal values of control parameters will be described.

B.4.1 Co-validation of Simulation Model/Analytical Results

The simulation model is capable of analyzing the uncontrolled M/M/1, M/M/1 – d and M/M/1/N systems for which there are analytical results available. The output from the simulation model was compared against expected values derived analytically for many environmental conditions, and results matched as expected. Comparison with analytical results was also used to determine appropriate warm-up times, run lengths and replication numbers for the experiments reported in Chapter 5.

B.4.2 Discussion of Model Precision and Speed Considerations

Because of the high variability of the exponential interarrival and processing times, obtaining precise estimates of output statistics of interest (particularly PPUAR) may require significant run lengths. Under most of the configurations tested, run speed was approximately thirty seconds for 144000 simulation minutes (one hundred 24-hour days) on a Pentium III 800 MHz PC. Note that run speed is primarily dependent on the number of arriving jobs, and the above speed is for a traffic intensity of 90% with a mean processing time of one minute (lower traffic intensity or higher mean processing time increases speed[16]).

Required warm-up length was determined using several pilot simulation runs with graphical monitoring of the state of the pool, queue, and performance measures throughout the run. A warm-up length of 10 days was deemed to be more than adequate (with a time cost of only about 3 second) for shorter investigative runs. For longer simulations where the relative run time increase would be negligible, warm-up length was increased to 50 days.

[16] These speeds are for *Arena* proper, running as the only CPU-intensive application. When using *Process Analyzer*, speed is about half as the program appears to limit itself to less than 50% of CPU usage.

Appendix C

Simulation Code

Simulation Model Code

This appendix includes the model (.mod) and experiment (.exp) files generated by the high-level .doe simulation model file.

C.1 Model (.mod) File

```
;
;
;     Model statements for module:  Create 1
;

48$            CREATE,
1,MinutesToBaseTime(eIAT),Job:MinutesToBaseTime(eIAT):NEXT(49$);

49$            ASSIGN:        Create Arriving Jobs.NumberOut=Create Arriving
Jobs.NumberOut + 1:NEXT(29$);


;
;
;     Model statements for module:  Record 10
;
29$            TALLY:         Actual Interarrival Time Tally,BET,1:NEXT(44$);


;
;
;     Model statements for module:  Assign 7
;
44$            ASSIGN:        aProcTime=ePT:NEXT(30$);


;
;
;     Model statements for module:  Decide 4
;
30$            BRANCH,        1:
                              If,vNumInWhole >= eSystemWorkLimit,52$,Yes:
                              Else,53$,Yes;
52$            ASSIGN:        Reject Incoming Job if N Exceeded.NumberOut True=
                              Reject Incoming Job if N Exceeded.NumberOut True +
1:NEXT(31$);

53$            ASSIGN:        Reject Incoming Job if N Exceeded.NumberOut False=
                              Reject Incoming Job if N Exceeded.NumberOut False +
1:NEXT(34$);


;
;
;     Model statements for module:  Record 11
;
31$            COUNT:         Rejected Job Count,1:NEXT(32$);


;
;
;     Model statements for module:  Record 12
;
32$            TALLY:         Rejected Job Interarrival Tally,BET,1:NEXT(46$);


;
;
;     Model statements for module:  Record 19
;
46$            TALLY:         Exiting Profit Tally,0,1:NEXT(33$);


;
;
;     Model statements for module:  Dispose 4
;
33$            ASSIGN:        Dispose Rejected Jobs.NumberOut=Dispose Rejected
Jobs.NumberOut + 1;
```

```
54$             DISPOSE:        No;


;
;
;     Model statements for module:  Record 13
;
34$             COUNT:          Accepted Job Count,1:NEXT(35$);


;
;
;     Model statements for module:  Record 14
;
35$             TALLY:          Accepted Jobs Interarrival Tally,BET,1:NEXT(0$);


;
;
;     Model statements for module:  Assign 1
;
0$              ASSIGN:         vNumInWhole=vNumInWhole+1:
                                vNumInPRP=vNumInPRP+1:
                                vSigNum=vSigNum+1:NEXT(16$);


;
;
;     Model statements for module:  Assign 3
;
16$             ASSIGN:         aInTime=tnow:
                                aSigNum=vSigNum:
                                aReleasedFromPRP=0:
                                aSchdPRPReleaseTime=tnow+ePRPDelay:
                                aDueDate=tnow+eFlowAllowance:NEXT(1$);


;
;
;     Model statements for module:  Separate 1
;
1$              DUPLICATE,      100 - 0:
                                1,57$,0:NEXT(56$);

56$             ASSIGN:         Separate 1.NumberOut Orig=Separate 1.NumberOut Orig +
1:NEXT(5$);

57$             ASSIGN:         Separate 1.NumberOut Dup=Separate 1.NumberOut Dup +
1:NEXT(2$);


;
;
;     Model statements for module:  Store 1
;
5$              STORE:          strPRP:NEXT(24$);


;
;
;     Model statements for module:  Decide 3
;
24$             BRANCH,         1:
                                If,eIncomingCheck == 1,58$,Yes:
                                Else,59$,Yes;
58$             ASSIGN:         Allow Incoming Check.NumberOut True=Allow Incoming
Check.NumberOut True + 1:NEXT(7$);

59$             ASSIGN:         Allow Incoming Check.NumberOut False=Allow Incoming
Check.NumberOut False + 1:NEXT(10$);


;
;
;     Model statements for module:  Decide 1
```

```
;
7$              BRANCH,         1:
                                If,(NR(rMachine) == 0) && (NSTO(strPRP)
>=eReleaseLimit),60$,Yes:
                                Else,61$,Yes;
60$             ASSIGN:         Decide Early Release.NumberOut True=Decide Early
Release.NumberOut True + 1:NEXT(8$);

61$             ASSIGN:         Decide Early Release.NumberOut False=Decide Early
Release.NumberOut False + 1:NEXT(10$);


;
;
;     Model statements for module:  Separate 2
;
8$              DUPLICATE,      100 - 0:
                                1,64$,0:NEXT(63$);

63$             ASSIGN:         DuplToTriggerEarlyRelease.NumberOut
Orig=DuplToTriggerEarlyRelease.NumberOut Orig + 1:NEXT(10$);

64$             ASSIGN:         DuplToTriggerEarlyRelease.NumberOut
Dup=DuplToTriggerEarlyRelease.NumberOut Dup + 1:NEXT(25$);


;
;
;     Model statements for module:  Decide 2
;
10$             BRANCH,         1:
                                If,aReleasedFromPRP==0,65$,Yes:
                                Else,66$,Yes;
65$             ASSIGN:         Decide If Real Job.NumberOut True=Decide If Real
Job.NumberOut True + 1:NEXT(11$);

66$             ASSIGN:         Decide If Real Job.NumberOut False=Decide If Real
Job.NumberOut False + 1:NEXT(18$);


;
;
;     Model statements for module:  Hold 1
;
11$             QUEUE,          PRPHold.Queue;
                WAIT:           aSigNum:NEXT(13$);


;
;
;     Model statements for module:  Unstore 1
;
13$             UNSTORE:        strPRP:NEXT(28$);


;
;
;     Model statements for module:  Record 9
;
28$             COUNT:          Total PRP Release Count,1:NEXT(17$);


;
;
;     Model statements for module:  Assign 4
;
17$             ASSIGN:         aReleasedFromPRP=1:
                                aPRPExitTime=tnow:
                                vNumInPRP=vNumInPRP-1:
                                vNumInQSys=vNumInQSys+1:
                                aReleasedOnSchedule=tnow == aSchdPRPReleaseTime:
                                Picture=Picture.Blue Ball:NEXT(37$);


;
```

```
;
;     Model statements for module:  Decide 5
;
37$            BRANCH,        1:
                              If,aReleasedOnSchedule==0,67$,Yes:
                              Else,68$,Yes;
67$            ASSIGN:        Released Early.NumberOut True=Released Early.NumberOut True
+ 1:NEXT(40$);

68$            ASSIGN:        Released Early.NumberOut False=Released Early.NumberOut
False + 1:NEXT(39$);


;
;
;     Model statements for module:  Assign 6
;
40$            ASSIGN:        Picture=Picture.Yellow Ball:NEXT(6$);


;
;
;     Model statements for module:  Process 1
;
6$             ASSIGN:        Machine.NumberIn=Machine.NumberIn + 1:
                              Machine.WIP=Machine.WIP+1;
72$            QUEUE,         Machine.Queue;
71$            SEIZE,         2,VA:
                              rMachine,1:NEXT(70$);

70$            DELAY:         aProcTime,,VA;
69$            RELEASE:       rMachine,1;
117$           ASSIGN:        Machine.NumberOut=Machine.NumberOut + 1:
                              Machine.WIP=Machine.WIP-1:NEXT(7$);


;
;
;     Model statements for module:  Record 17
;
39$            TALLY:         Scheduled Release Interarrival Tally,BET,1:NEXT(38$);


;
;
;     Model statements for module:  Record 16
;
38$            COUNT:         Scheduled PRP Release Count,1:NEXT(6$);


;
;
;     Model statements for module:  Assign 5
;
18$            ASSIGN:        aCompletionTime=tnow:
                              vNumInQSys=vNumInQSys-1:
                              vNumInWhole=vNumInWhole-1:
                              aTardiness=max(0,aCompletionTime-aDueDate):
                              aEarliness=max(0,aDueDate-aCompletionTime):
                              aRevenue=eCompletedProfit:NEXT(19$);


;
;
;     Model statements for module:  Record 1
;
19$            TALLY:         PRPTimeTally,aPRPExitTime-aInTime,1:NEXT(20$);


;
;
;     Model statements for module:  Record 2
;
20$            TALLY:         QSys Time Tally,aCompletionTime-aPRPExitTime,1:NEXT(21$);
```

```
;
;
;     Model statements for module:  Record 3
;
21$           TALLY:         Whole Time Tally,aCompletionTime-aInTime,1:NEXT(22$);


;
;
;     Model statements for module:  Record 4
;
22$           TALLY:         Tardiness Tally,aTardiness,1:NEXT(23$);


;
;
;     Model statements for module:  Record 5
;
23$           TALLY:         Earliness Tally,aEarliness,1:NEXT(45$);


;
;
;     Model statements for module:  Record 18
;
45$           TALLY:         Exiting Profit Tally,aRevenue,1:NEXT(47$);


;
;
;     Model statements for module:  Record 20
;
47$           TALLY:         AcceptedProfitTally,aRevenue,1:NEXT(42$);


;
;
;     Model statements for module:  Store 2
;
42$           STORE:         Finished Goods:NEXT(41$);


;
;
;     Model statements for module:  Delay 2
;
41$           DELAY:         max(0, aDueDate-tnow),,NVA:NEXT(43$);


;
;
;     Model statements for module:  Unstore 2
;
43$           UNSTORE:       Finished Goods:NEXT(15$);


;
;
;     Model statements for module:  Dispose 3
;
15$           ASSIGN:        Dispose Processed Jobs.NumberOut=Dispose Processed
Jobs.NumberOut + 1;
120$          DISPOSE:       No;


;
;
;     Model statements for module:  Record 6
;
25$           COUNT:         Early PRP Release Count,1:NEXT(26$);


;
;
```

```
;     Model statements for module:  Record 7
;
26$           TALLY:         Early PRP Release Contents Tally,vNumInPRP,1:NEXT(27$);


;
;
;     Model statements for module:  Record 8
;
27$           TALLY:         Earlyness of Release
Tally,a(nsym(aSchdPRPReleaseTime),FirstInQ(PRPHold.Queue))-TNOW,1:NEXT(36$);


;
;
;     Model statements for module:  Record 15
;
36$           TALLY:         Early Release Interarrival Tally,BET,1:NEXT(9$);


;
;
;     Model statements for module:  Signal 2
;
9$            SIGNAL:        a(nsym(aSigNum),FirstInQ(PRPHold.Queue)):NEXT(14$);


;
;
;     Model statements for module:  Dispose 2
;
14$           ASSIGN:        Dispose Early Release Triggers.NumberOut=Dispose Early
Release Triggers.NumberOut + 1;
121$          DISPOSE:       No;


;
;
;     Model statements for module:  Delay 1
;
2$            DELAY:         ePRPDelay,,NVA:NEXT(3$);


;
;
;     Model statements for module:  Signal 1
;
3$            SIGNAL:        aSigNum,1:NEXT(4$);


;
;
;     Model statements for module:  Dispose 1
;
4$            ASSIGN:        Dispose Scheduled Release Triggers.NumberOut=Dispose
Scheduled Release Triggers.NumberOut + 1;
122$          DISPOSE:       No;
;
```

C.2 Experiment (.exp) File

```
PROJECT,        "MM1 ORR Testbed","Yannai Segal",,,No,Yes,Yes,Yes,No,No,No;

ATTRIBUTES:     aRevenue:
                aInTime:
                aCompletionTime:
                aReleasedFromPRP:
                aTardiness:
                aSchdPRPReleaseTime:
                aEarliness:
                aReleasedOnSchedule:
                aPRPExitTime:
                aSigNum:
```

```
              aProcTime:
              aDueDate;

STORAGES:     Finished Goods:
              strPRP;

VARIABLES:    Dispose Scheduled Release
Triggers.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
              Released Early.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
              vMPT,CLEAR(System),CATEGORY("User Specified"),1:
              vNumInQSys,CLEAR(System),CATEGORY("User Specified"):
              Machine.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
              vServiceStream,CLEAR(System),CATEGORY("User Specified"),2:
              Decide Early Release.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
              Reject Incoming Job if N Exceeded.NumberOut
False,CLEAR(Statistics),CATEGORY("Exclude"):
              Dispose Early Release
Triggers.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
              vIncomingCheck,CLEAR(System),CATEGORY("User Specified"),1:
              vRL,CLEAR(System),CATEGORY("User Specified"),999999:
              Decide If Real Job.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
              Decide If Real Job.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
              Allow Incoming Check.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
              vMIAT,CLEAR(System),CATEGORY("User Specified"),1.33333333:
              Machine.NumberIn,CLEAR(Statistics),CATEGORY("Exclude"):
              Dispose Processed Jobs.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
              Dispose Rejected Jobs.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
              Allow Incoming Check.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
              vNumInPRP,CLEAR(System),CATEGORY("User Specified"):
              DuplToTriggerEarlyRelease.NumberOut
Orig,CLEAR(Statistics),CATEGORY("Exclude"):
              Create Arriving Jobs.NumberOut,CLEAR(Statistics),CATEGORY("Exclude"):
              Released Early.NumberOut False,CLEAR(Statistics),CATEGORY("Exclude"):
              vInterarrivalStream,CLEAR(System),CATEGORY("User Specified"),1:
              Separate 1.NumberOut Dup,CLEAR(Statistics),CATEGORY("Exclude"):
              Machine.WIP,CLEAR(System),CATEGORY("Exclude-Exclude"):
              Decide Early Release.NumberOut True,CLEAR(Statistics),CATEGORY("Exclude"):
              vSigNum,CLEAR(System),CATEGORY("User Specified"):
              vkm,CLEAR(System),CATEGORY("User Specified"),1:
              vkr,CLEAR(System),CATEGORY("User Specified"),10:
              vd,CLEAR(System),CATEGORY("User Specified"),0:
              vm,CLEAR(System),CATEGORY("User Specified"),29.9573:
              vN,CLEAR(System),CATEGORY("User Specified"),999999:
              vNumInWhole,CLEAR(System),CATEGORY("User Specified"):
              DuplToTriggerEarlyRelease.NumberOut
Dup,CLEAR(Statistics),CATEGORY("Exclude"):
              Separate 1.NumberOut Orig,CLEAR(Statistics),CATEGORY("Exclude"):
              Reject Incoming Job if N Exceeded.NumberOut
True,CLEAR(Statistics),CATEGORY("Exclude");

QUEUES:       Machine.Queue,FIFO,,AUTOSTATS(Yes,,):
              PRPHold.Queue,FIFO,,AUTOSTATS(Yes,,);

PICTURES:     Picture.Airplane:
              Picture.Green Ball:
              Picture.Blue Page:
              Picture.Telephone:
              Picture.Blue Ball:
              Picture.Yellow Page:
              Picture.EMail:
              Picture.Yellow Ball:
              Picture.Bike:
              Picture.Report:
              Picture.Van:
              Picture.Widgets:
              Picture.Envelope:
              Picture.Fax:
              Picture.Truck:
              Picture.Letter:
              Picture.Box:
              Picture.Woman:
              Picture.Package:
              Picture.Man:
              Picture.Diskette:
              Picture.Boat:
```

```
              Picture.Red Page:
              Picture.Green Page:
              Picture.Red Ball;

RESOURCES:
rMachine,Capacity(1),,,COST(0.0,0.0,0.0),CATEGORY(Resources),,AUTOSTATS(No,,);

COUNTERS:     Rejected Job Count,,,,DATABASE(,"Count","User Specified","Rejected Job
Count"):
              Scheduled PRP Release Count,,,,DATABASE(,"Count","User
Specified","Scheduled PRP Release Count"):
              Total PRP Release Count,,,,DATABASE(,"Count","User Specified","Total PRP
Release Count"):
              Accepted Job Count,,,,DATABASE(,"Count","User Specified","Accepted Job
Count"):
              Early PRP Release Count,,,,DATABASE(,"Count","User Specified","Early PRP
Release Count");

TALLIES:      Early Release Interarrival Tally,,DATABASE(,"Between","User
Specified","Early Release Interarrival Tally"):
              Accepted Jobs Interarrival Tally,,DATABASE(,"Between","User
Specified","Accepted Jobs Interarrival Tally"):
              AcceptedProfitTally,,DATABASE(,"Expression","User
Specified","AcceptedProfitTally"):
              Tardiness Tally,,DATABASE(,"Expression","User Specified","Tardiness
Tally"):
              Earliness Tally,,DATABASE(,"Expression","User Specified","Earliness
Tally"):
              Early PRP Release Contents Tally,,DATABASE(,"Expression","User
Specified","Early PRP Release Contents Tally"):
              Whole Time Tally,,DATABASE(,"Expression","User Specified","Whole Time
Tally"):
              Rejected Job Interarrival Tally,,DATABASE(,"Between","User
Specified","Rejected Job Interarrival Tally"):
              PRPTimeTally,,DATABASE(,"Expression","User Specified","PRPTimeTally"):
              Earlyness of Release Tally,,DATABASE(,"Expression","User
Specified","Earlyness of Release Tally"):
              QSys Time Tally,,DATABASE(,"Expression","User Specified","QSys Time
Tally"):
              Scheduled Release Interarrival Tally,,DATABASE(,"Between","User
Specified","Scheduled Release Interarrival Tally"):
              Actual Interarrival Time Tally,,DATABASE(,"Between","User
Specified","Actual Interarrival Time Tally"):
              Exiting Profit Tally,"",DATABASE(,"Expression","User Specified","Exiting
Profit Tally");

DSTATS:       vNumInQSys,DSNumInQSys,"",DATABASE(,"Time Persistent","User
Specified","DSNumInQSys"):
              vNumInPRP,DSNumInPRP,"",DATABASE(,"Time Persistent","User
Specified","DSNumInPRP"):
              nsto(Finished Goods),DSNumInFinishedGoods,"",DATABASE(,"Time
Persistent","User Specified","DSNumInFinishedGoods"):
              (NR(rMachine) == 0) && (NSTO(strPRP) >
1),DSInsertedIdleTime,"",DATABASE(,"Time Persistent","User Specified",
              "DSInsertedIdleTime"):
              vNumInWhole,DSNumInWhole,"",DATABASE(,"Time Persistent","User
Specified","DSNumInWhole");

FREQUENCIES:  Value(vNumInQSys),QSysFreq,"",DATABASE(,"Frequency","User
Specified","QSysFreq"),Constant(0),S00,Include&Constant(1),

S01,Include&Constant(2),S02,Include&Constant(3),S03,Include&Constant(4),S04,Include&Const
ant(5),S05,Include&

Constant(6),S06,Include&Constant(7),S07,Include&Constant(8),S08,Include&Constant(9),S09,I
nclude&Constant(10),S10,

Include&Constant(11),S11,Include&Constant(12),S12,Include&Constant(13),S13,Include&Consta
nt(14),S14,Include&

Constant(15),S15,Include&Constant(16),S16,Include&Constant(17),S17,Include&Constant(18),S
18,Include&Constant(19),

S19,Include&Constant(20),S20,Include&Constant(21),S21,Include&Constant(22),S22,Include&Co
nstant(23),S23,Include&
```

```
Constant(24),S24,Include&Constant(25),S25,Include&Constant(26),S26,Include&Constant(27),S
27,Include&Constant(28),

S28,Include&Constant(29),S29,Include&Constant(30),S30,Include&Constant(31),S31,Include&Co
nstant(32),S32,Include&

Constant(33),S33,Include&Constant(34),S34,Include&Constant(35),S35,Include&Constant(36),S
36,Include&Constant(37),

S37,Include&Constant(38),S38,Include&Constant(39),S39,Include&Constant(40),S40,Include:
              Value(vNumInPRP),PRPFreq,"",DATABASE(,"Frequency","User
Specified","PRPFreq"),Constant(0),P00,Include&Constant(1),

P01,Include&Constant(2),P02,Include&Constant(3),P03,Include&Constant(4),P04,Include&Const
ant(5),P05,Include&

Constant(6),P06,Include&Constant(7),P07,Include&Constant(8),P08,Include&Constant(9),P09,I
nclude&Constant(10),P10,

Include&Constant(11),P11,Include&Constant(12),P12,Include&Constant(13),P13,Include&Consta
nt(14),P14,Include&

Constant(15),P15,Include&Constant(16),P16,Include&Constant(17),P17,Include&Constant(18),P
18,Include&Constant(19),

P19,Include&Constant(20),P20,Include&Constant(21),P21,Include&Constant(22),P22,Include&Co
nstant(23),P23,Include&

Constant(24),P24,Include&Constant(25),P25,Include&Constant(26),P26,Include&Constant(27),P
27,Include&Constant(28),
              P28,Include&Constant(29),P29,Include&Constant(30),P30,Include:
              Value(NSTO(Finished Goods)),FGFreq,"",DATABASE(,"Frequency","User
Specified","FGFreq"),Constant(0),FG00,Include&

Constant(1),FG01,Include&Constant(2),FG02,Include&Constant(3),FG03,Include&Constant(4),FG
04,Include&Constant(5),

FG05,Include&Constant(6),FG06,Include&Constant(7),FG07,Include&Constant(8),FG08,Include&C
onstant(9),FG09,Include&

Constant(10),FG10,Include&Constant(11),FG11,Include&Constant(12),FG12,Include&Constant(13
),FG13,Include&Constant(14),

FG14,Include&Constant(15),FG15,Include&Constant(16),FG16,Include&Constant(17),FG17,Includ
e&Constant(18),FG18,

Include&Constant(19),FG19,Include&Constant(20),FG20,Include&Constant(21),FG21,Include&Con
stant(22),FG22,Include&

Constant(23),FG23,Include&Constant(24),FG24,Include&Constant(25),FG25,Include&Constant(26
),FG26,Include&Constant(27),

FG27,Include&Constant(28),FG28,Include&Constant(29),FG29,Include&Constant(30),FG30,Includ
e;

OUTPUTS:      11,vMIAT,"",MeanIAT,DATABASE(,"Output","User Specified","MeanIAT"):
              12,vMPT,"",MeanPT,DATABASE(,"Output","User Specified","MeanPT"):
              13,vm,"",Flow Allowance m,DATABASE(,"Output","User Specified","Flow
Allowance m"):
              21,vkm,"",km CostPar,DATABASE(,"Output","User Specified","km CostPar"):
              22,vkr,"",kr CostPar,DATABASE(,"Output","User Specified","kr CostPar"):
              31,vd,"",Delay d,DATABASE(,"Output","User Specified","Delay d"):
              32,vN,"",Work Limit N,DATABASE(,"Output","User Specified","Work Limit N"):
              33,vRL,"",Release Limit RL,DATABASE(,"Output","User Specified","Release
Limit RL"):
              34,vIncomingCheck,"",Incoming Check,DATABASE(,"Output","User
Specified","Incoming Check"):
              35,vInterarrivalStream,"",Arrival RnStream,DATABASE(,"Output","User
Specified","Arrival RnStream"):
              36,vServiceStream,"",Service RnStream,DATABASE(,"Output","User
Specified","Service RnStream"):
              41,TAVG(Earliness Tally),"",AvgEarliness,DATABASE(,"Output","User
Specified","AvgEarliness"):
```

```
                42,TAVG(Tardiness Tally),"",AvgTardiness,DATABASE(,"Output","User
Specified","AvgTardiness"):
                44,100*NC(Accepted Job Count)/(NC(Accepted Job Count)+NC(Rejected Job
Count)),"",Percent Accepted,DATABASE(,
                "Output","User Specified","Percent Accepted"):
                45,100*NC(Early PRP Release Count) / NC(Total PRP Release
Count),"",%ReleasedEarly,DATABASE(,"Output",
                "User Specified","%ReleasedEarly"):
                52,(OVALUE(Percent Accepted))*(1-tavg(Tardiness Tally)/(vm*vkm*vMPT)-
tavg(Earliness Tally)/(vm*vkm*vkr*vMPT)),"",
                Calculated PPUAR,DATABASE(,"Output","User Specified","Calculated PPUAR"):
                53,TAVG(Exiting Profit Tally),"",Mean Tallied
PPUAR,DATABASE(,"Output","User Specified","Mean Tallied PPUAR"):
                54,100*Thalf(Exiting Profit Tally) / TAVG(Exiting Profit Tally),"",CiHw%
PPUAR,DATABASE(,"Output","User Specified",
                "CiHw% PPUAR"):
                55,TAVG(AcceptedProfitTally),"",AvgAccPPUAR,DATABASE(,"Output","User
Specified","AvgAccPPUAR"):
                61,tavg(Actual Interarrival Time Tally),"",Actual
IAT,DATABASE(,"Output","User Specified","Actual IAT"):
                62,orunhalf(Calculated PPUAR)*(nrep == (mrep-1))*mrep,"",PPUAR
HW,DATABASE(,"Output","User Specified","PPUAR HW");

REPLICATE,      50,,DaysToBaseTime(1050),Yes,Yes,DaysToBaseTime(50),,,24,Minutes,No,No;

EXPRESSIONS:    eIAT,expo(vMIAT, vInterarrivalStream):
                ePRPDelay,vd*vMPT:
                ePT,expo(vMPT, vServiceStream):
                eSystemWorkLimit,vN:
                eFlowAllowance,vm*vMPT:
                eReleaseLimit,vRL:
                eCompletedProfit,1-(aTardiness+aEarliness/vkr)/(vkm*vm*vMPT):
                eIncomingCheck,vIncomingCheck;

ENTITIES:       Job,Picture.Report,0.0,0.0,0.0,0.0,0.0,0.0,AUTOSTATS(No,,);
```