UNIVERSITY OF ALBERTA

ANALYSIS OF INTERMEDIATE STORAGE IN A CONTINUOUS MINING OPERATION

by

DONALD CHARLES DOE ©

A THESIS

SUBMITTED TO THE FACULTY OF GRADUATE STUDIES AND RESEARCH

IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE

OF MASTER OF SCIENCE

IN

MINING ENGINEERING

DEPARTMENT OF MINING, METALLURGICAL AND PETROLEUM

ENGINEERING

EDMONTON, ALBERTA

SPRING 1990

ISBN 0-315-60316-X

UNIVERSITY OF ALBERTA

RELEASE FORM

NAME OF AUTHOR DONALD CHARLES DOE

TITLE OF THESIS ANALYSIS OF INTERMEDIATE STORAGE IN A CONTINUOUS MINING OPERATION

DEGREE FOR WHICH THESIS WAS PRESENTED MASTER OF SCIENCE

YEAR THIS DEGREE GRANTED SPRING 1990

(SIGNED) ....

PERMANENT ADDRESS:

#408 109 STROUD BAY

FORT McMURRAY, ALBERTA

T9H 4Z6

DATED APRIL 20 1990

UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES AND RESEARCH

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research, for acceptance, a thesis entitled ANALYSIS OF INTERMEDIATE STORAGE IN A CONTINUOUS MINING OPERATION submitted by DONALD CHARLES DOE in partial fulfilment of the requirements for the degree in MASTER OF SCIENCE in MINING ENGINEERING.

W. H. Griffin
Prof. W.H. Griffin
Supervisor

Ramon Bentsen
Dr. R.G. Bentsen

Dr. R.K. Wood

J M Whiting
Dr. J.M. Whiting

Date. April 20 1990

Dedication

To my parents who taught me to believe in myself and were always there when needed.

and

To Louisa, who listened to my complaints and problems during the course of this thesis. Your words of encouragement and support helped to make this possible.

Abstract

Work has been performed to analyze the steady state effect of storage between continuous production lines. This effect has been simulated by a model constructed with SLAM (Simulation Language for Alternative Modelling) and analyzed using non-linear regression. The model is a one stream in, one stream out operation, with each stream having the same model of random failures with respect to operating time and random repair time. The expected production of input and output streams is equal. The steady state production for variations in storage volume has been compared to the analytical values for infinite and zero volume storage. From these results, a function relating system losses to bin size normalized by machine failure and repair time parameters has been derived. This allows calculation of expected production regardless of the absolute capacity of the system.

The normalized model has been compared to empirical work done previously. A simple economic analysis of the benefits of including a bin in a system has been done to illustrate a possible use for the model.

A technique from operations research, perturbation analysis, has been applied to the system and compared to the results provided by the model.

ACKNOWLEDGEMENTS

This thesis was done over an extended period of time, so there are many people that need to be thanked.

First and foremost my thanks to Professor W.H. Griffin for the opportunity to work on this project and for his patience. By providing me with the proper direction and then letting me find my own mistakes, the educational benefit of this thesis was larger than the work itself.

My thanks also to Mr. Doug Booth for helping with the PC version of the model and your suggestions along the way.

The occupants of Room 280, several different groups, are also remembered. They provided help, support, and made the experience enjoyable.

List of Figures

List of Tables

Table of Contents

1. Introduction

1.1 Motivation

Due to the harsh operating conditions that equipment in the mining industry is subjected to, machine stoppages are inevitable. One method of minimizing the impact of these shutdowns on up and/or downstream equipment is to place a volume of intermediate storage (a bin) between successive machines. This bin then has the effect of isolating the machines from one another and a productivity gain is realized.

In many operations, the number of interconnecting machines is very large, i.e. several shovels loading a fleet of trucks which then dump to a stockpile, which in turn feeds a processing plant. These large numbers cause complex interactions, and the actual effect of increasing the intermediate storage volume becomes difficult to evaluate. The storage volume becomes less important as the failure of an individual unit (trucks, shovels) only marginally alters productivity. Clearly, in this large system, bin volume is not the only parameter capable of improving system productivity; the machine failure parameters (time interval between failures and the duration of failures) are also significant. Any improvement in production attributable solely to the increase in storage volume is obscured by these other parameters as well as the system interactions.

Research by the mineral industry with respect to the productivity gains realized by increasing bin volume has generally been on a site-specific basis. The goal of this thesis is to produce a procedure for generalizing the results from a specific model that they may be extended to more general models.

1.2 Direction of Thesis

In order to elucidate the mathematical theory underlying the bin sizing problem, a simplified model of a mining operation is constructed. The model consists of one continuous stream, "the mine" feeding into a bin, which is then emptied by a single continuous stream, "the processing plant". This simplified model is, in theory, applicable to some mining operations such as the oil sands mines of northern Alberta and the brown coal mines of East Germany (15,22,26). In the case of the oil sands mines, large scale single or dual machines (bucketwheel excavators) supply continuous feed to a processing plant. (Figure 1.1). These plants have a continuous demand for feed, and long term shortages, especially in the winter months, can cause severe operational problems. Increase of the bin volume between the mine and the processing plant minimizes the occurrence of these shortages. The quantification of the value of the expected increase in production with respect to the increase in bin volume is the objective of this thesis.

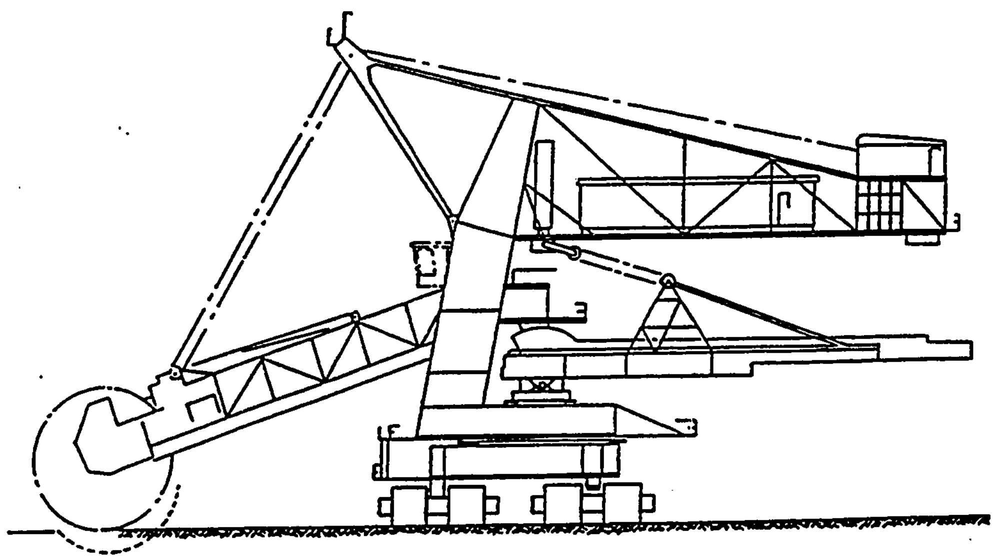

Figure 1.1 Schematic of bucketwheel.

As this work is directed at engineers, and a prime concern of any engineer is a cost benefit analysis, the model is utilized in a simplified analysis. The goal is to provide the design engineer with the analytical tools to quantitatively evaluate the effects of bin size on the productivity of simple, balanced systems. Specifically, for a simple, balanced system a functional equation is developed that quantitatively relates expected production to bin size. The independent variables of this function are mean time to failure (MTTF), mean time to repair (MTTR) and production rate. The coefficient of variation (COV) of the MTTF and MTTR remain fixed throughout the process. It is then illustrated that this equation may be used on different sets of simple, balanced conditions by normalizing the bin volume by a function of the input parameters (MTTF, MTTR and production rate) of both streams. Finally, the applicability of an operations research related technique (13,19) is examined. This technique has the capability to greatly reduce the amount of computer time required, and may be applied using operating history alone, or through simulation.

1.3 Outline of Thesis

The final section in this chapter provides a brief introduction to the theory of simple transfer lines and the terminology involved. Chapter 2 contains a review of previous bin sizing research in the mineral industry as well as applicable work from the operations research field. In Chapter 3, the simulation model is explained and the experimental design outlined. The simulation results, statistical analysis and function derivation are also included in Chapter 3. Chapter 4 contains a comparison of the results against an analytical formula, an extension of the model to other simple, balanced systems, and a sample economic analysis. An operations research technique of perturbation analysis is introduced in Chapter 5 and a detailed comparison is made with the results from this method and those obtained in Chapter 3. The final chapter contains a summary of the results, offers some conclusions, and suggests areas for future research.

1.4 Transfer Lines

1.4.1 Basics of Transfer Lines

The model utilized in this thesis is a basic, single stage transfer line. There is a single machine, feeding into a volume of intermediate storage, which feeds a single output machine. An initial simplifying assumption is that machine 1 (input) is supplied by an infinite storage and machine 2 (output) feeds into an infinite sink, any external constraints are then removed and it is possible to study the transfer line in isolation. The line is depicted in Figure 1.2.

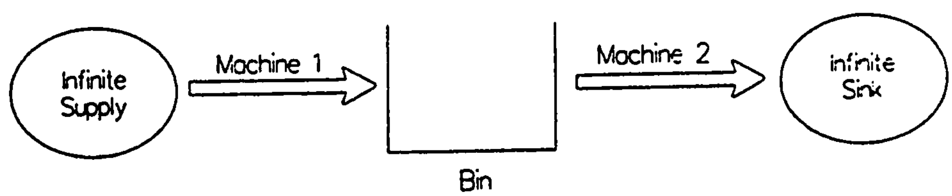

Figure 1.2 A simple transfer line

As both machine 1 and 2 are imperfect, they are subject to internal failures. Scheduled (planned) maintenance are differentiated from the random (unplanned) failures. What is said is that, on average, machine 1 or 2 will operate for X hours and then be down (unavailable for production) for Y hours. As the system is balanced, by assigning equal expected production rates to both input and output, the operating (X) and non-operating (Y) durations are equivalent for both machines. For the purpose of this thesis, the key point is not the event that caused the machine to fail; but the fact that the machine is unavailable for production. It is suggested that as the occurrence and duration of the planned outages are known in advance, the time remaining for production may be calculated, and only random failures considered. For either machine in the system, input (in) or output (out) , the mean time to failure (MTTF) and mean time to repair (MTTR) is the expected value (long term average) of the time intervals that either machine operates between failures or is down for repair, respectively. The production rate (PROD) of each system is the average uninterrupted sustainable operating rate. In the model used in this thesis this rate is considered to be achievable instantaneously on start up and stops similarly.

The system may be considered a first order, balanced system since all the machine parameters are equivalent, i.e.

$$MTTF_{in} = MTTF_{out}$$

$$MTTR_{in} = MTTR_{out}$$

$$PROD_{in} = PROD_{out}$$

A system may still be balanced, but have different production values (PROD) and/or failure parameters. This type would be a second order balanced system. For the purposes of this thesis, only first order systems are considered.

In the material follows, the terminology of Ho (19) is used to define the states of the machines. If there is no volume of intermediate storage, a failure of one machine leads to a Forced Down (FD) state of the other machine. If machine 1 fails, machine 2 does not have any material to process and is in a No Input (NI) state. Conversely, if machine 2 has failed machine 1 has no place for its production and is then in a No Output (NO) state. In either condition, both machines are down and production is zero until the failed machine returns to production. The system can produce only when both machines are "up", and the proportion of time that the line is functioning is the limiting availability, π_0 of the system. Barlow and Proschan (2) give a rigorous proof of calculating the system "up" time over an interval [0, t] with general failure and repair distributions. The major assumptions of Barlow and Proschan are as follows, (using their symbols):

1. System is composed of k serially connected components.
2. All k components have a finite mean operating time to failure (MTTF) of μ_j, j = 1,2,...k
3. The mean time to repair (MTTR) is ν_j, j = 1,2,...k, and repairs result in a "like new" component.
4. No preventative maintenance is done at forced down (FD) points.

5. The probability of failure of two units at exactly the same instant is zero.

6. μ_j and ν_j are stationary, mutually-independent random variables.

7. Failures are based on operating time.

Given these assumptions, and letting U(t) = the time that the system is operative in the interval [0, t];

$$\lim_{t \to \infty} \frac{U(t)}{t} = \frac{1}{1 + \sum_{i=1}^{J} \frac{\nu_i}{\mu_i}} = \pi_0 \qquad \text{..........1.1}$$

Note that this gives a different value for the limiting availability than would a product rule for availability (with failures based on both operating and non-operating time). Where the availability, or proportion of time a machine spends in the operating state, is defined as A_i and;

$$A_i = \left[\frac{\mu_i}{\mu_i + \nu_i}\right] \qquad \text{.....................1.2}$$

The limiting availability, π_0', with the product rule now becomes;

$$\pi_0' = \left[\frac{\mu_1}{\mu_1 + \nu_1}\right] \cdot \left[\frac{\mu_2}{\mu_2 + \nu_2}\right] \qquad \text{...........1.3}$$

This product rule formula does not take into account assumptions (5) and (7). As a machine, by virtue of assumption (7), can deteriorate only while it is working, it may not fail while it has been forced down. Since μ_j is operating time to failure, the time spent in the forced down state must be considered when evaluating the limiting availability of the system.

A heuristic approach would be to evaluate the possible states of the simple two machine system, and account for time lost due to FD states. The possible combinations of states for the system are:

1. M_1 up, M_2 up
2. M_1 up, M_2 down
3. M_1 down, M_2 up
4. M_1 down, M_2 down

Note that state 4 is not possible, as both machines would have to fail at precisely the same moment in time. This "invalid" state, therefore must be removed from the time line. The total probability space is then reduced by the "invalid" amount.

for example, the probability of M_1 up, and M_2 up equals:

$$\left[\frac{\mu_1}{\mu_1 + \nu_1}\right] \cdot \left[\frac{\mu_2}{\mu_2 + \nu_2}\right]$$

and the probability M_1 down, M_2 down equals:

$$\left[\frac{\nu_1}{\mu_1 + \nu_1}\right] \cdot \left[\frac{\nu_2}{\mu_2 + \nu_2}\right]$$

The limiting availability of the system is then found by reducing the whole probability space of unity by the invalid amount:

$$\pi_0 = \frac{\left[\frac{\mu_1}{\mu_1 + \nu_1}\right] \cdot \left[\frac{\mu_2}{\mu_2 + \nu_2}\right]}{1 - \left[\frac{\nu_1}{\mu_1 + \nu_1}\right] \cdot \left[\frac{\nu_2}{\mu_2 + \nu_2}\right]} \qquad \ldots\ldots.1.4$$

This equation can be reduced to Equation 1.1. The associated algebra is in Appendix A.

As an otherwise "healthy" machine is forced down and valuable production time is lost, this is an inefficient use of the machines. The production of this system may be improved by one (or a combination) of the following methods:

1. Increase the operating time for the machines
2. Decrease the repair time of the machines
3. Insert a volume of intermediate storage between the two machines

The implications of (1) and (2) are that more time is spent in the operational state, and less time in the failure state. Note that the equations previously defined remain valid for (1) and (2) only, and the new system availability and productivity may be calculated.

Calculating the gain in productivity by inserting (or increasing) the bin volume, option (3), is not as straight forward. Equation 1.1 is not valid when any bins are included in the system. Inserting the bin tends to isolate production histories of the two machines. That is, assuming there is some volume of material in the bin when machine 1 fails, machine 2 will continue to operate. Conversely, if machine 2 fails, and there remains some unused volume, machine 1 will continue to operate. Clearly, the larger the bins, the more isolated the machines become. Bins of infinite volume would completely isolate the machines and allow the system production to approach that of its weakest member. Defining the production of the system as a function of the bin volume, BIN;

$$\lim_{BIN \to \infty} PROD(BIN) = PROD_{min}$$

where:

$PROD_{min}$ = Minimum possible continuous production rate of machines 1 and 2. $(Min(A_1*PROD_1, A_2*PROD_2))$

Because an infinitely large bin is neither possible, nor practical, one must find a quantification of bin volume increment with respect to production gain. This quantification will be a function of the mean time to failure, and the mean duration of these failures, as well as the production rate. The final bin size will be determined by the incremental increase in production versus the cost of the added volume.

The methods previously used to calculate this increase in production will be outlined in the following chapter.

2. Background

2.1 Simulation

Simulation is a broad term, that encompases many different methodologies. By simulating a model of reality, we are able to pose various "what-ifs" without actually introducing any upset to reality. This modelling may occur in a variety of ways from iconic (physical) to the type of mathematical modelling employed here.

Utilizing a mathematical description of the system under study, the engineer may use the power of digital computers to further explore system operation. Equations may be of a strictly deterministic form, i.e. given a set of inputs, one, unique output set is generated. Conversely, the simulation may be stochastic. With stochastic simulation, a sequence of pseudo-random numbers (20) drives the problem logic and associated equations to a realization. One simulation experiment yields one realization of the possible set of outcomes from the underlying probability distribution and systems logic. The random number streams are generated by an algorithm that utilizes an initial or seed value. By altering these "seed" values, the engineer may perform multiple simulations, each yielding a different realization. From these multiple runs, statistical inferences may be made about the range and distribution of values likely to occur in reality. As the model is being used to predict what *might* happen, stochastic simulation is the prudent choice.

Verification of an assumption or approval of a project based solely on a single run is very risky when some (or all) of the controlling factors are subject to random occurrences.

2.1.1 SLAM II Simulation Language

The SLAM II (Simulation Language For Alternative Modelling) (23) is a versatile, multi-purpose FORTRAN based simulation language. SLAM has been used previously to model systems in the mining industry, from oil sands plants to truck shovel operations. (16,17)

SLAM provides for the following categories of systems:

1. Discrete
2. Continuous
3. Combined

Discrete systems only change at certain times, i.e. a definable point in time such as a piece of equipment failing at time T, Fig. 2.1. Continuous systems have variables which change continuously over time, Fig. 2.2. These variables are modelled with either differential or difference equations. Chemical reactions, and tankage levels are two types of continuous systems. A combined system contains both discrete and continuous variables.

The system modelled here is a combined model. The discrete variables are the machine failure events, the continuous variables are the production rates and the level of material in the bin.

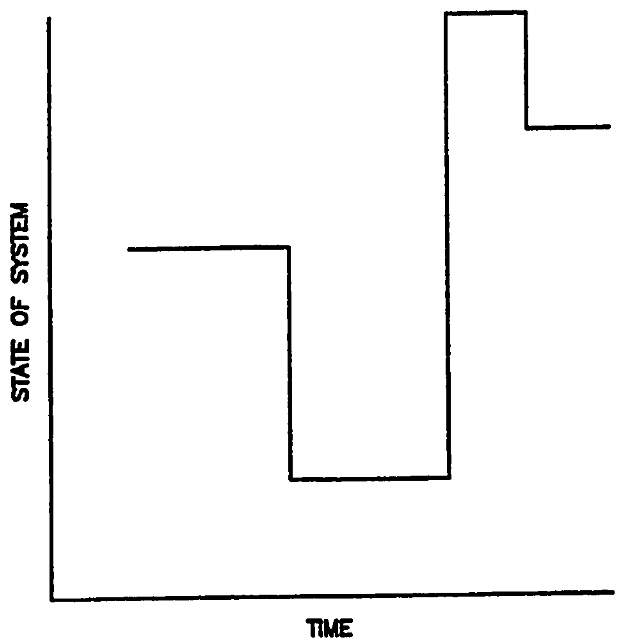

Figure 2.1 Discrete system state plot

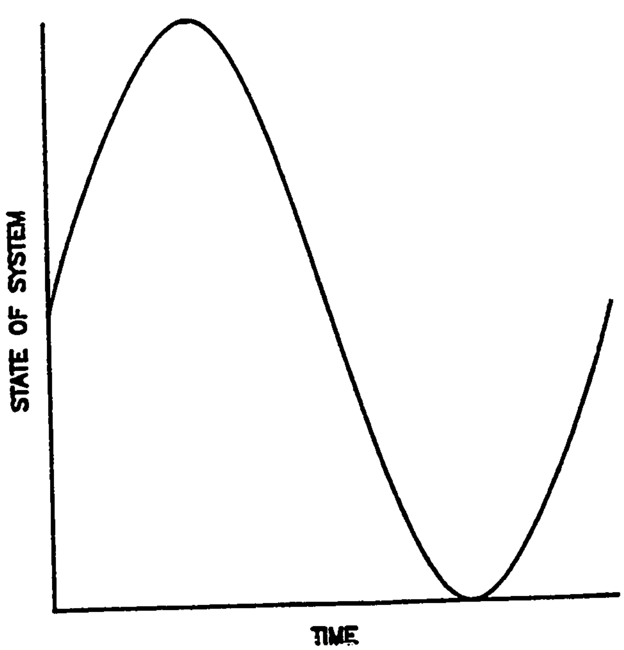

Figure 2.2 Continuous system state plot

SLAM is available for a wide configuration of hardware, from mainframe to personal computer. Presently, the personal computer (PC) version is limited in size by program memory constraints, but is large enough for most simulations. Simulation on personal computers is quite advantageous as the time is free once the machine has been purchased. A somewhat more subtle bonus when using the PC version, is that the modeller is constrained in the size of the model, in terms of both storage (memory) and time. Keeping these limits in mind, the modeller is forced to produce the most efficient model possible, eliminating extraneous details and thus producing the best tool possible.

2.2 Previous Bin Sizing Models in the Minerals Industry

While the use of computers in the mining industry is widespread, the question of surge bin volumes has not received the attention of other areas, such as rock mechanics and mine planning. The majority of literature with respect to bins concerns the flow characteristics of various materials.

Cruz et. al. (7) utilize a Monte Carlo simulation program to evaluate the most cost effective stockpile between a mine and mill. Chu and Ermolowich (5) use a similar approach to design the optimum capacity of a coal silo. This model incorporates some operating rules which considerably complicate the system. Once again, a silo capacity that maximizes productivity with minimum cost is the end result. Silo volumes are also examined by Bradely et. al. (3) to determine train waiting times. Fusinato (13) utilizes a detailed model of a bucket wheel coal mining operation to find the optimum "bunker" volume with respect to lost production and lifetime cost of the bunker.

There are several similarites between all of these studies namely:

1. The models are quite detailed and have complex interactions.
2. Results are site specific.
3. All use brute force simulation.

These investigations indicate that there is concern and awareness about the effectiveness of intermediate storage. What they fail to provide is a tool for an engineer to use, short of writing his own simulation model. All are very site specific, and no listing of the program code used is given, although in some cases it may be possible to obtain the code. A better understanding of the theory involved is required, so that a generalized tool can be produced and offered to engineers.

An analytical approach by J. Elbrond (11) attempts to relate system productivity with respect to bin volume for the general cases. Elbrond examines only a simple system, similar to the one considered in this thesis. While both equal and unequal internal capacity cases are considered by Elbrond, only the balanced case is examined here. The procedures and equations used to calculate system capacity based on system parameters and bin volume are outlined in Appendix A. It is important to note that Elbrond uses the product formula rule (Eq. 1.3) for calculating the system availability and therefore assumes that the machines deteriorate while forced down. While this provides for different limiting availabilities than would be calculated by the Barlow and Proschan method, a comparison is still made. The derivation of these formulas is empirical and therefore provides for comparisons against the proposed simulation generated function. It should be noted that the formulas are anything but simplistic and some of the background is unclear. It appears that the goal was a concave function, going asymptotically towards a limit and this is what was generated. Interestingly, Elbrond is presently validating this set of procedures through simulation (12).

The procedures and equations as provided by Elbrond have been coded. The FORTRAN code and sample output are given in Appendix A. A major drawback in this procedure is that the bin content frequency function had to be assumed. It is this function that determines the amount of time the bin spends at any particular level. As no better information was available, the same function was used for all bin volumes simulated. Intuitively, one would expect different functions as the bin volumes were changed, but without observing the system under consideration, this information would not be available. As the design engineer would be faced with a similar problem, the assumption of one function for all bins was used.

The values obtained from Elbrond's procedures will be compared to the experimentally derived function in later sections.

2.3 Operations Research Techniques

Operations research (OR) may be defined as "a scientific approach to problem-solving for executive management" (27). One modification that should be made to this definition is that operations research is not (and should not) be limited to only upper management. As will be illustrated later, these techniques may be applied at any level of a company. One application of OR is queueing theory and the associated study of transfer lines. Queueing theory is "the mathematical treatment of waiting lines" (6). While queueing theory does have applications in mining (i.e. a truck-shovel operation), its applicability to our production model is limited. Transfer lines, or production lines, however, initially offer an interesting comparison. These are the well-researched assembly lines which have become so prevalent in modern day manufacturing. The benefits of including some type of intermediate storage between successive machines has long been recognized (21,25). Koenigsberg (21) provides a detailed review of the available approaches as of 1958. Buzacott and Hanifin (4) offer a similar review, some twenty years later, in 1978.

The major stumbling blocks in applying this theory to many real world transfer lines are the restrictive assumptions placed upon certain characteristics of the system. The work done by Sevat'yanov (25) and reviewed by both Koeningsberg and Buzacott and Hanifin is one of the earliest studies in this area. The assumptions (4) placed upon the system are:

1. Only one stage can be in a failed state.
2. Failure rate of a stage is constant in time when both stages are operating.
3. Downtimes have exponential distributions.

Assumption (1) is not valid in actual practice. A machine may fail when another is down, if it has been permitted to continue operating. Very few events will be constant, especially failures and assumption (2) is again restrictive. Failure distributions are likely quite site specific, dependent upon operator skill, material type and climatic conditions. The assumption of one failure distribution for all sites is unwarranted. As well, the exponential distribution, Figure 2.3, does not have the required "shift" from zero. While it is not known how long a machine will be down, it is certain that it will be some non zero duration.

Another area of concern is that Sevat'yanov only examined storage and forced down conditions with respect to machine 2, i.e. how long machine 2 is idle due to machine 1 being under repair and the bin level at 0. Clearly, the opposite is also important, the bin full conditions that

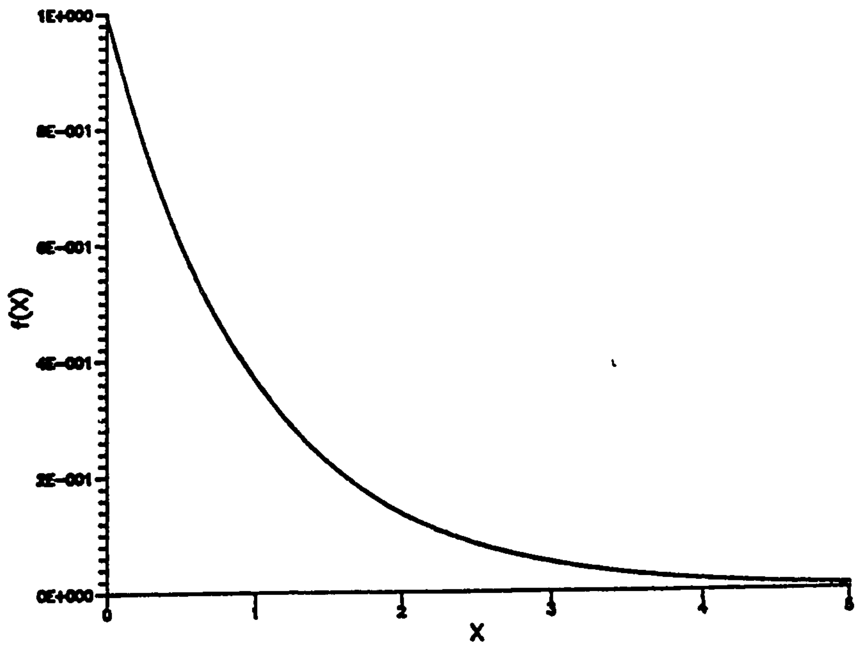

Figure 2.3 Exponential distribution

limit machine 1 need to be considered.

A model proposed by Finch and summarized in (4) had its basis in the continuous flow model. This characteristic of the flow should be noted here, i.e. whether it is in discrete packages, or a continuous flow of material (as in a mining operation). The majority of work in OR that has been done with respect to transfer lines is for the manufacturing industry and the flow is usually discrete, equally sized packages, be they engine blocks or computer chips. Extensions of this theory into continuous flow are not readily available. The model as proposed by Finch was interesting, but again limited because of exponetially distributed downtimes and uptimes. The failures are assumed also to be time dependent, i.e. deterioration occurs while the machine is not operating. This was in conflict with the previously stated assumptions of Chapter 1.

Other models (4) remove some of the assumptions, but not sufficiently to be implemented here. Downtime and uptime distributions are a vital characteristic of the system under study, and cannot be assumed for mathematical simplification. A modeller may make certain simplfying assumptions in the construction of his model, but must not alter those which are representative of the system.

2.3.1 Perturbation Analysis

The technique of perturbation analysis, developed by Ho et. al. (18,19) and Eyler (13), presents a very interesting opportunity for comparison with the simulation based technique. A brief background of the method is presented here, leaving the detailed exposition until its implementation in Chapter 5. It is hoped that this structure will keep the theory fresh in the reader's mind while the results are being presented.

This work, while again mainly concentrated in the area of manufacturing and thus discrete flow, has its basis in the continuous domain (14). The assumption was made that it would be possible to discretize the procedure to make it applicable to the manufacturing areas. As this procedure has its origins in continuous flow and is not restrictive of the types of failure and repair distributions, it is well suited to the mining industry.

Simply put, perturbation analysis (it is not restricted to bin sizing only) allows for the estimation of system partial derivatives within a single simulation run. Using "standard" methods to evaluate N variables would require N simulations, whereas only 1 is required using perturbation analysis (PA) (18). Ho gives a good introduction to the technique and background in Reference (19).

PA achieves its goal by perturbing (changing) by small amounts variables in the system. By monitoring the end effect of these changes, various statements about the system can be made. The perturbations may be observed while the simulator is running as usual; so, in addition to the usual point estimate, the modeller is provided with extra information about the system.

Perturbation techniques were selected as an addition to the simulation based results. There are many areas in which PA techniques may be applied in order to analyze system productivity and this thesis offered a chance to examine their applicability with respect to the mining industry.

3. Model and Experimentation

3.1 Model Overview

This chapter contains the simulation work done to produce the final analytic function. The system that is modelled is defined in detail and the program logic is presented. Later sections in the chapter contain the experimental design considerations, simulation results and function derivation. The development of the normalized function completes the chapter.

3.2 Detailed Model Description

3.2.1 Machine Parameters

As mentioned previously, the system under consideration is simple and balanced. Each of the machines involved is defined by the following 5 parameters:

1. Production Rate, PROD ($\frac{m^3}{h}$)
2. Mean Time to Failure, μ (h)
3. Mean time to repair, ν (h)
4. Standard deviation of $\mu = 0.20*\mu$
5. Standard deviation of $\nu = 0.15*\nu$

As the system is balanced, the above parameters suffice for both input and output machines. During the study, while the μ and ν values are changed, the coefficent of variation remains constant.

Productions rates are given in cubic metres per hour ($\frac{m^3}{h}$), and all operating parameters are in hours. Assuming that a system with the above parameters is operating without any external interference, the nominal production (NP) is:

$$NP = PROD \cdot \frac{\mu}{\mu + \nu} \left[\frac{m^3}{h}\right]$$

This is the maximum expected production rate of a machine in the system. As the system is balanced, the rates match (i.e. $NP_1 = NP_2$). This type of balancing is often achieved with machines of different availabilities by increasing the production rate of the more failure prone machine. In practice, production rate matching of two successive machines is preferable, since the larger machine will be forced down by the slower. This results in an under utilization of the capacity of the larger machine, and means some capital investment has been wasted. Assuming balanced machines also allows for simplification of the mathematics involved.

3.2.2 Bin Parameters

The bin in the model is assumed to have:

1. No operational failures.
2. No material flow restrictions.

As the bin may be anything from a steel structure to a stockpile on the ground, the failure free assumption is not a large departure from reality. The second assumption has been made to simplify the model, since material flow characteristics are not the subject matter here. This does not deny their importance when designing a bin. All volumes are "live", i.e. all material in the bin will flow, there is no "dead" or unmoving volume. When designing a bin, the amount of dead volume can be minimized through proper construction material selection, wall angle, etc., so therefore the second assumption is reasonable.

The bin is defined by the following parameters:

1. V_m the maximum volume (m^3)
2. C the control factor, $.5 < C > 1$.

When bin contents reach V_m machine 1 must stop and cannot be restarted until bin contents are below V_m*C. Similarly, when bin contents reach zero machine 2 must cease and cannot be restarted until bin contents reach $(1 - C)*V_m$.

3.2.3 Other Parameters and Considerations

The time intervals between failures are measured by operating time, not calendar time. If a machine is forced down, this period of time is ignored when scheduling the next failure. A machine is not expected to deteriorate when it is not working. The concept of scheduling failures is explained in a following section.

The duration of time the system is simulated is a user input and the values utilized and reasoning are explained in the Experimental Design section of this chapter.

While a machine is operating, it can produce only at exactly its maximum capacity. The model will not adjust the production rate of a machine in anticipation of an NI or NO state. That is, the production rate of a machine will not be decreased, so that the flow into or out of the bin is lessened and the bin full or empty event postponed.

3.3 Computer Model

The model was constructed using the SLAM II system (23). Simply put, SLAM II provides the bookkeeping functions, scheduling failures, end of failures, duration of failures, the random number generation, statistical distributions, and a level of reporting.

The model is stochastic with both discrete and continuous components. The discrete events are the machine failure and associated events. The continuous events are those initiated by the bin level. As seed values drive the random number generator, changing these seeds allows for independent samples of the outcomes. A listing of the program is contained in Appendix B.

Efficiency was a prime concern when the model was coded so that implementation on personal computers would be possible. The model has been run on an Amhdahl 5870 requiring 2 minutes of CPU (central processing unit) time. Running the same model on an IBM PC XT required two hours. In light of the rates charged for mainframe use, running simulations on PCs was a very viable alternative, if the programs were written with this in mind. To keep the computing costs at a minimum, the majority of the simulation work was done on the PC. While the two computers use different rounding methodologies, and have different instruction sets, it has been shown (9) that the two produce results that are, statistically, equivalent.

3.4 Experimental Design

Before beginning any experiment, be it actual laboratory work or simulation, the researcher must ask him/herself "what do I want out of this and how can I get the best results?". Asking this question at the start of a study will save much time and effort in the later stages. As the objective of this thesis is to study and explain the effect of additional storage between two successive machines, the model has been constructed with this in mind. The main output from the program is the mean production rate over the time period simulated. This type of simple model allows the researcher to focus on the problem at hand and not be needlessly confused with extraneous (for the time) details.

The actual machine and bin parameters utilized in the program are not the focal point. It is not the goal of this thesis to prove that the failure and repair parameters of the machines may be best modelled by certain type of distributions with parameters equal to x,y,... etc. The parameters selected are of "real-world" magnitude, yet allow for simple calculations with respect to availability and expected production. The machine parameters used are listed in Table 3.1.

Machine Parameter	Value
Production rate (m^3/hr)	1400.0
MTTF (hours)	5.0
Standard Deviation MTTF (hours)	1.0
MTTR (hours)	2.0
Standard Deviation MTTR (hours)	0.3

Table 3.1 Table of Parameters

3.4.1 Framing of Production

The production capacity of the system will be framed with respect to the minimum and maximum capacity that this system will generate, given the set of failure and repair distributions and production rate. The maximal capacity of the system will vary with bin volume as follows, assuming that there is no volume of storage between the two machines, then by equation 1.1 the fraction of time the system is operative is:

$$\pi_0 = \frac{1}{1 + \frac{2}{5} + \frac{2}{5}} = \frac{5}{9}$$

and the expected production rate of the system without a bin is:

$$PROD_{sys(0)} = 1400 \frac{m^3}{h} * \frac{5}{9} = 777.78 \frac{m^3}{h}$$

Now, assuming that there is an infinite volume bin between the two machines to effectively isolate their production histories from one another, the production becomes (cf p.12):

$$PROD_{sys(\infty)} = PROD \cdot \frac{\mu}{\mu + \nu}$$

$$= 1400 \cdot \frac{5}{5 + 2}$$

$$= 1000 \frac{m^3}{h}$$

This is the maximum expected rate over time that the system could be expected to achieve, if both machines were isolated from one another.

The value for the no bin case is calculated using a program written to simulate this type of case. The maximum production is verified using the base program with a bin size that is for all intents, infinite.

3.4.2 Definition of the Base Case

The set of parameters listed in Table 3.1 will be called the "base case" and from multiple simulation runs, an empirical equation will be derived relating system productivity to bin size. Each simulation will consist of twenty four (24) separate runs, each of an equal duration.

This duration will be a function of the failure time parameters and will be discussed shortly. The bin volumes simulated will be scaled according to μ.

After the "base case" has been run, the input failure parameters will be scaled downwards by a factor of ten, and the simulations repeated and an equation generated. Finally, the parameters will be scaled upwards (from the base case) by a factor of ten and the same procedure followed. The run time of each of these cases will be adjusted to ensure homoscedasticity (equality of variance) between runs. In order for valid comparisons to be made between cases, the variance among cases must be statistically equivalent. For the longer duration failure and repair times, the simulation run length must be increased to make an equivalent number of observations (as compared with the smaller parameters). Alterations of the failure time parameters by a factor of ten, results in similar adjustments of simulated times.

The plots resulting from these three sets of runs should, in theory, follow the same trend as in Figure 3.1. One would expect that for a given production rate, the system with the smaller failure and repair durations would require a smaller bin than would a system with longer duration outages. This is so because the bin to effectively provide storage or supply for the outages, would be of a smaller volume for the shorter duration outages.

Once this stage is complete, it will be illustrated how the initial function can be used to predict the system

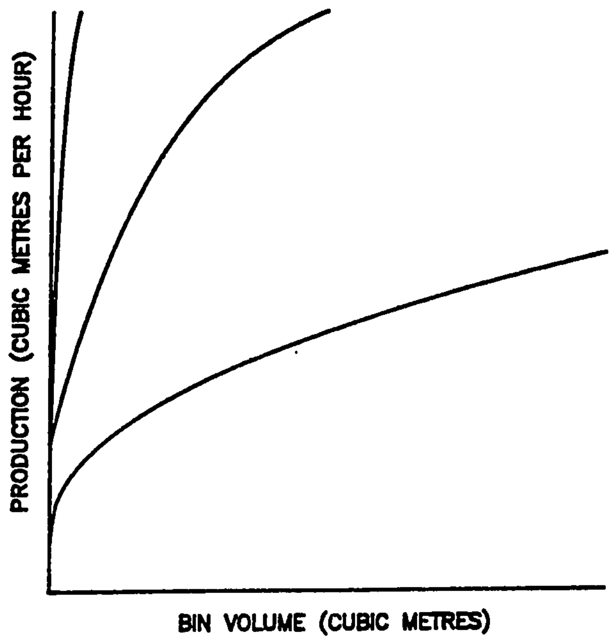

Figure 3.1 General trend of systems

productivity for the other cases by normalizing the bin volumes of these cases by a function of the failure parameters. This procedure will allow extension of the work done for one balanced system to another with different failure parameters.

3.4.3 Theoretical Distribution and Random Seeds

The model is capable of simulating outages with a variety of statistical distributions. As mentioned previously, the actual type of distribution will not be the crux of the matter. The log-normal distribution will be used for all cases. This distribution has been used previously to model machine failure parameters and it has been shown that there is no statistically significant difference in results when a different (triangular) distribution is used (9).

The log-normal distribution has been chosen for several reasons. As with most real-life distributions of failure and repair, the log-normal is skewed to the right. In practical terms this corresponds to the rare yet catastrophic failure that most operating machinery are prone to. A major concern with the exponential distribution is its lack of a definite shift from zero. The log-normal (Figure 3.2) distribution has the required shift.

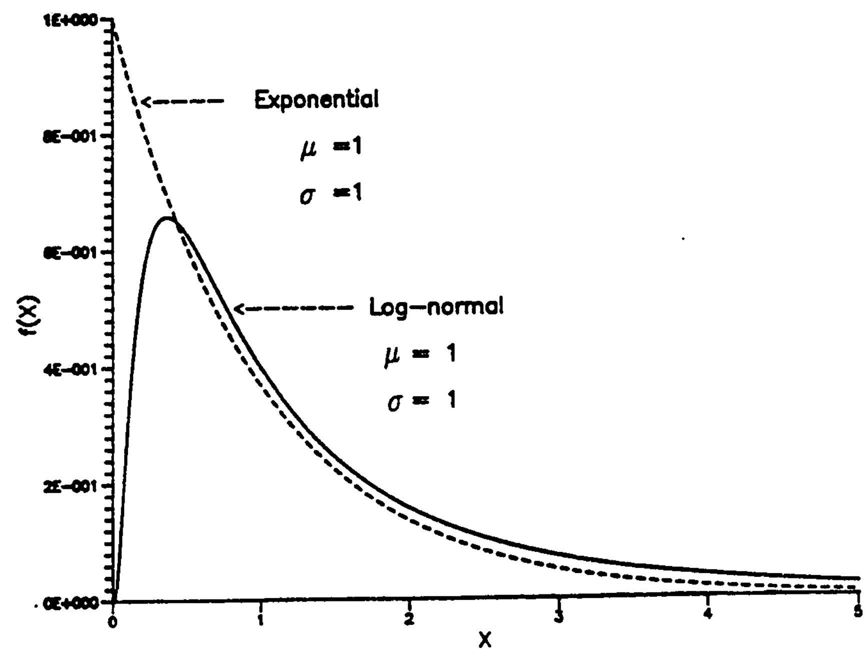

Figure 3.2 Log-normal Distribution

Additionally, the log-normal distribution is defined with two parameters, the exponential by only one. The fixed COV of unity of the exponential distribution is too large for most field data. The COV of the log-normal distribution is not fixed so it may be chosen to be more representative of actual operating data. In this work the COV was arbitrarily chosen at 0.2 for μ and 0.15 for ν.

In order to allow for more accurate comparisons between the different cases, each simulation run utilizes the same random number seed. The seeds are used in initialization of the generation of random variates for the machine failure and repair distributions. This similar starting point allows for higher resolution pairwise comparisons between cases rather than the usual comparisons between means.

3.4.4 Continuous Variables

The continuous variables are modelled by difference equations. A control level for starting or stopping the flow in or out of the bin is set in the initial SLAM statements. This control level is set at one percent of bin volume. Without this, a small bin could have a control level equivalent to its entire volume if it followed a run with a large bin. In order to meet this control level, the smallest time step that SLAM advances simulated time is also adjusted accordingly.

3.4.5 Program Logic

At simulation start up the bin volume is set at the fifty percent level, both machines are set operating and the first "down" events are randomly scheduled for each machine. The SLAM processor advances time to the first event, or by the maximum step size, whichever unit is the smallest. When the down event occurs, the completion of repair is scheduled, and the simulation advances in time. If during this time the bin becomes full or empties (depending on which machine has failed), the other machine is forced down. Since this machine is no longer operating, its future failure event is taken out of the future events file and placed on hold. When the other machine becomes available and has replenished/diminished the bin volume to an acceptable level, the forced down machine is restarted and its next failure is delayed by an amount equal to the time it spent in the delay file. It is through this delay file mechanism that the machines only deteriorate during operating time. When the simulated time equals the set duration, the simulation is halted and an output report printed. SLAM provides an output report on availability statistics of the machines and the current, maximum and minimum volumes of the bin. There is an added custom report which contains the bin volume and the mean production rate for the simulation. It is this data that is used as input for the equation derivations. A sample output file is provided in Appendix B.

3.5 Selection of Regression Equation

In the selection of the regression equation, a priori knowledge of the general behavior of the system is used, along with more rigorous criteria. Intuitively, one would expect the system to behave as depicted in Figure 3.1, with production (PROD) being a function of the bin volume (BIN).

At low bin volumes larger gains in production should be realized for a given increase in volume than at the larger volumes. The large bin volumes have sufficient volume/material available to effectively buffer the majority of machine outages. Any increase beyond this volume may be wasted expenditure as the utilization of this excess capacity is very low. It is at the "optimum" bin volume that the simple system is approaching its maximum capacity; the rate of the weakest member. The production histories have been effectively isolated at this point.

The first derivative, $\frac{\partial PROD}{\partial BIN}$, of this function should indicate the above mentioned trend. The slope of the function should be greatest at the lower bin volumes, gradually decreasing and eventually approaching zero as the bin volume increases. In more specific terms, the model was chosen using the following criteria:

1. Most parsimonious
   a. the method that would provide the best fit with the least number of parameters.
2. Total Error Term
   a. minimum departure of predictions from simulated realizations.
3. Distribution of Residuals
   a. the residuals must be unbiased, $\Sigma resid = 0$
   b. the residuals are statistically tractable; that is,they approach a

normal distribution with expectation 0, and variance σ^2, i.e. $(N(0,\sigma^2))$

4. Stability of Regression
   a. various subsets would be examined and the resulting equations evaluated against that done using the entire set.
   b. the parameters of the regression would have to be significant, i.e. not include zero in a 95% confidence interval.

The above points, will be expanded upon later where it is necessary.

3.5.1 Methodology

The data as output from the program were deficient in one area in that it did not contain values for production when the bin volume was zero, i.e. no bin present. The base simulation model was not constructed to consider such a case. Therefore, using the theory explained previously, the production for a zero bin volume was calculated, i.e. with Bin = 0;

$$PROD_{sys} = 777.78 \ \frac{m^3}{h} = PROD_{sys(0)}$$

This system is simulated with a program specifically written to model a system without a bin (see Appendix B for the code). The model used for all other cases assumes some bin volume between the two machines and would not have given accurate values for the expected production. The mean value for production is then found to be:

$$PROD_{sys}=781.10\ \frac{m^3}{h} = PROD_{sys(0)}$$

This value substantiates the logic and code to be used for the simulations that follow. In order to maintain consistency with the previous methods, the simulated value is used. In practice, this step could be omitted and the analytical value used for the y intercept. Inserting the analytical value would have biased the regression equation and this was not desired. As all other bin volumes consisted of independent realizations, inserting one constant value at any point would result in a zero variance at that point. This zero variance would have an effect on the final error term and the parameters produced by the regression.

Following the above methodology and experimental design, the three cases were simulated. The cases have been labelled by μ, i.e., 5.0 for the base case, 50.0 for the upward scaling and 0.5 for the downward scaling. Plots of the output are shown in Figures 3.3 - 3.5.

3.5.2 Types of Regression Models

Initially, two types of regression models were attempted:

1. Stepwise Linear Regression (10)
2. Non-Linear Regression (24)

The modelling was done using the PC (1988) version of BMDP (8).

In all cases there was no transformation applied to the dependent variable. Non-linear transformations of the

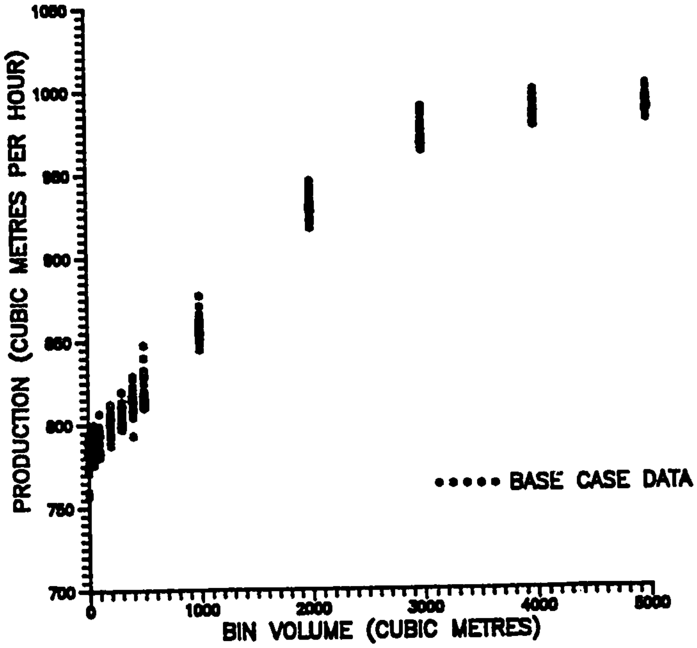

Figure 3.3 Simulation output for 5.0 case

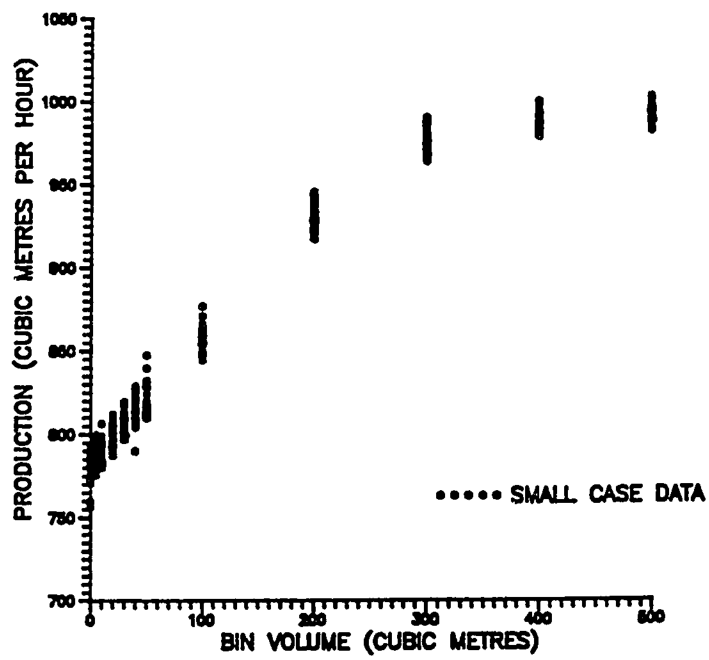

Figure 3.4 Simulation output for 0.50 case

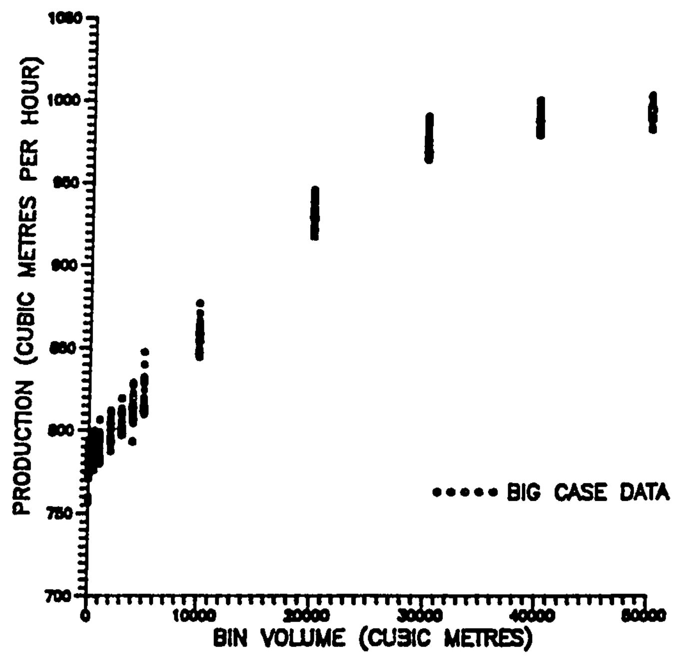

Figure 3.5   Simulation output for 50.0 case

dependent variable would have complicated the statistics of the regression.

The parameters for stepwise regression were all linear transformations of the bin volume:

1. BIN
2. BIN^2
3. BIN^3
4. $BIN^{1/2}$
5. $BIN^{1/3}$

As stepwise regression enters the parameters into the equation in the order of statistical importance (the greatest potential reduction in the sum of squares of the residuals), it would indicate which of the above paramters were the most significant.

The non-linear model was a three (3) parameter exponential:

$$PROD_{sys} = B0*e^{B1*BIN} + B2$$

BMDP program 2R was used for the stepwise regression and program AR for the non-linear regression.

3.5.3 Selection of Equation and Validation

Using stepwise regression, and the simulation output (with values for a bin volume of zero included), the following equation was generated:

$$PROD_{sys} = 786.27+8.654*BIN^{1/2} - 19.11*BIN^{1/3} - 6.313E-10*BIN^{3}$$

The parameters are shown in the order they entered into the equation, illustrating their relative significance. Other results of the regression were:

MSE (residual) = 88.36

R^2 = 0.9868

Std. Error of Estimate = 9.3999

A partial BMDP output is given in Appendix C. With respect to the initial goals listed previously, the first two are satisfied (parsimonious and low total error). Plots of the residual (R) versus the predicted, and R^2 versus predicted are shown in Figures 3.6 and 3.7. These plots

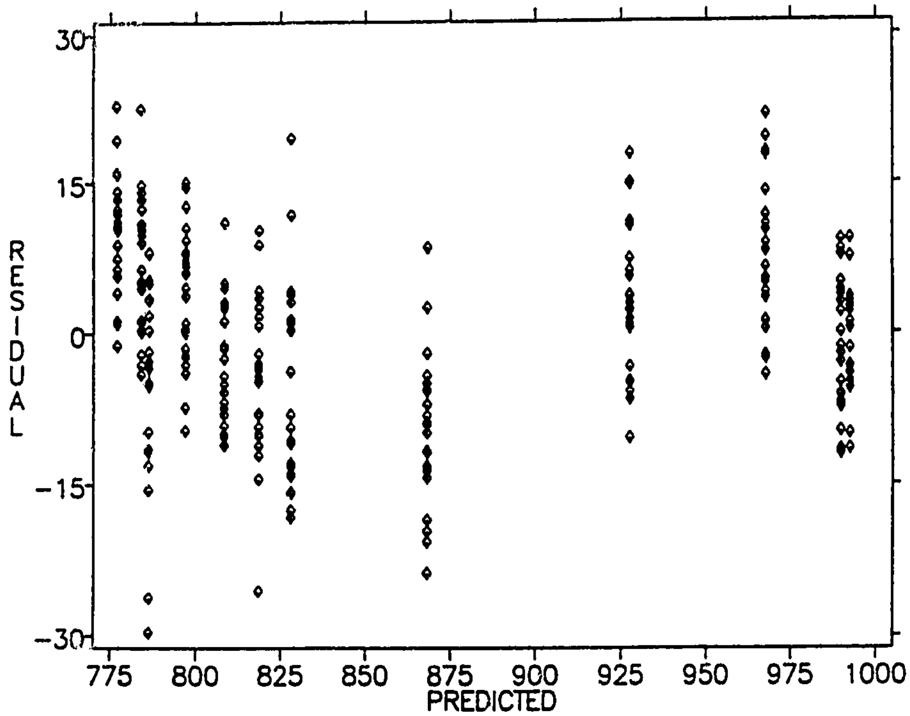

Figure 3.6 Resid. vs. predicted - stepwise model

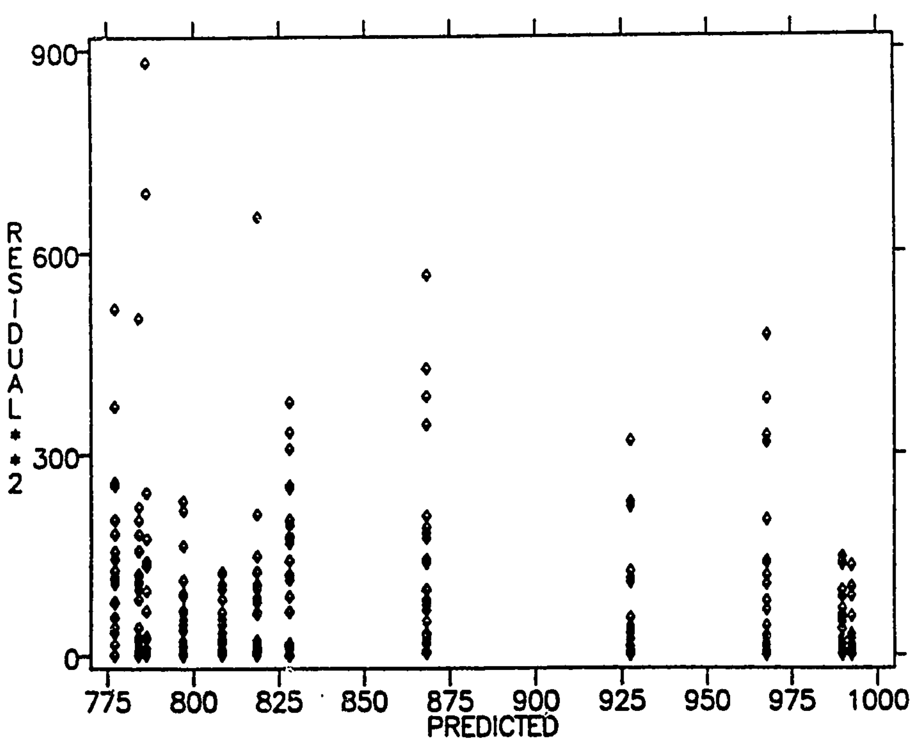

Figure 3.7 R² vs. predicted - stepwise model

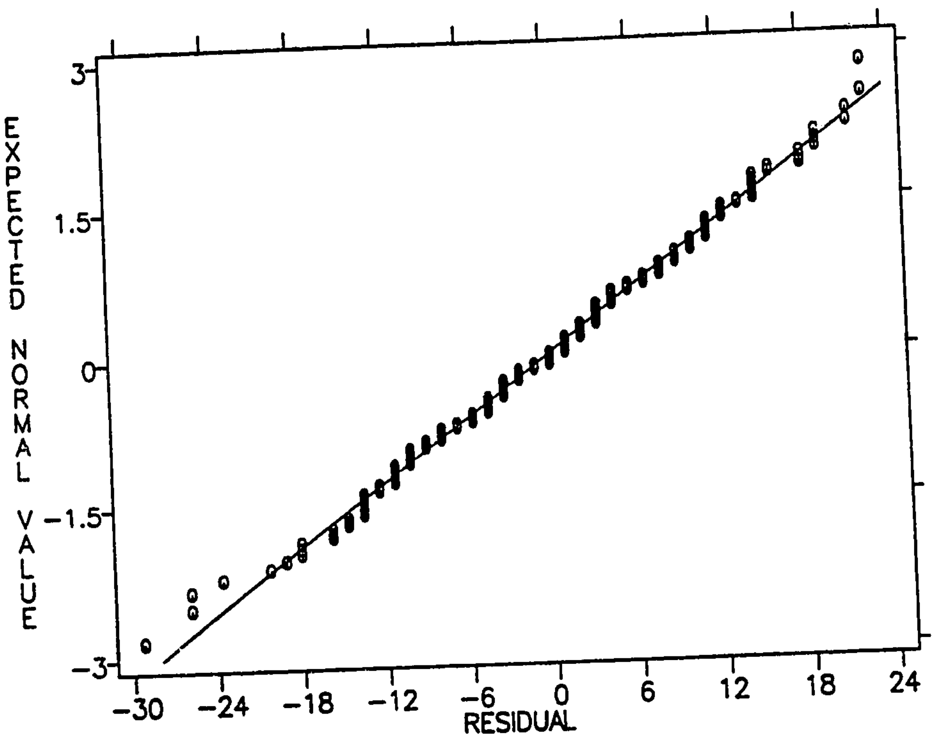

Figure 3.8 Normal probability plot - stepwise model

indicate a trend in the residual, with respect to the predicted values, and a non-equality of variance. These discrepancies are expanded upon later. A normal probability plot of the residuals (Figure 3.8) supports an assumption of normally distributed residuals.

The non-linear regression produced the following equation:

$$PROD_{sys}=-249.78*e^{-0.000455*BIN} + 1026.66$$

Estimated MSE = 92.73

Psuedo R^2 = 0.9861

A partial BMDP output is given in Appendix C. Plots of the residual indicate a similar trend as exhibited by the stepwise regression, see figures 3.9 and 3.10.

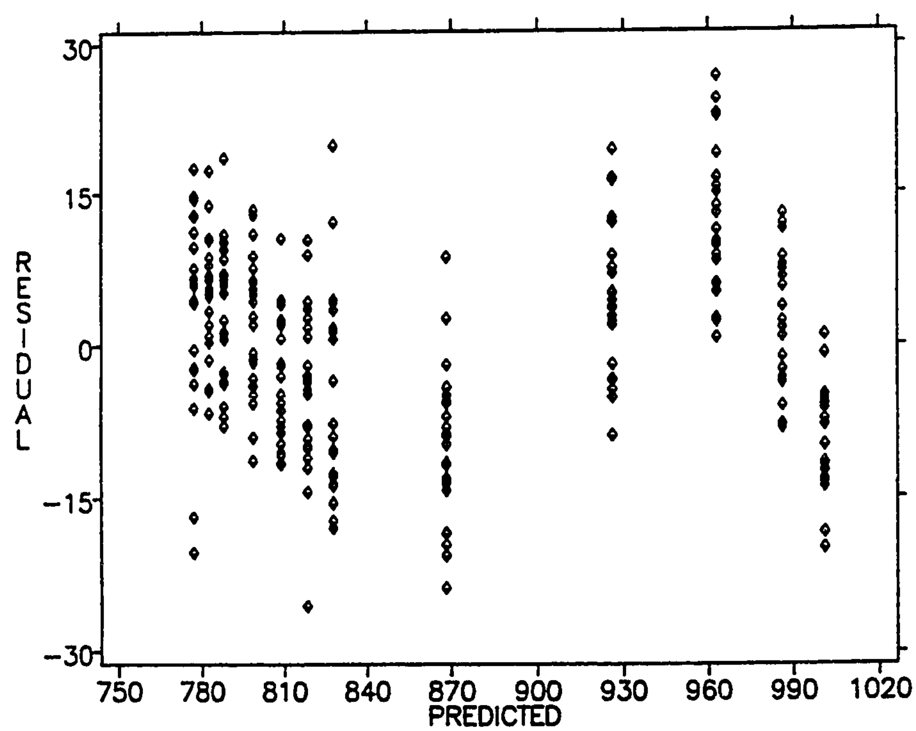

Figure 3.9 Resid. vs. predicted - exponential model

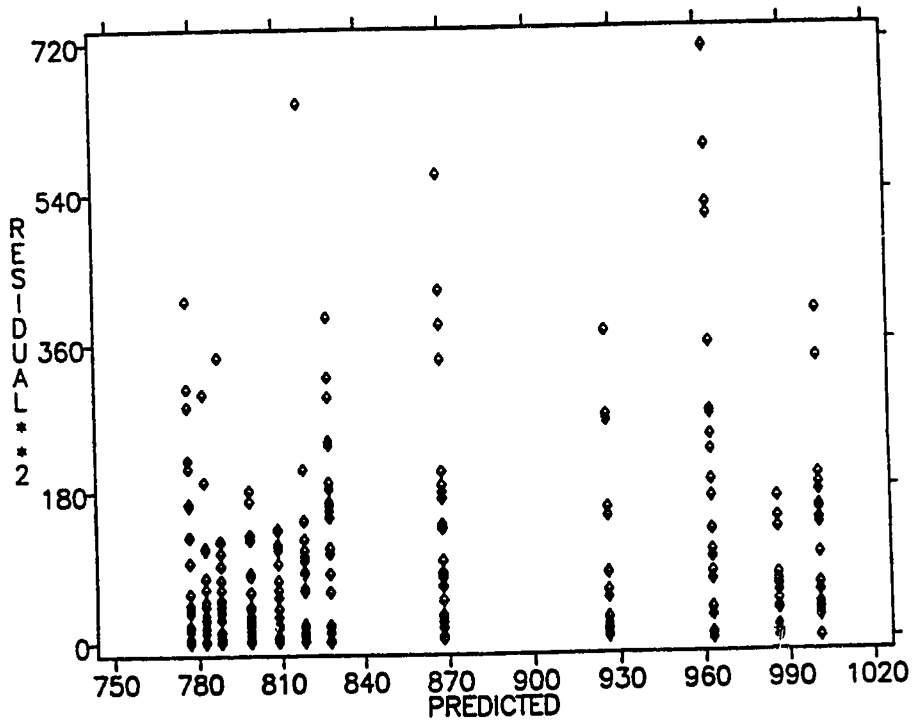

Figure 3.10 R² vs. predicted - exponential model

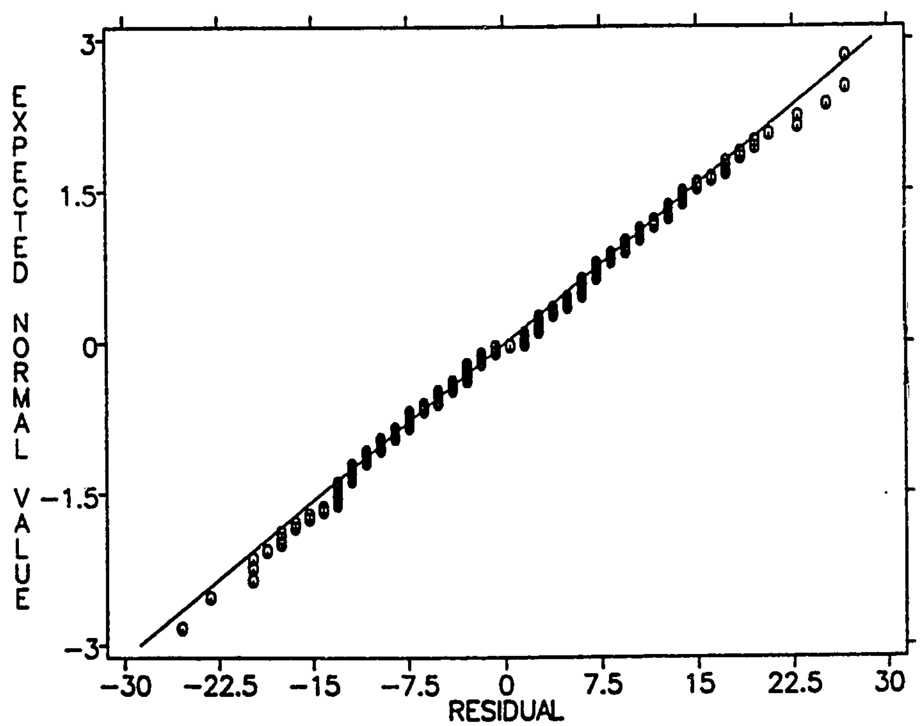

Figure 3.11 Normal probability plot - exponential model

The normal probability plot, Figure 3.11, indicates normality of the residuals.

At this point, further investigation was made into the behaviour of the model, particularly with respect to the residuals. Utilizing the parameters selected by stepwise regression and the AR package, the correlation of the parameters was examined. A listing of the BMDP control code is given in Appendix C. The parameters remained similar to those found by the stepwise package, (Table 3.2), but additional information about the correlation between the parameters was found, see Table 3.3.

Parameter	Coefficient Estimate	Asymptotic Standard Deviation
B0	786.3	1.833
B1	8.653	0.2197
B2	19.11	0.8469
B3	6.313	0.3315

Table 3.2 Stepwise regression parameters

	B0	B1	B2	B3
B0	1.0000			
B1	0.7006	1.0000		
B2	0.8101	0.9806	1.0000	
B3	0.2601	0.7299	0.6155	1.0000

Note: B0 - Coefficient of Intercept
B1 - Coefficient of $Bin^{1/2}$
B2 - Coefficient of $Bin^{1/3}$
B3 - Coefficient of Bin^3

Table 3.3 Correlation of stepwise parameters

The correlation between B1 and B2 is high, but could be expected because of their similar functional shapes. This procedure then validated the use of the AR program to study further the results from the stepwise regression and compare them with the exponential model. The residual versus predicted, residual squared versus predicted and normal probability plots are presented as Figures 3.12 to 3.14.

As one criteria for choosing a model was significant parameters, a 95 % confidence interval for the parameter estimates was found for both the stepwise and exponential models. For example,

For the stepwise;

Estimate = 8.654

σ = 0.21971

A 95 % confidence interval is then;

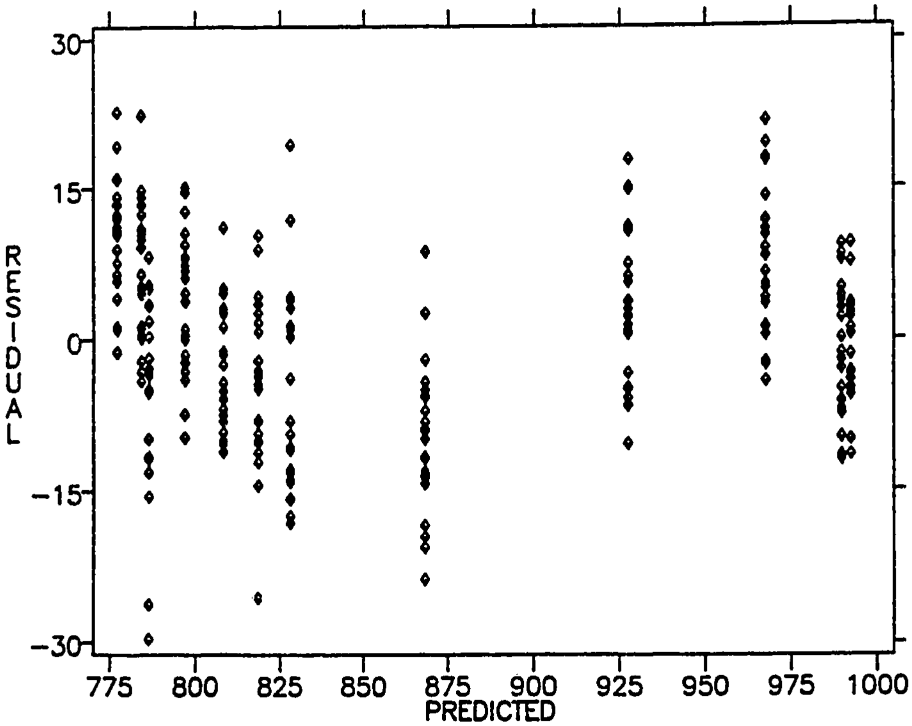

Figure 3.12 Resid. vs. predicted - AR model

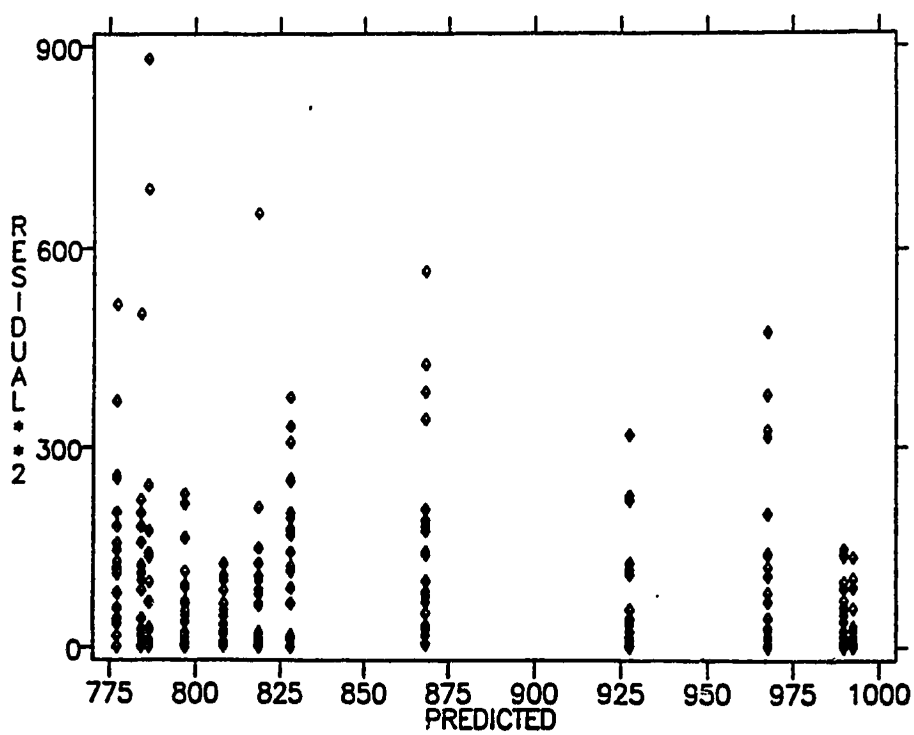

Figure 3.13 R^2 vs. predicted - AR model

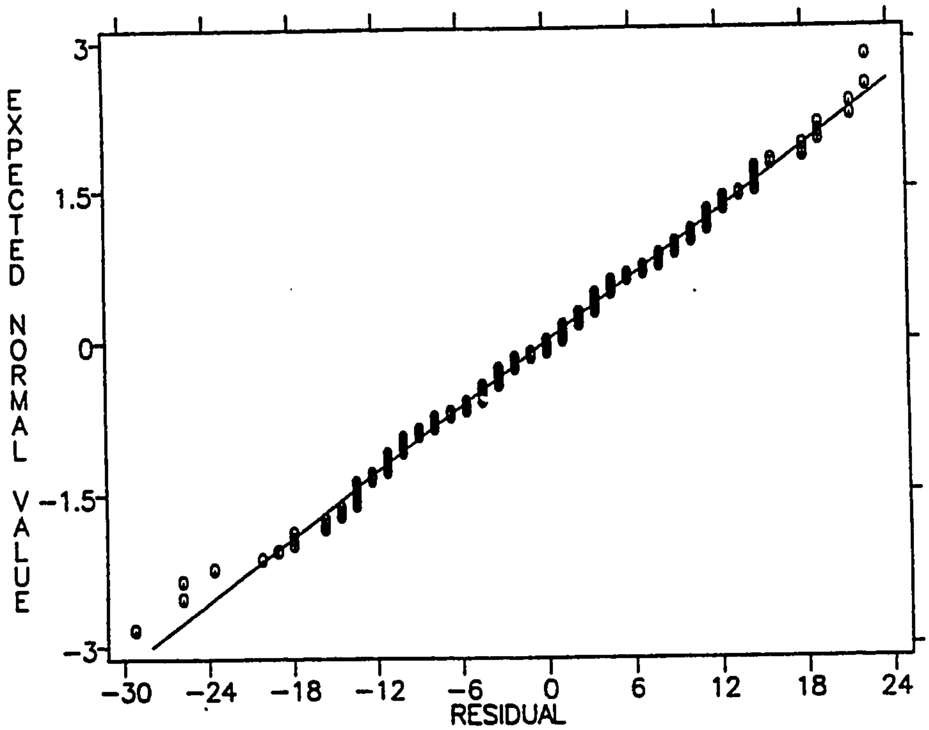

Figure 3.14 Normal probability plot - AR model

8.654 ± 1.96*0.21971

= 8.653 ± 0.4306

Notice that zero is not in the interval, indicating a significant parameter.

The other paramters;

B0: 786.27 ± 3.593

B1: 19.11 ± 1.66

B3: 6.313 ± 0.650

The confidence intervals for the exponential are;

B0: -249.78 ± 6.39

B1: -0.000455 ± 0.0000314

B3: 1026.66 ± 7.00

Both the exponential and stepwise models had met an equal number of the criteria listed; however, the exponential model was selected as it had one less parameter. To further test the stability, subsets comprising fifty and eighty percent of the data were randomly selected and the regression re-run. Output from these runs appears in Appendix C and no new or significantly different parameters arise. The regression equations from these runs were:

50%: $-250.21 * e^{-0.000452 * BIN} + 1026.51$

80%: $-249.80 * e^{-0.000452 * BIN} + 1026.58$

All of the above coefficients fit within the 95% confidence intervals described above.

Since the regression was run at selected bin values, non-standard bin volumes were also tested and compared to the results from the regression equation. The bin volume was modelled as a uniform random variable between 50 and 5000 m^3. The results are illustrated graphically in Figure 3.15.

The independence of the residuals and equality of variance was a concern that surfaced at each regression attempt. It is beyond the scope of this thesis to formulate explanations and solutions to this problem. This is an important concern and requires more research. This is discussed further in recommendations for further research.

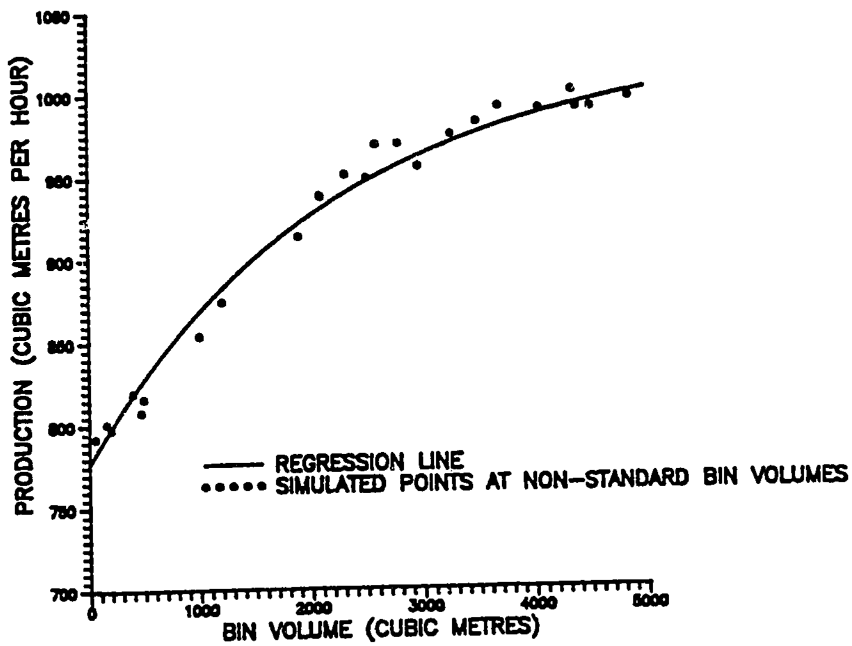

Figure 3.15 Randomly sampled bin volumes

Notwithstanding the above point, the method of regression had been proven to be sufficiently robust, and was applied to the other simulation cases:

For $\mu = 0.5$, $\nu = 0.2$;

$$PROD_{sys} = -249.76 * e^{-0.004538*BIN} + 1026.74$$

For $\mu = 50.0$, $\nu = 20.0$;

$$PROD_{sys} = -249.76 * e^{-0.000045*BIN} + 1026.65$$

Plots of the three regression equations, with the simulation output superimposed, are shown as Figures 3.16 to 3.18. These equations will be analyzed to show that they may be normalized and used on other simple, balanced systems with different failure and repair time parameters and production rates.

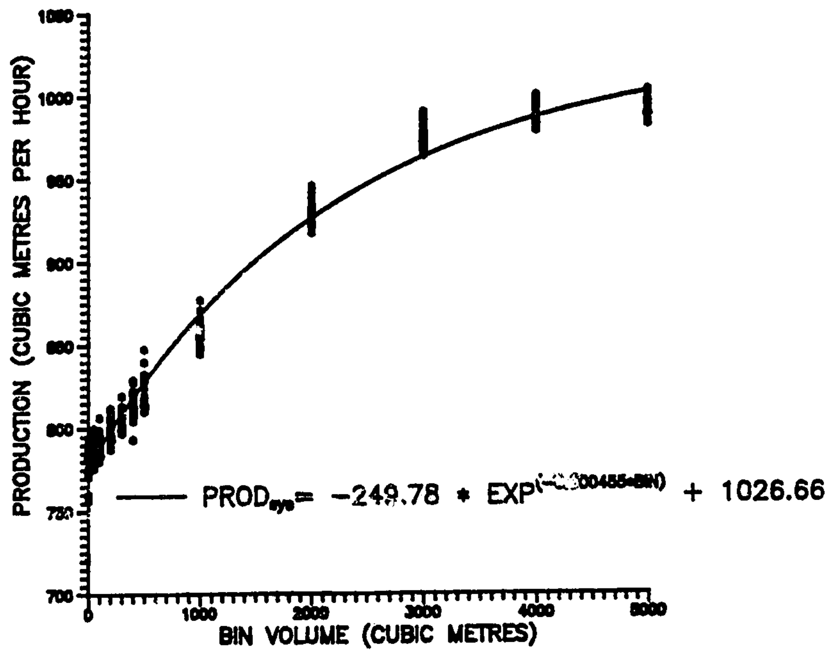

Figure 3.16 Regression line with simulated points, 5.0 case

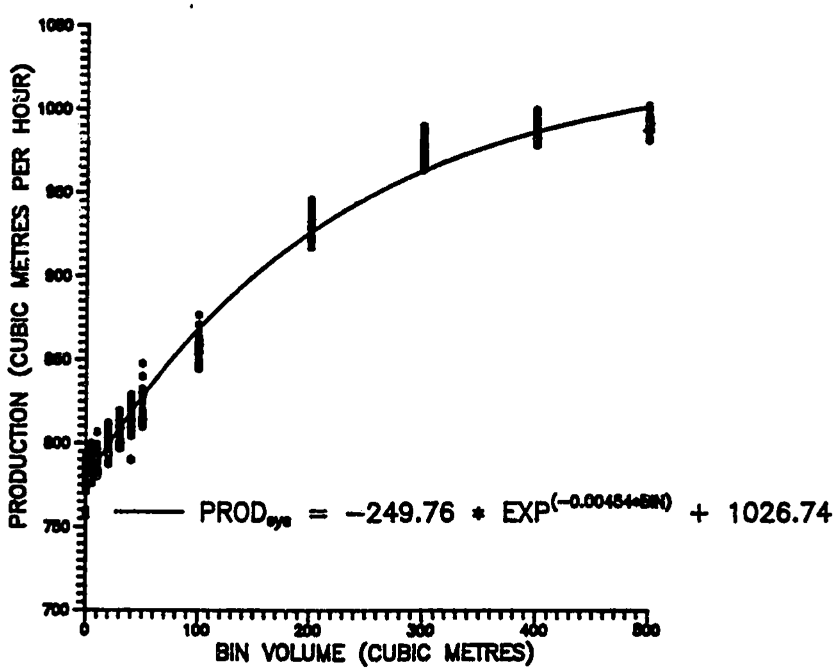

Figure 3.17 Regression line with simulated points, 0.50 case

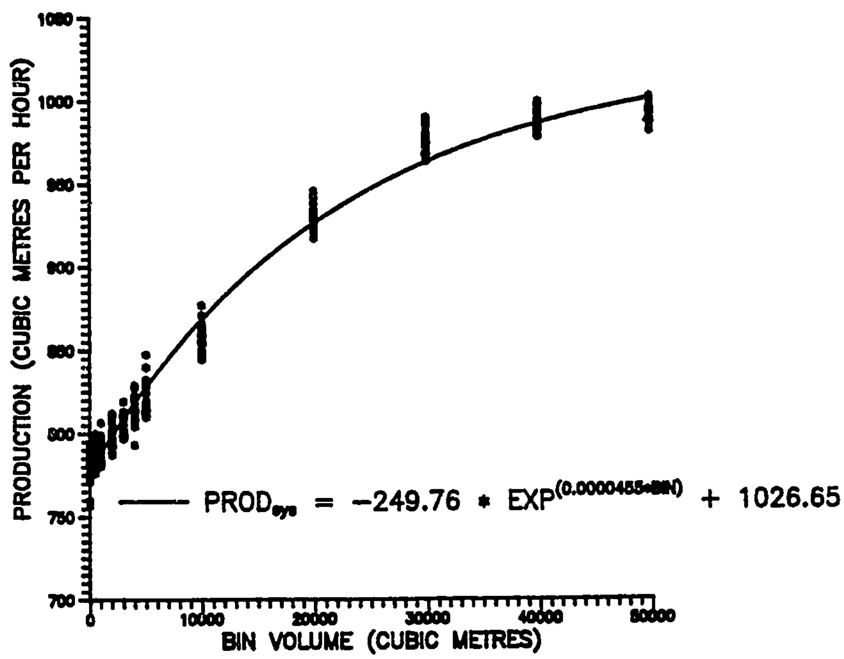

Figure 3.18 Regression line with simulated points, 50.0 case

3.6 Normalization of Equations

It may be easily seen that the three derived equations given with Figures 3.16 to 3.18 are not significantly different except for the B2 parameter and that this parameter is related by the scale of the failure parameters. For example, there is a factor of ten difference between the 5.0 and 50.0 case. This factor is reflected in the B1 parameter: 0.000455 for the 5.0 case and 0.000045 for the 50.0 case. To apply one equation to another case, one would need only to multiply B1 by the ratio of the failure parameters. In every case, the resultant equation would contain parameters within the 95% confidence intervals for that case. It would seem reasonable to assume that this scaling could be applied to other, different situations if a normalization of the equation was to occur.

First, the production rate was scaled by dividing by the maximum expected rate, 1000 $\frac{m^3}{h}$ and to avoid altering the equation, the bin volume was divided by this amount as well. The result is that the ordinate, production, was now dimensionless, and the abscissa, (originally bin volume), was now in terms of hours (with respect to the maximum expected production). A plot of the three cases is shown in Figure 3.19. There is no change to the structure of the plot, only a scaling has been done.

In order to "pull" the regression lines into one, it had to be decided what parameter should be used. Obviously, whatever the parameter, it must be with respect to time so that the abscissa would also be dimensionless. In effect, the line had already been scaled by both system parameters, μ and ν, when the values were divided through by the maximum expected production rate. It was decided that since μ of the system was of the larger magnitude, and that on average, either machine would feed into, or draw out of the bin for no longer than this duration, this would be the scaling factor.

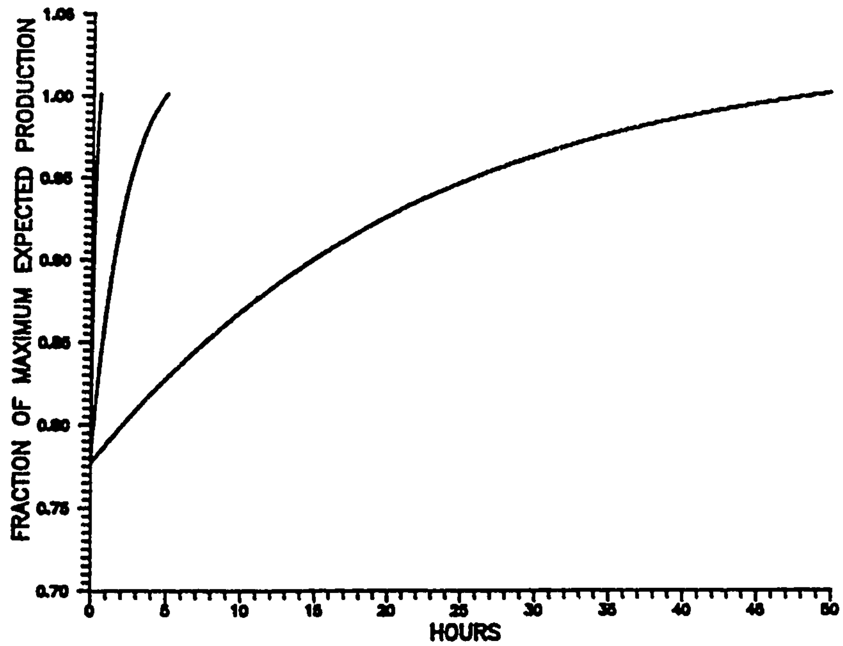

Figure 3.19 Scaled regression lines

By dividing the bin parameter (at this time in units of hours) by the μ parameter, all the functions plotted as shown in Figure 3.20. Clearly, each case is represented by a very similar function. These functions are:

For the 5.0 case;

$$PROD_{scl} = -0.2498 * e^{-2.273*BINSCL} + 1.027$$

For the 50.0 case;

$$PROD_{scl} = -0.2498 * e^{-2.273*BINSCL} + 1.027$$

For the 0.5 case;

$$PROD_{scl} = -0.2498 * e^{-2.269*BINSCL} + 1.027$$

Note: where $BINSCL = \dfrac{BIN}{PROD_{sys(\infty)} \cdot \mu}$

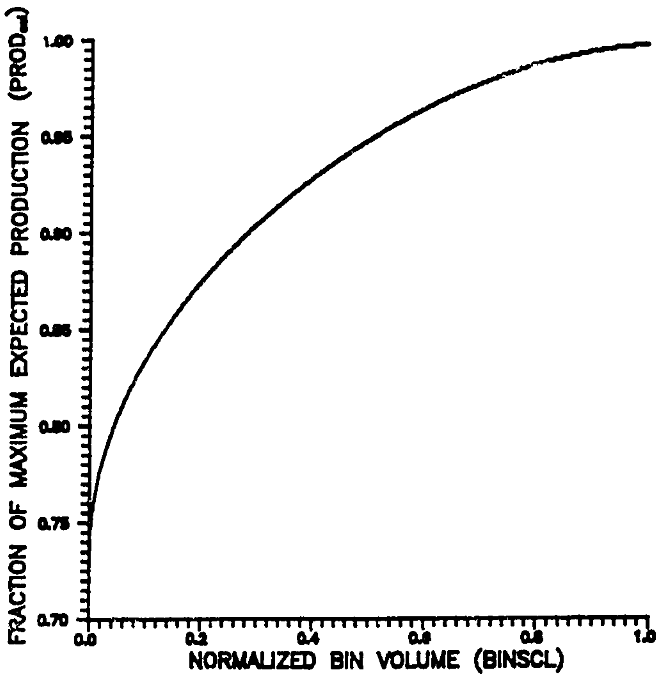

Figure 3.20 Normalized regression function

To convert the scaled value of production rate back to $\frac{m^3}{h}$, the result is multiplied by $PROD_{sys(\infty)}$, (as this value was not divided by μ). The form of the equation is intuitively correct since;

$$\lim_{BINSCL \to \infty} PROD_{scl} = 1.027 \approx 1.0$$

and since;

$$(1.0) * PROD_{sys(\infty)} = PROD_{sys(\infty)}$$

and for BINSCL = 0 (and hence BIN = 0);

$$PROD_{scl} = -0.2498 + 1.027 = 0.7772$$

where;

$$PROD_{sys} = 0.7772 * 1000 = 777.2 = PROD_{sys(0)}$$

The above value, calculated from the derived formula, is not significantly different than that calculated using Equation

1.1.

The power of this equation is that it can be used to predict the increase in productivity with respect to an increase in bin volume, regardless of the magnitude of μ and ν. Note that this particular equation was developed with a certain minimum and maximum expected production. These values would have to be altered to suit the situation, but as these may be calcuated directly and do not require extensive simulation, this is not a difficult task.

As there has been shown to exist a similarity between the three cases when all have been scaled, the scaled data can be used collectively to produce a regression equation. The regression line thus obtained is:

$$PROD_{scl} = -0.2498 * e^{-2.27*BINSCL} + 1.027 \text{ ....} (3.1)$$

This regression line will be used in the following chapters for both the cost analysis and comparison against the operations research technique.

3.7 Uses and Applicability of the Function

As the mining industry is diverse and operations must conform to their available equipment and conditions, the likelihood of a system exactly matching the constraints that have been employed is small, so it must be stressed that the function that as been developed is only applicable under the previously stated conditions which are:

1. Input Rate = Output Rate
2. Identical Failure Distributions for Input and

Output

3. Identical Repair Distributions for Input and Output

4. Constant COV (=0.20 for input, 0.15 for output)

Assumption (1) is the least restrictive of the four and it may not hold true in many operations. While it is possible that the failure and repair distributions types may be equivalent, i.e. lognormal, the parameters of the distributions are not likely to be similar. This unbalances the system, i.e., the expected rates of the input and output systems will differ. In order to rebalance the system, the rates of the input/output systems must be adjusted (which would violate assumption (1)).

The assumption of a fixed coefficient of variation is also unlikely in an operating mine environment.

These four restrictions must be relaxed before any detailed work may be done with the normalized function. The function does have application in the initial design phases of a project. In these stages, the information available is likely quite general, and the function may be applied, keeping in mind the accuracy of the input. Using the "order of magnitude" information that is usually available in the initial stages of a project to generate data for a detailed model is not only a waste of resources, but an obfuscation of facts.

Further research that is directed at removing these four constraints is required so that a similar, normalized function can be developed that could be applied more specifically.

The distributions of the residuals is an area that also requires more study. It is apparent from the plots of the regression equations that a more detailed study as to the causes of the non-equality of variance and the biased nature of the residuals is required. The areas to concentrate in in such a study would be the experimental design and run lengths at each of the bin volumes. It is possible that due to the shape of the curve and the effect of the bin size on the system, unequal run lengths may be required in order to stabilize the residuals.

4. Application of the Model

4.1 Outline of Chapter

This chapter contains three applications of the model defined in Chapter 3. The first is a comparison against the analytical work of Elbrond (11). Next, the model is used to predict the productivity of other simple, balanced systems. The estimates provided by the model are compared against simulated results of the system under consideration. Finally, the model is used in a sample economic analysis of the installation of a bin.

4.2 Comparison with Analytical Techniques

As mentioned, the analytically derived equations of Elbrond (11) have been programmed and the calculated results are plotted in Figures 4.1 to 4.3. The regression line obtained in Chapter 3, Equation 3.1, is also plotted on these graphs. Due to a lack of real knowledge of Elbrond's system and procedures, no more rigorous, statistical comparisons could be made. The graphs do however indicate that with respect to the simulation generated equation, Elbrond's estimates are very conservative. There is also the case of the different intercepts. It is felt that the case where the machines do not deteriorate while forced down is more representative of real world conditions. The product rule formula, as applied by Elbrond, produces a lower value for the intercept, consistent with his overall lower

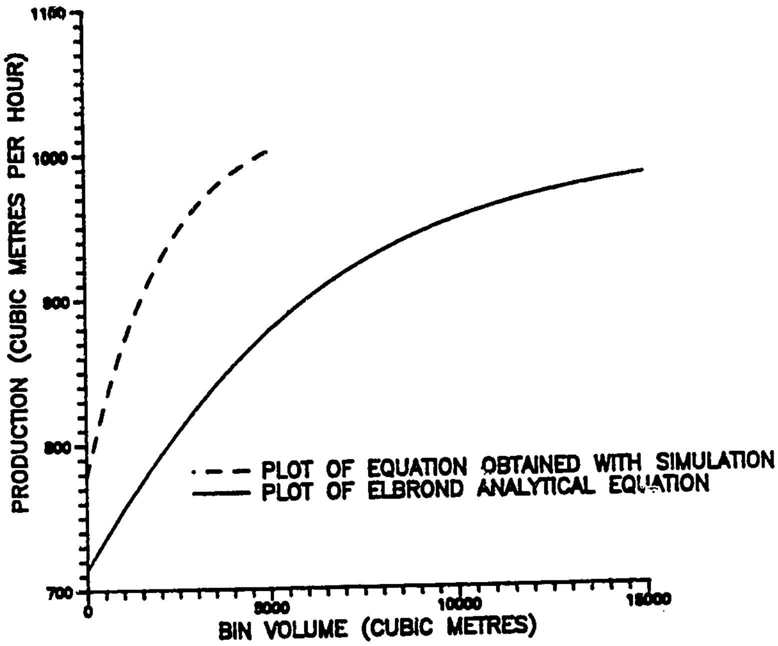

Figure 4.1 Elbrond function and 5.0 case

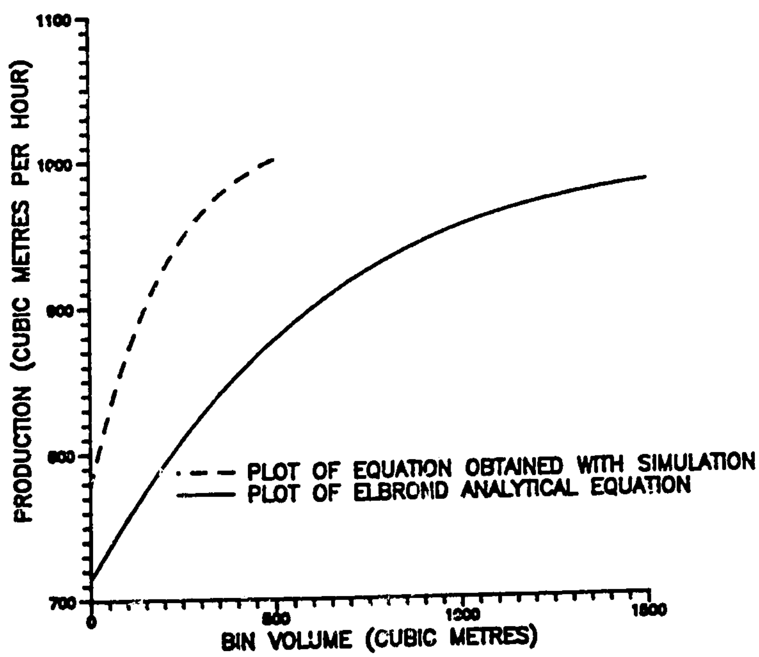

Figure 4.2 Elbrond function and 0.5 case

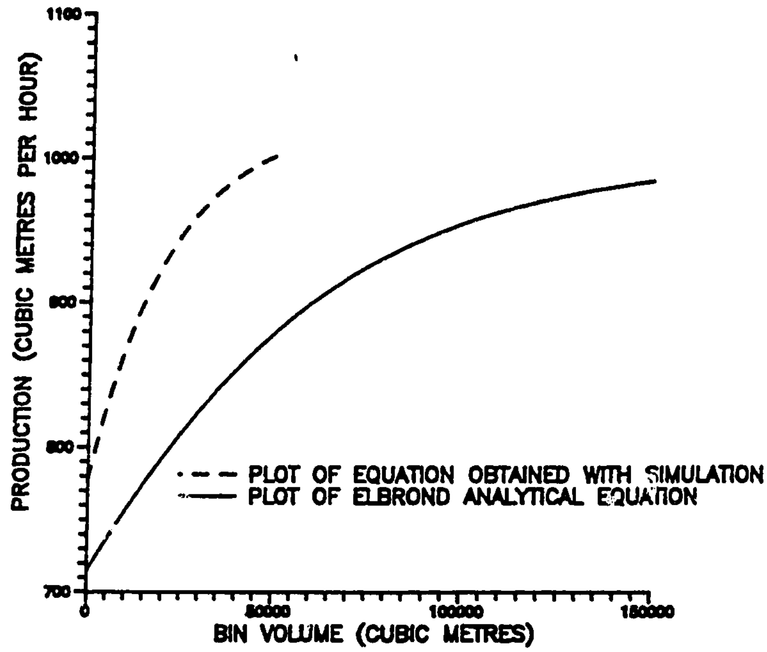

Figure 4.3 Elbrond function and 50.0 case

estimates of productivity for given bin volumes.

4.3 Predicting Production of Other Balanced Systems

The regression model was used to predict the production of several, different systems. All systems remained balanced, and the COV was the same as for the previous work. The MTTF ,MTTR and production rate were selected at random. A list of the selected simulation parameters is shown in Table 4.1. Using Equation 3.1, the production rate was

Case	PROD (m^3/hr)	MTTF (hrs)	MTTR (hrs)	BIN (m3)
1	2500.0	10.0	6.0	5000.0
2	5000.0	10.0	2.0	1500.0
3	800.0	5.0	4.0	500.0
4	2340.0	9.0	3.6	6000.0
5	2340.0	18.0	3.6	1200.0

Table 4.1 Selected test case parameters

estimated for each system at various bin volumes. The basic equation as developed previously, can be seen to contain error in the parameters, as would be expected for an experimentally derived equation. With sufficient sampling the B2 parameter should approach unity, so that at the large bin volumes, the normalized productivity is equal to unity. By multiplying through by the maximum expected production, the theoretical limit is found. The equation then takes the form;

$$PROD_{scl} = -B0*e^{-2.27*BINSCL} + 1.0 \quad \ldots\ldots\ldots\ldots\ldots (4.1)$$

where:

$$BINSCL = \frac{BIN}{PROD_{sys(\infty)} \cdot \mu}$$

Formula 4.1 is then the final equation to be extended to the other systems. The only unknown quantity is B0 and this may be found from Equation 4.1. Given BIN = 0, then BINSCL = 0 and Equation 4.1 becomes;

$$PROD_{scl} = B0 + 1.0$$

Using algebra, (see Appendix A), B0 can be reduced to:

$$B0 = \frac{\nu_1}{(\mu_1 + 2 \cdot \nu_1)}$$

Now all three parameters of the equation are known and may be easily calculated. Once the normalized production has been found for the bin volume in question, multiplying by $PROD_{sys(-)}$ gives the estimate in $\frac{m^3}{h}$.

To verify these values, the simulation model used previously was run and the estimates recorded. Each verification run contained 24 independent estimates of the expected production rate, from which the standard deviation was calculated. Table 4.2 contains the mean value from the simulation runs and the estimate from the model. The functional values approximate very closely that of the simulated cases. In all of the cases except (4) the function value is within a 95% confidence interval. Even with the small sample size, the function appears to be unbiased, with a mean percentage error of 0.80.

Case	Simulated Mean (m^3/hr)	Standard Deviation	Function Value (m^3/hr)	%Error
1	1314.6	51.2	1356.2	3.2
2	3706.1	63.7	3617.6	-2.4
3	345.5	16.4	362.0	4.8
4	1572.0	14.8	1521.4	-3.2
5	1719.6	16.4	1692.2	-1.6

Table 4.2 Comparison of function and simulation

4.4 Economic Justification of a Bin

This section is not intended to be a rigorous, economic proof of the benefits of inserting (or increasing) the bin volume between two serially connected machines. The assumptions made concerning bin construction costs, and commodity values limit the exercise to a simple demonstration of the use of the function.

The base system, defined in the Chapter 3, (cf Table 3.1) will be used. It will be assumed that the system is presently operating without a bin and a payback period for each bin volume will be calculated. The material will assumed to be oil sand, yielding one barrel of synthetic crude oil per cubic metre mined. Profit per barrel of synthetic crude has been estimated at four dollars per barrel during normal operation. As when the bin is operating due to machine failure the cost of moving the material will increase (due to dozing of "dead" volume etc.) the incremental profit has been decreased to $3.25 per barrel.

Any costs will be highly dependent upon the location of the mine, the type of material and the production requirements of the plant, among other conditions. Since only an outline of the use of the function is to be given, a cost per cubic metre of bin volume was assumed. This cost was arrived at after discussions with others in the mineral industry and was taken to be $3500 per m^3 of capacity. This cost has been assumed to cover all construction costs for the bin structure as well as all associated feeding and transport equipment. Obviously a straight line cost function is simplistic, but it will suffice for this simplistic example.

The actual analysis will be limited to finding the time to repay the cost of the bin. Any more detailed measurement, such as return on investment, would require a knowledge of the tax structure of the company involved. As this situation would be vastly different for different companies, and changing within a single company, this type of analysis will not be pursued.

For each bin volume, the cost of the bin is calculated, and with Equation 3.1, the increase in production over a zero bin situation is found and then multiplied by the value of the product, $3.25 per cubic metre, to find the savings realized in one year due to the bin. The results are tabulated in Table 4.3.

Bin Volume (m^3)	Bin Cost ($)	Savings Over 1 Yr. ($)	Payback (years)
50	175,000	142,002	1.23
100	350,000	280,808	1.25
200	700,000	549,153	1.27
300	1,050,000	805,591	1.30
400	1,400,000	1,050,644	1.33
500	1,750,000	1,284,819	1.36
1000	3,500,000	2,308,711	1.52
2000	7,000,000	3,774,936	1.85
3000	10,500,000	4,706,104	2.23
4000	14,000,000	5,297,477	2.64
5000	17,500,000	5,673,050	3.09

Table 4.3 Payback period for bin construction costs

Clearly, the bin is a very cost efficient method of increasing production. The costs and profits have been assumed, and are open to discussion. Again, the absolute value of the numbers is not important, rather the use of the function is the key point.

5. Bin Perturbation Analysis

5.1 Introduction

The major drawback to the previous method (brute force simulation) is the number of runs required to produce the final result. In many real world situations, computer time is always a scarce resource; therefore, any procedure that reduces processing and turnaround time would be a welcome one. Such a procedure has been developed by Ho and others (14,18,19) and will be employed here and the results compared with those from the previous sections. This procedure will allow for the estimation of the partial derivative of system performance with respect to bin size, in addition to the usual point estimate obtained from a single simulation result.

Previously, many simulation runs were performed, each yielding a single point estimate of production. These were then used to produce an analytical function relating bin size to system production. As this function is continuous and differentiable over the interval $0 \le x < \infty$, where x is the bin size, the partial derivative of system performance with respect to bin size exists. With only minor modifications, the previous model can be used to estimate this partial derivative within a single run. This estimate will then be compared with the partial derivative obtained from the analytical function derived in the preceeding chapters. This comparison will be done using standard, statistical methods. The accuracy and the applicability of the perturbation method will be explored.

5.1.1 Outline of Chapter

This chapter is divided into two sections:

1. explanation of bin perturbation analysis
2. examination of results

In order to completely (and intially, simply) define what bin perturbation analysis (BPA) is, Section one will contain two separate (but related) discussions. Firstly, there will be a written explanation of the theory. Second, a more rigorous, algebraic presentation will be given

The final section in this chapter contains the results and all associated comparisons and conclusions.

Appendix B details the changes made to the model, the initial conditions and any other related modifications.

5.2 Discussion of Bin Perturbation Analysis

Bin perturbation techniques allow for evaluation of the estimate of the partial derivative of system performance with respect to bin size by simulating the nominal (original) system and evaluating a "what if" question at key event times. Specifically, at each bin full event, the question "what if the bin had an extra volume equal to one unit of production?" is asked. The production that could have been realized if that extra unit of storage had been available is evaluated. By summing this quantity over simulated time, the partial derivative mentioned previously may be calculated.

To visualize the technique, consider a worker standing by the bin with instructions to hold one unit of production should the bin become full. By taking the unit of production, the worker allows the input machine to continue to produce for a time period equal to the unit volume divided by the production rate of the machine. This gain in production time is termed a local gain. The next time this worker holding the unit of production may act constructively is when the bin becomes empty. He now relinquishes the volume, thus allowing the output machine to process one more unit of production. The local gain that was initiated by the previous bin full condition has been realized by the system and thus becomes a system gain.

It is important to realize that a local gain does not necessarily become a system gain. If the bin becomes full only once during the production history and never empties, the worker does not get the opportunity to realize his local gain. Also, as the worker can only hold one unit of production, suceeding "bin full" events without intervening "bin empty" events are of nil value. There also exist "propagation rules" when there are multiple bins. These rules describe how gains at one point in the system are transmitted through, eventually becoming a system gain or being cancelled altogether. For the simple system under consideration here, these rules are not necessary as there is only one bin from which a "gain" can originate and with upstream feed and downstream sink infinite local gains are always global.

5.3 Algebraic Presentation of BPA Techniques

A more rigorous, algebraic definition of the preceeding section will be given. This explanation follows, in essence, that presnted by Eyler (14) and Ho (19) but will contain the modifications and simplifications allowed for by the simple system under consideration. For a more detailed treatment of the proofs, the reader is directed to the above references. The list of symbols employed in this development ,which are based on those of Eyler (14), are given in Table 5.1.

As this thesis deals with a two machine, one buffer (bin) operation, the algebra will reflect this. The method is not limited to this configuration and may be extended to far more complex cases.

The transfer line consists of the input machine, M_1, the intermediate storage bin, B_1, and the output machine, M_2. The assumption of Chapter 1 regarding infinite supply and sink remain in effect. The possible states for this system are then;

Assumption 1:

1. M_1 up, M_2 up, production at 1400 $\frac{m^3}{h}$
2. M_1 down, M_2 up, $B > B_{min}$, production at 1400 $\frac{m^3}{h}$
3. M_1 down, M_2 up, $B = B_{min}$, production at 0, M_2

Symbol	Meaning
M_1	Machine one; Input
M_2	Machine two; Output
B_1	Bin between Input and Output
B_{min}	Minimum bin volume
B_{max}	Maximum bin volume
P_i	Observation period for machine i
m_i	Total production for machine i
ρ_i	Production rate for machine i
d_k	Duration of k^{th} failure
$y_i(t)$	Production history of machine i over interval
$x(t)$	Bin content over interval
g_i	Local gain of machine i
$y_i(t)$	Perturbed production history of machine i
$x_i(t)$	Perturbed bin content
$M_i(t)$	Perturbed machine state

Table 5.1 List of symbols used

is forced down due to M_1

4. M_1 up, M_2 down, $B < B_{max}$, production at 1400 $\frac{m^3}{h}$

5. M_1 up, M_2 down, $B = B_{max}$, production at 0, M_1 is forced down due to M_2

6. M_1 down, M_2 down, production at 0

Notice here that the system is either producing at full capacity or zero. In this system, operation at less than capacity is not possible; hence, there is no slowed down (SD) state as considered by Ho and Eyler.

In order that the system start and end with the same condition, an empty bin, each unit (m^3) "mined" by M_1 will be "processed" by M_2. The observation period will be set to accommodate this condition. Previously, the observation period had been set as a function of the bin volume. Now, as the concern is with the number of times the bin becomes full/empty, it is necessary to ensure that all of these occurrences have had the opportunity to make their way through the system.

Assumption 2:

Given that;

P_i = observation period for machine i

m_i = total production for machine i

ρ_i = production rate for machine i

d_k = k^{th} failure

Then,

$$m_i = (P_i - \Sigma d_k)*\rho_i \text{ for } i = 1,2$$

Random failures have been assumed to be operation time dependent, so the sequence and duration of failures for a machine in the line will be equivalent to those of an "isolated machine". That is, by placing the machine in the line, we have not affected the sequence or duration of random failures as they would have occurred if the machine had been operating without the interference of the second machine. However, what will change will be the actual time that these events occur. This is so since the machine may be forced down by the other machine and undergo a non operating period where it will not deteriorate, and thus the random failure will occur later in time than for the stand alone machine. This is an important concept since, by placing the machine in the line, we are not altering anything physical, merely the timing of certain events. This concept of "shifting time" is central to perturbation techniques. By placing the machine in the line, we have shifted the occurrence of events further down the time line (due to machine interference). In perturbation analysis, we are attempting to analyze the effect on the system, when events are shifted to the left on the time line (making the events occur sooner than in the non-perturbed path). Again remember, nothing physical is being altered. Only the occurrence of the event in time is being changed.

Definition: The nominal production history for M_i, during the interval $0 \leq t \leq s$, is:

$$y_i(t) = 0 \text{ if } r_k - d_k \leq t \leq r_k, \quad 1 \leq k \leq N$$

$$y_i(t) = \rho_i \text{ otherwise}$$

where:

r_k is the end of the k^{th} failure

d_k is the duration of the k^{th} failure

N is the number of failures observed

The total production, m_i, through time is now:

$$m_i = \int_0^s y_i(t)dt$$

The amount of material in the bin at time t is denoted as $x(t)$ with the maximum value of $x(t)$ equal to the bin capacity, B_{max}. The change in the bin content is then;

$$x(t) = y_1(t) - y_2(t) \quad 0 \le t \le s$$

As stated previously, $x(0) = 0$.

formulations will now be used to explain the of a small change, or perturbation, in the history of a machine. This perturbation may be of any variable in the system, but for the purposes here, only changes in the bin size will be examined. By changing (increasing) the bin volume by a small amount, δ, a machine will achieve production values that would have previously occurred later in time. This is the shifting of the time line mentioned earlier. The machine is now said to have realized a local gain, g_i, and y_i becomes:

$$y_i'(t) = y_i(t), \ t \le t_0$$
$$= y_i(t + g_i) \ t_1 \le t \le t_2$$

$[t_1, t_2]$ is the interval in time that the system is being perturbed. The bin content will also be changed:

$$x_1'(t) = x_1(t) + \rho_i * g_1 - \rho_2 * g_2$$

In order that this perturbation does not cause any new forced down (FD) states, it is assumed that g_i is sufficiently small so that no new bin full/empty conditions are generated, i.e.:

$$B_{min} \le x_1'(t) \le B_{max} \text{ during all } g_i$$

Assuming that the perturbation has taken place, it still remains to define the path of the perturbation, or its propagation through our simple system. An important point to realize here is that the two machines only affect one another through full output (FO), i.e. bin is full and the input machine must shut down, or no input (NI) conditions, bin is empty and the output machine must shut down. Between these two occurrences, the bin volume has no effect on the production of the two machines. The unit gain in production that began with the first FO condition will remain in the system until the first NI condition occurs. This is the "bounce" that Ho (19) refers to, i.e. the bin volume has changed from full to empty. Since the gain originated at the input machine, through the bin full event, its production history has already been shifted forward in time:

$$M_1'(t) = M_1(t + g_i) \ \forall \ t < t_2$$

This is providing a "shift" does not occur. The concept of shifting will be discussed later in this chapter.

This gain in time will be propagated to machine 2, the output machine, only when there is a no input conditon. Let there be the first NI during $[t_3, t_4]$ after the initial bin full conditon. Now, since M_2 cannot be affected by the status of M_1 as long as $x(t) > B_{min}$, we have:

$$M_2 = M_2' \qquad \forall \ t_1 < t \le t_3$$

The ending of the no input state is dependent solely upon M_1, whose state has been translated forward by the size of the initial gain, g_1. So now;

$$NI_2'[t_3, t_4'] = NI_2[t_3, t_4] - g_1$$

$$M_2'(t) = M_2(t + g_1)$$

The gain has now been propagated to M_2 and is of the same magnitude as the original gain at M_1. This gain is now termed a "system gain". This cycle or "bounce" is illustrated in Reference (19).

5.4 Shifting

The term shifting refers to a major shift in the production history of a machine. If, by virtue of the extended production time caused by the increase in the bin volume, M_1 fails, the FO condition is eliminated. This condition (the FO) is replaced by simultaneous down states of both machines. This phenomena causes a large increase in productivity, not accounted for by BPA theory. The production rates used for the experiments in this work are quite large and the time required to produce an extra "unit" is very small;

With $\rho_1 = \rho_2 = 1400 \frac{m^3}{h}$,

Time to produce $1 \frac{m^3}{h} = 1. / 1400.$

$= 0.00071$ hours

The probability of a machine failing during this time span is low and thus, shifting will be ignored.

5.5 Calculation of the Estimate

If, throughout the simulation, these time gains are summed, and the time to produce m units is denoted as S, then the gain in time due to the increase in volume is:

$$s = S(B) - S(B+1)$$

Taking the change in B as ΔB:

$$s = \frac{1}{\Delta B} * (S(B) - S(B+\Delta B))$$

Taking the limit as ΔB approaches 0:

$$-s = \frac{\partial S}{\partial B}$$

For production, P, $= \frac{m}{S}$, where

m = units produced (constant)

S = time to produce m units

then;

$$s = \frac{m}{P^2} * \frac{\partial P}{\partial B}$$

or:

$$s = \frac{m}{(m/s)^2} * \frac{\partial P}{\partial B}$$

$$s = \frac{S^2}{m} * \frac{\partial P}{\partial B}$$

Finally,

$$\frac{\partial P}{\partial B} = \frac{s*m}{S^2}$$

If m is set, S will be the time to produce these units and s is the estimate of the system gain. We are able to calculate the estimate of the partial derivative of system performance with respect to bin size, $\frac{\partial P}{\partial B}$. It is this value that will be compared with the partial derivative of the previously developed equation.

5.6 Simulation

5.6.1 Implementation and Comparison

The changes to the original simulation code, as shown in Appendix B, were coded and the model re-run at each of the previously simulated bin volumes. The results from this simulation are tabulated in Table 5.2. The trend exhibited by these estimates is as expected. The estimates are approximately constant, and then gradually decrease. As the estimates are of the partial derivative with respect to bin volume, they indicate a linear relationship between bin volume and system productivity up to a certain bin volume, after which system productivity is independent of bin volume. This "asymptote" occurs at approximately 2000 m^3 of bin volume. From bin capacities of 2000 m^3 to 3000 m^3 the derivative sharply decreases and loses an order of magnitude by 4000 m^3. In validating this technique, the derivative of the previously obtained regression equation may be calculated easily and then compared with BPA results at the various bin volumes.

The equation:

$$PROD = -0.2498 * e^{-2.271 * BINSCL} + 1.0267$$

Taking the first derivative with respect to BIN:

$$\frac{\partial PROD}{\partial BINSCL} = (0.5673) * e^{-2.271 * BINSCL}$$

This equation was then evaluated at each bin volume. The results compare favourably with the BPA technique, as the trends are similar. The area in which the BPA technique

Bin Volume (m^3)	Perturbation Result	Function Value
50	0.05441	0.1109
100	0.05467	0.1086
200	0.05594	0.1036
300	0.05568	0.09901
400	0.05659	0.09461
500	0.05638	0.09040
1000	0.05715	0.07204
2000	0.05591	0.04574
3000	0.01629	0.02905
4000	0.00470	0.01844
5000	0.00244	0.01171

Table 5.2 BPA estimates and function derivative values

deviates most noticeably from the analytical value is at the smaller bin volumes. It is here that the most "noise" will be exhibited. The frequent bin full/empty conditions, the tolerance and step size all have a relatively larger effect on the accuracy of the final result than for the other sizes. BPA analysis indicates an almost linear relationship up to approximately 3000 m^3. The analytical function does not exhibit this property, but the magnitude of the derivative of both methods does begin to decrease most noticeably at approximately 3000 m^3. The type of information provided by BPA is comparable to that given by brute force simulation. It can be concluded that the BPA techniques underestimate the derivative for volumes up to 1000 m^3, fit well between 1000 and 3000 ^{3} and then underestimate once again for the larger bin volumes. Again this is not suprising with the large number of events and accompanying noise at the lower levels, and the relatively (statistically) small number of observations that will occur at the large volumes. Since BPA underestimates, this would provide a conservative estimate in terms of productivity improvement, with respect to increasing the bin volume.

Stability of Estimate

While work done by Zazanis (28) indicates a certain stability of the PA estimate, this warrants further investigation for the system under study here. Zazanis' work is in the area of single server queues. The system here is different in that it is a transfer line, and the flow is continuous. There are several methods for doing this, one being multiple runs as was done for the regression equations, or alternatively, examination of the estimate $\frac{\partial P}{\partial B}$ at each "bounce" of the production history. Each bounce will provide a realization and the distribution of this variate will be examined. Specifically,

$$\frac{\partial P}{\partial B} = \frac{m_t * s[t]}{t^2 * T}$$

where:

$\frac{\partial P}{\partial B}$ = estimate of partial derivative with respect to bin volume.

m_t = volume of material processed at time t.

s(t) = number of bounces at time t.

T = cycle time.

t = time of event.

By recording the estimate, its stability, as well as its independence (or dependence) on time and material processed will be indicated. Recall that a volume of material was placed before M_1 and when this volume had been processed, the simulation was halted and the estimate calculated. The progression of this estimate would indicate whether the volume chosen was sufficient.

Initial simulation runs were performed and the results indicated that the total volume to be produced by the system, m, was not sufficient. The progression of the estimate demonstrated that a steady-state had not yet been achieved. Closer examination of the estimate revealed large fluctuations in the initial values. This situation indicated that the estimate is sensitive to the start-up conditions. The model was re-run with the same initial volume, but with statistics only being recorded after five hundred (500) hours had been simulated. The results still indicated a problem with the estimate. In order that the estimate achieve a steady state, the volume to be processed, m, was increased by a factor of 10 to $1 * 10^7$ m^3. For a system with a 1000 m^3 bin, this would require, on average, 11,429 hours of simulated time. The estimates were found to be relatively stable, and the final estimate of each simulation was more accurate, with respect to the the mean value from the 24 separate simulations than was the case for the smaller volume to be processed.

Since the individual estimates had been recorded, an analysis of their distribution could be done. Using the SLAM summary reports, weak normality is indicated for most cases. With the larger bin volumes, the number of bounces is small and it becomes difficult to comment on the distribution type.

While Zazanis reports generally well-behaved and stable estimates, this has not been found in this study. The estimate is sensitive to start-up conditions and requires a (proportionately) long individual simulation run, with respect to the brute force simulation method, to stabilize the final results. It should be kept in mind that in order to obtain the estimate of the derivative with the brute force method, many simulation runs were necessary, in addition to a regression of the data.

The normality of the estimates, while demonstrated in some cases, has not been proven to exist in all cases.

5.6.3 Conclusions of PA Applicability

Perturbation analysis is a derivative estimation technique, and, as with any such technique, there is error involved. PA does allow for the estimation of system partial derivatives within a single run with a minimum of computational effort. These estimates have been shown to exhibit similar trends as the experimentally derived function. The accuracy of the estimates is a function of the shape of the analytical equation and the portion of the curve that is under consideration. PA estimates perform best near the "optimum value"; that is, where the curve begins to approach the theoretical maximum limit. In areas of quickly changing derivatives, such as in the initial bin volumes, PA techniques provide an order of magnitude type estimate. The main drawback, with the function under study here, is that PA techniques assume a linearly increasing function of production with respect to bin volume. Therefore, the derivative is relatively constant up to the asymptote. The derived function indicates strongly non-linear behaviour in this area, and the change in productivity with an increase in bin volume in this area is much more significant than PA indicates. What PA then fails to tell the engineer, in this case, is which portion of the curve his system currently operates on. The proportionate increase in productivity per increase in bin volume is unclear using PA only.

The benefits of PA are that it does allow for quick evaluation of system performance within a single run, and may help to justify additional and more detailed study.

A very important benefit is that PA may be implemented through a simulator, or with actual field data. In regards to bin studies, the information about the bin full/empty conditions is likely recorded at many mine sites. This information could be utilized on a reqular basis to evaluate the effectiveness of the operation. As the information is available, the cost of implementation would be likely quite low.

6. Recommendations and Conclusions

In this thesis, a function was developed that related system productivity to bin volume for three different, yet controlled systems. It was then shown that these functions could be normalized by machine failure parameters, bin volume, and system expected production rate to produce one function. This function could then be extended to other systems that met the design criteria. Reiterating these criteria, they are:

1. Input Rate = Output Rate
2. Identical Failure Distributions for Input and Output
3. Identical Repair Distributions for Input and Output
4. Constant COV

As these conditions may place unrealistic constraints on the usage of the function, future research should be directed at their removal. The recommended place to start would be to develop a function that allows for different coefficients of variation. Further work then could extend to simultaneous removal of the other three conditions, as they are likely inter-related. The more failure prone system will likely be overdesigned so there is some "catch-up" capacity so that the expected production rate of input equals the expected production rate of the output.

Further extensions of the work could lead into more complex lines, with multiple bins and greater than two production systems.

The development of a truly general function would provide major benefits to engineers, reducing time and costs in the preparation of productivity studies.

In the course of developing the more general function, a detailed study of the experimental design must be done so that the distribution of the residuals may be better understood. Until this is done, no detailed statistical statements can be made about the functions that are developed. As mentioned previously in Chapter 3, the run lengths at each bin volume may need to be adjusted. At the lower bin volumes there is proportionately more noise involved due to the large number of bin full/empty events, than would be present at the larger bin volumes.

The techniques of PA were analyzed and compared to the simulation generated function. PA provides an opportunity for the better use of data, which in many cases may already exist. While it is sensitive to simulation start-up conditions and duration, it does provide the engineer with a general idea of how the system may operate. The accuracy of PA statistics also depends on the level at which the system is currently operating and the type of relationship between productivity and bin volume.

Care must be used to not infer too much from the PA results as they may be misleading, especially in the areas where the addition (or enlargement) of a bin has the most potential benefits.

References

1. Ang, A.H.S., and Tang, W.H., 'Probability Concepts in Engineering Planning and Design', Vol. 1, Basic Principles, John Wiley and Sons, 1975

2. Barlow, R.E., and Proschan, F., 'Availability Theory of Multicomponent Systems', Report OFR 72-8, University of California, Berkley, April, 1972

3. Bradely, C., Taylor, S.G., and Gray, W.I., 'Sizing Storage Facilities for Open Pit Coal Mines', IIE Transactions, Vol.17, No. 4, 320-326, 1985

4. Buzacott, J.A., and Hanifin, L.E., 'Models of Automatic Transfer Lines with Inventory Banks, A Review and Comparison', AIIE Transactions, June, 197-207, 1978

5. Chu, Jean, and Ermolowich, M., 'Determining the optimum size of a coal silo', Simulation, December, 191-197, 1980

6. Cooper, R.B., 'Introduction to Queueing Theory', MacMillan Company, New York, 1972

7. Cruz, L.D., Solari, M.A., and Violic, I.A., 'Matching Los Bronces Mine to Mill System Capacities through Computer Simulation', Mining Engineering, Feb., 159-162, 1983

8. Dixon, W.J., ed., 'BMDP Statistical Software Manual', University of California Press, 1985

9. Doe, D.C., and Griffin, W.H., 'Experimental Design and Mining Systems Simulation', Continuous Surface Mining , T.S. Golosinski and F.G. Boehm, eds., Trans Tech Publications, 317-324, 1987

10. Efroymson, M.A., 'Multiple Regression Analysis', Mathematical Methods for Digital Computers, Vol. 1, John Wiley and Sons Inc., 1967

11. Elbrond, Jorgen, 'A Procedure for the Calculation of Surge Bin Sizes', bulk solids handling, Vol. 12, No. 3, Sept., 1982

12. Elbrond, Jorgen, personal communication (1988).

13. Eyler, M.A., 'A New Approach to the Productivity Study of Transfer Lines', Ph.D. Thesis, Harvard University, Division of Applied Science, 1979

14. Fusinato, L.A., 'Computer Simulation to Aid Open Cut Plant Selection',The Aus. I. M. M. Conference, Melbourne, Victoria, Aug., 1982

15. Golosinski, T.S.,'Surface Mining Systems in Athabasca Oil Sands, A Review of Performance',bulk solids handling, Vol. 5, No. 1, 149-154, 1985

16. Golosinski, T.S., and Griffin, W.H.,'Modelling of an Oil Sands Plant Operation Using SLAM II',Bergund Huttenmmannisher Tag, Freiberg,G.D.R., June 1985

17. Griffin, W., 'Perturbation theory applied to truck-shovel system simulation',Off Highway Haulage in Surface Mines,T.S. Golosinski and V. Srajer eds.,A.A. Balkema,Rotterdam,83-87,1989

18. Ho, Y.C., ed., 'SPEEDS, A New Technique for the Analysis and Optimization of Queueing Networks', Technical Report No. 675, Divison of Applied Sciences, Harvard University, Cambridge, Massachusetts, Feb., 1983

19. Ho, Y.C., Eyler, M.A., and Chien, T.T., 'A gradient technique for general buffer storage design', Int. J. Prod. Res. , Vol. 17, No. 6, 557-580, 1979

20. Hull, T.E., and Dobell, A.R., 'Random Number Generators', SIAM Review, Vol.4, 230-254, 1962

21. Koenigsberg, E., 'Production Lines and Internal Storage, A Review', Management Science, Vol. 5, 410-433, 1959

22. Lowell, S., and Baca, R., 'Bucket Wheel Excavators and Their Use at the Suncor Mine', Continuous Surface Mining, 45-50, T.S. Golosinski and F.G. Boehm, Eds., Trans Tech Publications, 1987

23. Pritsker, A.A.B., 'Introduction to Simulation and SLAM II, 2nd' , Halstead Press, New York, 1984

24. Ralston, M., 'PAR Derivative-Free Nonlinear Regression', BMDP Statistical Software Manual Dixon, W.J., chief editor, University of California Press, 305-314, 1985

25. Sevat'yanov, B.A., 'Influence of Storage Bin Capacity of the Average Standstill Time of a Production Line', Theory of Probability and Its Application, Vol. 7, No. 4, 1962

26. Tilmann, W., and Weise, H., FRG, 'The Continuous Surface Mining Technique and its Present State as seen from Western Europe', Continuous Surface Mining,T.S. Golosinski and F.G. Boehm, eds., Trans Tech Publications, 317-324, 1987

27. Wagner, H.M., 'Principles of Operations Research, 2nd', Prentice Hall, 1975

28. Zazanis, M.A., 'Statistical Properties of Perturbation Analysis Estimates for Discrete Event Systems', Ph.D. Thesis, Harvard University, Division of Applied Science, 1986

Appendix A

Contents of Appendix A:

-pp 94 - 97 Explanation and source listing of Elbrond code.

-page 99-100 Algebra for reduction of heuristic approach to that of Barlow and Proschan.

page 98 Algebra for calculation of the B0 parameter.

The procedures of Elbrond (11) have been coded in FORTRAN and run on the system at the University of Alberta. The listings of the code are provided at the end of this appendix. The program has been validated by using it to reproduce graphs presented in Elbrond's paper.

While a relatively simple tool to use, Elbrond's formulations only indirectly account for the randomness of machine outages. In comparing results, Figures 4.1-4.3 we see that Elbrond's values are always conservative with respect to those obtained through simulation. As the procedure is not using the random outage data directly, the answers may be purposely conservative.

It must be noted here that only the balanced case has been coded and tested. Addition of the logic for the non-balanced (with respect to internal outage data) cases would prove to be more difficult.

When running the data, a drawback becomes apparent in providing "loss calculation" data for the bin. By assuming a frequency, some knowledge of the bin and system is implied. Clearly in an initial design stage, this type of knowledge may not be available. As well, the same frequency was used throughout the tests as no better data existed. The larger bin volumes would clearly have a different loss frequency than the smaller bins. As it was not the purpose here to investigate the sensitivity of this procedure to various input parameters, this simplistic approach was selected.

The net result of this study has been a validation, of sorts, of the simulation technique applied throughout this thesis, with the analytical approach of Elbrond. The results of the simulation compare well with those of Elbrond and increase the confidence in the correctness of the model.

In light of the opportunities presented by perturbation analysis and the benefits of dynamically modelling a stochastic system, any reduction in computing and set-up provided by the analytic technique is to a large extent negated.

```
      PROGRAM ELBRND
C-----------------------------------------------------------------------
C
C  SIMPLE PROGRAM TO EVALUATE ELBROND FUNCTION FOR MATHCED MACHINE
C  CASE ONLY.
C
C  REFERENCE: ELBROND,J.,'A PROCEDURE FOR THE CALCULATION OF
C             SURGE BIN SIZES',BULK SOLIDS HANDLING,VOL.12,NO.3,
C             SEPT.,1982
C-----------------------------------------------------------------------
      REAL I1,I2
C                                    READ IN THE DATA FROM THE SCREEN
      PRINT *,'ENTER THE TWO SYSTEMS PRODUCTION TIMES, DOWNTIMES,
     + AND RATES RESPECTIVELY'
C
      READ(*,*)P1,P2,A1,A2,I1,I2
C                                     CALCULATED AVAILABILITIES
      EFF1=P1/(P1+A1)
      EFF2=P2/(P2+A2)
C                                     THE "STANDARD DEVIATION"
      S1=I1*A1*SQRT(EFF1)
      S2=I2*A2*SQRT(EFF2)
C                                     THE BIN "STANDARD DEVIATION"
      BINSD=SQRT(S1**2 + S2**2)
C                                     CALCULATE FOR A SET OF BIN VOLUMES
      DO 10 I = 0, 5000, 500
       BINVOL = FLOAT(I)
        F = 0.5*(BINVOL/BINSD)
        A=F*.5
        B=F*.75
        TERM1=BIN(A)
        TERM2=BIN(B)
        TERM3=BIN(F)
         BINLOS = 0.25*TERM1+0.50*TERM3+0.25*TERM2
         CAP=(MIN(I1,I2)*EFF1*EFF2*BINLOS)+I2*EFF2*(1-BINLOS)
        WRITE(7,100)BINVOL,CAP
 100    FORMAT(F8.1,F7.2)
  10  CONTINUE
C
      STOP
      END
```

```
C
      FUNCTION BIN(VAL)
      CORRV=0.5*ERF(VAL/SQRT(2.))+0.5
      T1=(1./SQRT(2.*3.14159))*(1./(EXP((VAL**2)/2.)))
      BIN=(SQRT(2.*3.14159))*(T1-(VAL*(1-CORRV)))
      RETURN
      END

================================================================================
SAMPLE OUTPUT
     0.0 714.29  <----   NOTE THIS IS EQUIVALENT TO 1400*(5/7)*(5/7) WHICH
   500.0 735.46          IS THE PRODUCT RULE AVAILABILITY CALCULATION.
      .
      .
      .
```

educe the heuristic approach to that of Barlow Proschan:

$$\text{Given, } \pi_0 = \frac{\left[\dfrac{\mu_1}{\mu_1 + \nu_1}\right] \cdot \left[\dfrac{\mu_2}{\mu_2 + \nu_2}\right]}{1 - \left[\dfrac{\nu_1}{\mu_1 + \nu_1}\right] \cdot \left[\dfrac{\nu_2}{\mu_2 + \nu_2}\right]}$$

Gathering like terms:

$$\pi_0 = \left[\frac{1}{1 - \dfrac{\nu_1\nu_2}{\mu_1\mu_2 + \mu_1\nu_2 + \nu_1\mu_2 + \nu_1\nu_2}}\right] \cdot \frac{\mu_1\mu_2}{\mu_1\mu_2 + \mu_1\nu_2 + \nu_1\mu_2 + \nu_1\nu_2}$$

$$= \frac{\mu_1\mu_2}{\mu_1\mu_2 + \mu_1\nu_2 + \nu_1\mu_2}$$

$$\pi_0 = \frac{1}{1 + \dfrac{\nu_1}{\mu_1} + \dfrac{\nu_2}{\mu_2}}$$

Which is equivalent to the Barlow–Proschan formula for $j = 2$.

From availability theory, we want B0 such that:

$$(-B0 + 1.0) \cdot PROD_{sys(\infty)} = \frac{PROD}{1 + \frac{\nu_1}{\mu_1} + \frac{\nu_2}{\mu_2}}$$

so then we want:

$$(B0 \cdot PROD_{sys(\infty)}) = PROD_{sys(\infty)} - \frac{PROD}{1 + \frac{\nu_1}{\mu_1} + \frac{\nu_2}{\mu_2}}$$

dividing by $PROD_{sys(\infty)}$:

$$B0 = 1 - \left[\frac{PROD}{1 + \frac{\nu_1}{\mu_1} + \frac{\nu_2}{\mu_2}}\right] \cdot \frac{1}{PROD_{sys(\infty)}}$$

but since; $$PROD_{sys(\infty)} = PROD \cdot \left[\frac{\mu_1}{\mu_1 + \nu_1}\right]$$

$$B0 = 1 - \left[\frac{1}{1 + \frac{\nu_1}{\mu_1} + \frac{\nu_2}{\mu_2}}\right] \cdot \left[\frac{\mu_1}{\mu_1 + \nu_1}\right]^{-1}$$

$$B0 = 1 - \frac{\mu_1 + \nu_1}{\mu_1 \cdot \left[1 + \frac{\nu_1}{\mu_1} + \frac{\nu_2}{\mu_2}\right]}$$

$$B0 = 1 - \frac{\mu_1 + \nu_1}{\left(\mu_1 + \nu_1 + \frac{\mu_1}{\mu_2} \right)}$$

and since $\mu_1 = \mu_2$ and $\nu_1 = \nu_2$:

$$B0 = 1 - \frac{(\mu_1 + \nu_1)}{(\mu_1 + 2 \cdot \nu_1)}$$

simplyfing;

$$= \frac{\nu_1}{\mu_1 + 2 \cdot \nu_1}$$

Appendix B

SLAM PC student version 3.0 used.

Microsoft PC FORTRAN complier version 3.31 used.

Contents of Appendix B:

FORTRAN subroutines for perturbation analysis

```
C*********************************************************************
C FOR DISCRETE EVENT PERTURBATION ANALYSIS ON A SURGE BIN FED        *
C BY A CONTINUOUS INPUT THAT HAS A SINGLE RANDOM FAILURE MODE        *
C AND AN OUTPUT THAT HAS THE SAME MEAN AS THE INPUT BUT MAY HAVE     *
C A DIFFERENT RANDOM DISTRIBUTION OF FAILURES THAN THE INPUT.        *
C                                                                    *
C CHANGES HAVE BEEN INCORPORATED TO TRY AND DUPLICATE THE            *
C WORK BY Y.C. HO AND COMPANY.  THESE CHANGES ARE NOTED IN           *
C IN THE CODE.                                                       *
C                                                                    *
C EVENT CODES:                                                       *
C    1=INPUT PROCESS HAS STOPPED OPERATING DUE TO BREAKDOWN.         *
C    2=INPUT BREAKDOWN HAS BEEN REPAIRED.                            *
C    3=OUTPUT PROCESS HAS STOPPED OPERATING DUE TO BREAKDOWN.        *
C    4=OUTPUT BREAKDOWN HAS BEEN REPAIRED.                           *
C    5=SURGE BIN IS FULL. INPUT MUST STOP.                           *
C    6=SURGE BIN HAS FALLEN BELOW INPUT STARTUP LEVEL.               *
C    7=SURGE BIN IS EMPTY. OUTPUT MUST STOP.                         *
C    8=SURGE BIN HAS RISEN ABOVE OUTPUT STARTUP LEVEL.               *
C    9=THE DESIRED PRODUCTION HAS BEEN MET.  INPUT IS SHUT OFF       *
C   FILES USED:                                                      *
C         1=FILE TO HOLD REMAINING TIME ON RUNNING TIME TO FAILURE FOR *
C          INPUT AND OUTPUT PROCESSES THAT ARE CONSTRAINED BY BIN FULL *
C          OR EMPTY CONDITIONS.                                      *
C                                                                    *
C      NCLNR=CALENDAR FILE HOLDING FUTURE DISCRETE EVENTS 1,2,3,4.   *
C                                                                    *
C  ATTRIBUTE CODES:                                                  *
C         1=EVENT CODE IN FILE 1.                                    *
C         2=TIME REMAINING OF RUNNING TIME TILL EVENT WILL OCCUR.    *
C         3=EVENT CODE IN FILE NCLNR.                                *
C         4=EVENT TIME IN FILE NCLNR.                                *
C                                                                    *
C  COMMON VARIBLES (SCOM1)                                           *
C         XX(1)=INPUT RATE                                           *
C         XX(2)=OUTPUT RATE                                          *
C         XX(3)=MEAN OPERATING TIME TO FAILURE OF INPUT              *
C         XX(4)=ST. DEV. OPERATING TIME TO FAILURE OF INPUT          *
C         XX(5)=MEAN TIME TO REPAIR INPUT FAILURE                    *
C         XX(6)=ST. DEV. TIME TO REPAIR INPUT FAILURE                *
C         XX(7)=MEAN OPERATING TIME TO FAILURE OF OUTPUT             *
C         XX(8)=ST. DEV. OPERATING TIME TO FAILURE OF OUTPUT         *
C         XX(9)=MEAN TIME TO REPAIR OUTPUT FAILURE                   *
C        XX(10)=ST. DEV. TIME TO REPAIR OUTPUT FAILURE               *
C        XX(11)=MAXIMUM SURGE BIN CAPACITY                           *
C        XX(12)=STARTUP LEVEL FOR STOPPED INPUT                      *
C              =(.98*XX(11))                                         *
```

```
C        XX(13)=STARTUP LEVEL FOR STOPPED OUTPUT                        *
C              =(.02*XX(11))                                            *
C        XX(14)=CUMULATED TIME GAINED FROM BIN SIZE PERTURBATION        *
C        XX(15)=DESIRED INPUT PRODUCTION                                *
C                                                                       *
C        SS(1) =BIN LEVEL                                               *
C        SS(2) =OUTPUT RATE                                             *
C                                                                       *
C    COMMON (USER) (LOGICAL SWITCHES)                                   *
C        BIN= .TRUE. SURGE BIN WILL ACCEPT INPUT                        *
C       BOUT= .TRUE. SURGE BIN WILL ALLOW OUTPUT                        *
C        PIN= .TRUE. INPUT IS CAPABLE OF OPERATING                      *
C       POUT= .TRUE. OUTPUT IS CAPABLE OF OPERATING                     *
C        STP= .TRUE. DESIRED PRODUCTION HAS BEEN MET.                   *
C      BPERT= .TRUE. BIN PERTURBATION IS BEING BEING PROPAGATED FORWARD *
C                                                                       *
C     PROGRAM MAIN                                                      *
C************************************************************************
$NOTSTRICT
$STORAGE:2
$NOTLARGE
$NOFLOATCALLS
      DIMENSION NSET(200)
      COMMON /SCOM1/ ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA,
     1       MSTOP, NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(100),
     2       SSL(100), TNEXT, TNOW, XX(100)
      LOGICAL*4 BIN,BOUT,PIN,POUT,STP,BPERT
      COMMON /USER/ BIN,BOUT,PIN,POUT,STP,BPERT
      COMMON QSET(200)
      EQUIVALENCE (NSET(1),QSET(1))
      OPEN(8,FILE='OTPUT.DAT',STATUS='OLD')
      NNSET = 200
      NCRDR = 5
      NPRNT = 0
      NTAPE = 7
      CALL SLAM
      STOP
      END
C************************************************************
C                                                           *
C     SUBROUTINE STATE SETS THE INPUT/OUTPUT RATE AND       *
C     UPDATES THE CONTINUOUS VARIABLE,SS(1), THE BIN        *
C     LEVEL AND THE OUTPUT RATE, SS(2).                     *
C                                                           *
C************************************************************
      SUBROUTINE STATE
      COMMON /SCOM1/ ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA,
     1       MSTOP, NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(100),
     2       SSL(100), TNEXT, TNOW, XX(100)
```

```
      LOGICAL*4 BIN,BOUT,PIN,POUT,STP,BPERT
      COMMON /USER/ BIN,BOUT,PIN,POUT,STP,BPERT
C***********************************************************
C                                                          *
C     THE INPUT/OUTPUT RATES ARE SET                       *
C                                                          *
C***********************************************************
      RTIN=XX(1)
      RTOUT=XX(2)
C***********************************************************
C                                                          *
C     IF THE BIN WILL ACCEPT INPUT AND THE INPUT STREAM    *
C     IS FUNCTIONING, A CHECK WILL BE MADE ON THE OUTPUT   *
C     CONDITIONS, OTHERWISE, THE INPUT RATE IS 0.          *
C     AS WELL, A CHECK IS MADE TO SEE IF THE DESIRED       *
C     INPUT LEVEL HAS BEEN MET, IF SO RATE IN = 0.0        *
C                                                          *
C***********************************************************
      IF((BIN.AND.PIN).AND..NOT.STP) GO TO 1
      RTIN=0.
C***********************************************************
C                                                          *
C     HERE THE CHECK IS ON THE OUTPUT STREAM AND THE       *
C     OUTLET VALVE ON THE BIN, IF THEY ARE BOTH TRUE,      *
C     I.E. THE BIN WILL ALLOW OUTPUT AND THE STREAM IS     *
C     FUNCTIONING, THE BIN LEVEL WILL BE UPDATED.  IF      *
C     ONE OR BOTH IS DOWN, THE OUTPUT RATE IS 0.           *
C                                                          *
C***********************************************************
    1 IF(BOUT.AND.POUT)GO TO 2
      RTOUT=0.
C***********************************************************
C                                                          *
C     THE BIN LEVEL IS UPDATED, AS WELL AS OUTPUT RATE     *
C     AS WELL AS THE INPUT LEVEL.                          *
C                                                          *
C***********************************************************
    2 SS(1)=SSL(1)+DTNOW*(RTIN-RTOUT)
      SS(2)=SSL(2)+DTNOW*RTIN
      SS(3)=SSL(3)+DTNOW*RTOUT
      RETURN
      END
C***********************************************************
C                                                          *
C     SUBROUTINE EVENT CALLS THE VARIOUS EVENTS (INPUT/    *
C     OUTPUT DOWN, BIN FULL/EMPTY, ETC.) AS WELL AS        *
C     PERFORMING THE BOOKKEEPING FUNCTIONS, I.E. WHEN      *
C     THE MACHINES ARE BIN CONSTRAINED, THEY ARE TAKEN     *
C     OUT OF THE CALENDAR FILE AND PLACED ON HOLD          *
```

```
C                                                          *
C***********************************************************
      SUBROUTINE EVENT(IX)
      COMMON /SCOM1/ ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA,
     1       MSTOP, NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(100),
     2       SSL(100), TNEXT, TNOW, XX(100)
      LOGICAL*4 BIN,BOUT,PIN,POUT,STP,BPERT
      COMMON /USER/ BIN,BOUT,PIN,POUT,STP,BPERT
      GO TO(1,2,3,4,5,6,7,8,9),IX
      PAUSE 'INVALID EVENT CODE'
C***********************************************************
C                                                          *
C     EVENT 1 IS THE INPUT PROCESS HAS HAD A BREAKDOWN     *
C     A CHECK IS MADE TO SEE IF BOTH THE STREAM AND THE    *
C     BIN INPUT VALVE ARE "TURNED ON".                     *
C                                                          *
C***********************************************************
    1 IF(BIN.AND.PIN)GO TO 100
      PAUSE 'ILLOGICAL STATE FOR EVENT CODE 1'
C***********************************************************
C                                                          *
C     THE INPUT STREAM IS TURNED OFF (PRODUCTION STOPS)    *
C     AND THE END OF BREAKDOWN SCHEDULED.                  *
C                                                          *
C***********************************************************
  100 PIN=.FALSE.
      CALL SCHDL(2,RLOGN(XX(5),XX(6),1),ATRIB)
      RETURN
C***********************************************************
C                                                          *
C     EVENT 2 IS THE END OF BREAKDOWN FOR THE INPUT        *
C     STREAM, IT IS STARTED UP AGAIN IF THE BIN IS         *
C     ACCEPTING INPUT, OTHERWISE THE STREAM IS PLACED      *
C     ON HOLD.                                             *
C                                                          *
C***********************************************************
    2 IF(PIN)PAUSE 'ILLOGICAL STATE FOR EVENT 2'
      PIN=.TRUE.
C***********************************************************
C                                                          *
C     RTIM WILL BE THE TIME TO THE NEXT BREAKDOWN          *
C                                                          *
C***********************************************************
      RTIM=RLOGN(XX(3),XX(4),2)
      IF(BIN)GO TO 200
      ATRIB(1)=1.
      ATRIB(2)=RTIM
C***********************************************************
C                                                          *
```

```
C     THE STREAM IS PLACED IN A HOLD FILE                             *
C                                                                     *
C*********************************************************************
      CALL FILEM(1,ATRIB)
      RETURN
C*********************************************************************
C                                                                     *
C     THE NEXT BREAKDOWN IS SCHEDULED                                 *
C                                                                     *
C*********************************************************************
  200 CALL SCHDL(1,RTIM,ATRIB)
      RETURN
C*********************************************************************
C                                                                     *
C     EVENT 3 IS THE BREAKDOWN OF THE OUTPUT STREAM, THE              *
C     LOGIC IS SIMILAR TO THE INPUT STREAM.(EVENT 1)                  *
C                                                                     *
C*********************************************************************
    3 IF(BOUT.AND.POUT)GO TO 300
      PAUSE 'ILLOGICAL STATE FOR EVENT 3'
  300 POUT=.FALSE.
      CALL SCHDL(4,RLOGN(XX(9),XX(10),3),ATRIB)
      RETURN
C*********************************************************************
C                                                                     *
C     EVENT 4 IS THE REPAIR OF AN OUTPUT BREAKDOWN, THE               *
C     LOGIC IS SIMILAR TO THE INPUT (EVENT 2)                         *
C                                                                     *
C*********************************************************************
    4 IF(POUT)PAUSE 'ILLOGICAL STATE FOR EVENT 4'
      POUT=.TRUE.
      RTIM=RLOGN(XX(7),XX(8),4)
      IF(BOUT)GO TO 400
      ATRIB(1)=3.
      ATRIB(2)=RTIM
      CALL FILEM(1,ATRIB)
      RETURN
  400 CALL SCHDL(3,RTIM,ATRIB)
      RETURN
C*********************************************************************
C                                                                     *
C     EVENT 5 IS A BIN FULL SITUATION                                 *
C                                                                     *
C*********************************************************************
    5 IF(BIN.AND.PIN)GO TO 500
      PAUSE 'ILLOGICAL STATE FOR EVENT 5'
C*********************************************************************
C                                                                     *
C     NO MORE FLOW IS ALLOWED INTO THE BIN.                           *
```

```
C                                                         *
C**********************************************************
  500 BIN=.FALSE.
C**********************************************************
C                                                         *
C     THE RANDOM FAILURES ARE TAKEN OUT OF THE CALENDAR   *
C     FILE AND PLACED ON HOLD. NFIND WILL LOCATE AN       *
C     ENTRY IN THE CALENDAR FILE WITH RANK GREATER THAN   *
C     OR EQUAL TO ONE WHOSE ATTRIBUTE 3 (I.E. BREAKDOWN   *
C     FOR INPUT LINE) IS EXACTLY EQUAL TO ONE.  RMOVE     *
C     THEN PLACES THE ATTRIBUTES OF THIS ENTRY IN THE     *
C     VECTOR ATRIB.                                       *
C     ATRIB(2) BECOMES THE WORKING TIME LEFT BETWEEN      *
C     THE BREAKDOWN AND THE PRESENT TIME.                 *
C     THE ATRIBUTES OF THE ENTRY ARE THEN FILED IN FILE   *
C     1.                                                  *
C                                                         *
C**********************************************************
      CALL RMOVE(NFIND(1,NCLNR,3,0,1.,0.),NCLNR,ATRIB)
      ATRIB(2)=ATRIB(4)-TNOW
      ATRIB(1)=1.
      CALL FILEM(1,ATRIB)
C**********************************************************
C                                                         *
C     HERE A CHECK IS MADE TO SEE IF THE BIN PERTURBATION *
C     WAS PREVIOUSLY SET, IF NOT, SET IT TO TRUE          *
C                                                         *
C**********************************************************
      IF(.NOT.BPERT) BPERT=.TRUE.
      RETURN
C**********************************************************
C                                                         *
C     EVENT 6 IS THE BIN LEVEL HAS FALLEN BELOW THE       *
C     START-UP LEVEL, THE VALVE IS OPENED AND THE         *
C     RESCHEDULED.                                        *
C                                                         *
C**********************************************************
    6 IF(BIN)RETURN
      BIN=.TRUE.
      IF(PIN)GO TO 600
      PAUSE 'ILLOGICAL STATE FOR EVENT 6'
C**********************************************************
C                                                         *
C     HERE THE ENTRY IN FILE 1 WHOSE ATTRIBUTE 1 IS       *
C     EQUAL TO 1 (INPUT BREAKDOWN) IS REMOVED AND THE     *
C     NEXT BREAKDOWN SCHEDULED ACCORDING TO ATRIBUTE 2    *
C     WHICH WAS THE TIME LEFT UNTIL BREAKDOWN WHEN THE    *
C     BIN FULL CONDITION OCCURRED.                        *
C                                                         *
```

```
C************************************************************
  600 CALL RMOVE(NFIND(1,1,1,0,1.,0.),1,ATRIB)
      CALL SCHDL(1,ATRIB(2),ATRIB)
      RETURN
C************************************************************
C                                                           *
C     EVENT 7 IS A BIN EMPTY CONDITION, THE OUTPUT IS       *
C     STOPPED, IT'S BREAKDOWN RESCHEDULED. THE FILING       *
C     IS SIMILAR TO THE BIN FULL CONDITIONS, EXCEPT         *
C     THAT ATTRIBUTE 3 IS EQUAL TO 3, (I.E. OUTPUT          *
C     BREAKDOWN).                                           *
C     FOR THE PERTURBATION CASE, A CHECK IS MADE ON         *
C     THE CURRENT INPUT LEVEL, IF IT HAS BEEN MET           *
C     THEN, THIS BIN EMPTY SIGNALS THE END OF THE           *
C     SIMULATION.  OTHERWISE, SET THE BIN PERTURATION       *
C     SWITCH TO OFF AND COLLECT THE TIME THAT COULD         *
C     HAVE BEEN GAINED IF ONE MORE UNIT OF STORAGE          *
C     HAD BEEN PROVIDED.  THIS VALUE IS EQUAL TO 1          *
C     DIVIDED THE OUTPUT PRODUCTION RATE, IN ORDER TO       *
C     MAINTAIN AN ACCEPTABLE LEVEL OF SIGNIFICANCE, THIS    *
C     VALUE IS MULTIPLIED BY 10000.                         *
C                                                           *
C************************************************************
    7 IF(STP) GO TO 701
      IF(BOUT.AND.POUT)GO TO 700
      PAUSE 'ILLOGICAL STATE FOR EVENT 7'
  700 BOUT=.FALSE.
      CALL RMOVE(NFIND(1,NCLNR,3,0,3.,0.),NCLNR,ATRIB)
      ATRIB(2)=ATRIB(4)-TNOW
      ATRIB(1)=3.
      CALL FILEM(1,ATRIB)
C************************************************************
C                                                           *
C     THE BPERT SWITCH IS CHECKED, IF TRUE, THE GAIN        *
C     IS CALCULATED AND THE SWITCH SET TO FALSE.            *
C                                                           *
C************************************************************
  702 IF(.NOT.BPERT) RETURN
      XX(14)=XX(14)+10000./XX(2)
      BPERT = .FALSE.
      RETURN
  701 MSTOP = -1
      GO TO 702
C************************************************************
C                                                           *
C     EVENT 8 INDICATES THE BIN LEVEL HAS RISEN ABOVE       *
C     THE START UP LEVEL, THE OUTPUT LINES ARE RE-          *
C     STARTED AND THEIR BREAKDOWN RESCHEDULED. AGAIN        *
C     THE PROCEDURE IS SIMILAR AS FOR THE BIN FULL CASE.    *
```

```
C                                                           *
C************************************************************
    8 IF(BOUT)RETURN
      BOUT=.TRUE.
      IF(POUT)GO TO 800
      PAUSE 'ILLOGICAL STATE FOR EVENT 8'
  800 CALL RMOVE(NFIND(1,1,1,0,3.,0.),1,ATRIB)
      CALL SCHDL(3,ATRIB(2),ATRIB)
      RETURN
C************************************************************
C                                                           *
C     EVENT 9 SIGNALS THAT THE DESIRED INPUT PRODUCTION     *
C     HAS BEEN MET, THE SWITCH IS SET TO TRUE SO THAT       *
C     THE NEXT BIN EMPTY WILL SIGNAL THE END OF THE         *
C     SIMULATION.                                           *
C                                                           *
C************************************************************
    9 STP=.TRUE.
      RETURN
      END
C************************************************************
C                                                           *
C     SUBROUTINE INTLC INITIALLIZES THE SYSTEM BY           *
C     STARTING ALL LINES OPERATING, THE UPPER/LOWER         *
C     BOUNDS ON THE BIN, SCHEDULES THE FIRST BREAK-         *
C     DOWNS, AND STARTS THE BIN EMPTY.                      *
C                                                           *
C************************************************************
      SUBROUTINE INTLC
      COMMON /SCOM1/ ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA,
     1       MSTOP, NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(100),
     2       SSL(100), TNEXT, TNOW, XX(100)
      LOGICAL*4 BIN,BOUT,PIN,POUT,STP,BPERT
      COMMON /USER/ BIN,BOUT,PIN,POUT,STP,BPERT
      BIN=.TRUE.
      BOUT=.TRUE.
      PIN=.TRUE.
      POUT=.TRUE.
      BPERT=.FALSE.
      STP=.FALSE.
      XX(12)=.98*XX(11)
      XX(13)=.02*XX(11)
      XX(14)=0.0
      CALL SCHDL(1,RLOGN(XX(3),XX(4),2),ATRIB)
      CALL SCHDL(3,RLOGN(XX(7),XX(8),4),ATRIB)
      SS(1)=0.
      SS(2)=0.
      SS(3)=0.
      RETURN
```

```
      END
C************************************************************
C                                                           *
C     THE RESULTS ARE WRITTEN OUT                           *
C                                                           *
C************************************************************
      SUBROUTINE OTPUT
      CHARACTER*1 D
      COMMON /SCOM1/ ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA,
     1       MSTOP, NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(100),
     2       SSL(100), TNEXT, TNOW, XX(100)
C************************************************************
C                                                           *
C     THE OUTPUT FILE IS CHECKED SO THAT MORE THAN          *
C     ONE RECORD MAY BE WRITTEN,I.E. MULTIPLE RUNS ARE      *
C     POSSIBLE. COMMENTED OUT.  NO LONGER REQUIRED.         *
C                                                           *
C************************************************************
C   1 READ(8,3,END=2)D
C   3 FORMAT(A1)
C     GO TO 1
C   2 BACKSPACE 8
      WRITE(8,100) XX(11)
  100 FORMAT(' ','BINSIZE = ',F8.1,'  M**3')
      WRITE(8,105) XX(1)
  105 FORMAT(' ','INPUT RATE = ',F8.1,' M**3/HR')
      WRITE(8,110) XX(2)
  110 FORMAT(' ','OUTPUT RATE = ',F8.1,' M**3/HR')
      WRITE(8,115) XX(20)
  115 FORMAT(' ','TARGET PRODUCTION ',F12.1,' M**3')
      WRITE(8,120) XX(20)/TNOW
  120 FORMAT(' ','PRODUCTION RATE ACHIEVED = ',F8.1,' M**3/HR')
      TN = 1.E4*TNOW*TNOW
      WRITE(8,125) XX(20)*XX(14)/TN
  125 FORMAT(' ','DIFF PROD WRT BIN VOL = ',G20.8,' M**3/HR (M**3 BIN)
      RETURN
      END
```

SLAM control code for bin perturbation analysis.

```
GEN,W H GRIFFIN,DIS EVENT PERTB ANAL,6/25/86,1,N,N,Y,N,N,72;
;
;      LIM SETS THE LARGEST FILE TO 1, THE MAXIMUM
;      NUMBER OF ATTRIBUTES TO 2 AND THE MAXIMUM
;      NUMBER OF ENTRIES TO 20.
;
LIM,1,2,20;
;
;      SEEDS INITIALLIZES THE RANDOM NUMBER SEED,
;      CON SETS THE NUMBER OF DIFFERENTIAL EQUATIONS
;      TO 0, THE NUMBER OF STATE EQUATIONS TO 1, THE
;      MINIMUM STEP SIZE BETWEEN UP-DATES AS .001 AND
;      THE MAXIMUM STEP SIZE AS 5.
;
SEEDS,1057751841(1);
SEEDS,1006701355(2);
SEEDS,984182907(3);
SEEDS,1475247003(4);
CON,0,2,.001,5.;
TIMST,SS(1),BIN LEVEL;
TIMST,XX(14),BIN PERT;
;
;      TIMST KEEPS TIME PERSISTENT STATISTICS ON
;      THE BIN LEVEL AND OUTPUT RATE.
;
;TIMST,SS(2),OUTPUT RATE;
;
;      INTLC INITIALLIZES THE OUTPUT/INPUT RATES, THE
;      OPERATING PARAMETERS AND THE BIN SIZE.
;      SEVENT CONTROLS THE BIN VALVE, I.E. SEVENT 5
;      WILL BE CALLED WHEN SS(1) CROSSES THE VALUE
;      SET BY XX(11) IN THE POSITIVE DIRECTION BY
;      A TOLERANCE OF .1.
;
INTLC,XX(1)=1400.,XX(2)=1400.,XX(3)=5.,XX(4)=1.,XX(5)=2.,XX(6)=.3;
INTLC,XX(7)=5.,XX(8)=1.,XX(9)=2.,XX(10)=.3,XX(11)=500,XX(20)=1.E7;
SEVENT,5,SS(1),XP,XX(11),.1;
SEVENT,6,SS(1),XN,XX(12),.1;
SEVENT,7,SS(1),XN,0.,.1;
SEVENT,8,SS(1),XP,XX(13),.1;
```

```
SEVENT,9,SS(2),XP,XX(20),.1;
;      INIT SETS THE RUN START TIME AND LENGTH
;      SIMULATE THEN BEGINS THE RUN, WHEN THE
;      RUN IS COMPLETE, A NEW SEED WILL BE SET
;      A NEW RUN BEGUN
;
INIT,0,15000;
SIMULATE;
```

The random seeds for the runs are omitted for brevity.

```
FIN;
```

SAMPLE OUTPUT FOR BIN PERTURBATION RUN

```
 INPUT RATE =   1400.0 M**3/HR
OUTPUT RATE =   1400.0 M**3/HR
BIN VOLUME =   1000.0 M**3
TARGET PRODUCTION   10000000.0 M**3
PRODUCTION RATE ACHIEVED =     854.4 M**3/HR
DIFF PROD WRT BIN VOL =   .59028800E-01 M**3/HR (M**3 BIN)
INPUT RATE =   1400.0 M**3/HR
OUTPUT RATE =   1400.0 M**3/HR
BIN VOLUME =   1000.0 M**3
TARGET PRODUCTION   10000000.0 M**3
PRODUCTION RATE ACHIEVED =     855.6 M**3/HR
DIFF PROD WRT BIN VOL =   .57567910E-01 M**3/HR (M**3 BIN)
INPUT RATE =   1400.0 M**3/HR
OUTPUT RATE =   1400.0 M**3/HR
BIN VOLUME =   1000.0 M**3
TARGET PRODUCTION   10000000.0 M**3
PRODUCTION RATE ACHIEVED =     858.0 M**3/HR
DIFF PROD WRT BIN VOL =   .57635130E-01 M**3/HR (M**3 BIN)
```

To collect the bin perturbation estimate at each bounce, change the calculation for XX(14) as shown below (see label 702):

```
     XX(14) = XX(14) + 1
     XX(21) = (SS(3)*(*XX(14)/XX(2)))/(TNOW*TNOW)
     CALL COLCT(XX(21),1)
     WRITE(11,991) XX(21)
991  FORMAT(' ',G20.8)
```

Now change the write statement in OTPUT for the differential with respect to production (see label 125):

```
     WRITE(8,125) XX(20)*((XX(14)/1400.)/TN))
```

Any other changes, such as opening and closing the unit to write to, also have to be done.

When running BPA initiallize SS(1), the bin, to zero, for base simulation case, initiallize bin to one half full.
All other changes are documented inline.

FORTRAN subroutines for no bin simulation.

```
C FOR DISCRETE EVENT PERTURBATION ANALYSIS ON A SYSTEM OF ONE
C  CONTINUOUS INPUT THAT HAS A SINGLE RANDOM FAILURE MODE  AND
C ONE OUTPUT THAT HAS THE SAME MEAN AS THE INPUT BUT MAY HAVE
C A DIFFERENT RANDOM DISTRIBUTION OF FAILURES THAN THE INPUT.
C
C EVENT CODES:
C    1=INPUT PROCESS HAS STOPPED OPERATING DUE TO BREAKDOWN.
C    2=INPUT BREAKDOWN HAS BEEN REPAIRED.
C    3=OUTPUT PROCESS HAS STOPPED OPERATING DUE TO BREAKDOWN.
C    4=OUTPUT BREAKDOWN HAS BEEN REPAIRED.
C
C   FILES USED:
C         1=FILE TO HOLD REMAINING TIME ON RUNNING TIME TO FAILURE FOR
C          INPUT AND OUTPUT PROCESSES THAT ARE CONSTRAINED BY
C          OUTPUT OR INPUT FAILURE REPAIR CONDITIONS.
C
C      NCLNR=CALENDAR FILE HOLDING FUTURE DISCRETE EVENTS 1,2,3,4.
C
C  ATTRIBUTE CODES:
C         1=EVENT CODE IN FILE 1.
C         2=TIME REMAINING OF RUNNING TIME TILL EVENT WILL OCCUR.
C         3=EVENT CODE IN FILE NCLNR.
C         4=EVENT TIME IN FILE NCLNR.
C
C  COMMON VARIBLES (SCOM1)
C         XX(1)=POSSIBLE INPUT-OUTPUT RATE
C         XX(2)=ACHIEVED OUTPUT RATE
C         XX(3)=MEAN OPERATING TIME TO FAILURE OF INPUT
C         XX(4)=ST. DEV. OPERATING TIME TO FAILURE OF INPUT
C         XX(5)=MEAN TIME TO REPAIR INPUT FAILURE
C         XX(6)=ST. DEV. TIME TO REPAIR INPUT FAILURE
C         XX(7)=MEAN OPERATING TIME TO FAILURE OF OUTPUT
C         XX(8)=ST. DEV. OPERATING TIME TO FAILURE OF OUTPUT
C         XX(9)=MEAN TIME TO REPAIR OUTPUT FAILURE
C        XX(10)=ST. DEV. TIME TO REPAIR OUTPUT FAILURE
C
C
C     PROGRAM MAIN
C********************************************************
```

```
      DIMENSION NSET(200)
      COMMON /SCOM1/ ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA,
     1       MSTOP, NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(100),
     2       SSL(100), TNEXT, TNOW, XX(100)
      COMMON QSET(200)
      EQUIVALENCE (NSET(1),QSET(1))
      NNSET = 200
      NCRDR = 5
      NPRNT = 6
      NTAPE = 7
      CALL SLAM
      STOP
      END
      SUBROUTINE EVENT(IX)
      COMMON /SCOM1/ ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA,
     1       MSTOP, NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(100),
     2       SSL(100), TNEXT, TNOW, XX(100)
      GO TO(1,2,3,4),IX
      PAUSE 'INVALID EVENT CODE'
    1 XX(2)=0.
      CALL SCHDL(2,RLOGN(XX(5),XX(6),1),ATRIB)
      CALL RMOVE(NFIND(1,NCLNR,3,0,3.,0.),NCLNR,ATRIB)
      ATRIB(2)=ATRIB(4)-TNOW
      ATRIB(1)=3.
      CALL FILEM(1,ATRIB)
      RETURN
    2 RTIM=RLOGN(XX(3),XX(4),2)
      CALL RMOVE(1,1,ATRIB)
      CALL SCHDL(3,ATRIB(2),ATRIB)
      CALL SCHDL(1,RTIM,ATRIB)
      XX(2)=XX(1)
      RETURN
    3 XX(2)=0.
      CALL SCHDL(4,RLOGN(XX(9),XX(10),3),ATRIB)
      CALL RMOVE(NFIND(1,NCLNR,3,0,1.,0.),NCLNR,ATRIB)
      ATRIB(2)=ATRIB(4)-TNOW
      ATRIB(1)=1.
      CALL FILEM(1,ATRIB)
      RETURN
```

```
4 RTIM=RLOGN(XX(7),XX(8),4)
  CALL RMOVE(1,1,ATRIB)
  CALL SCHDL(1,ATRIB(2),ATRIB)
  CALL SCHDL(3,RTIM,ATRIB)
  XX(2)=XX(1)
  RETURN
  END
  SUBROUTINE INTLC
  COMMON /SCOM1/ ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA,
 1       MSTOP, NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(100),
 2       SSL(100), TNEXT, TNOW, XX(100)
  XX(2)=XX(1)
  CALL SCHDL(1,RLOGN(XX(3),XX(4),2),ATRIB)
  CALL SCHDL(3,RLOGN(XX(7),XX(8),4),ATRIB)
  RETURN
  END
  SUBROUTINE OTPUT

  COMMON /SCOM1/ ATRIB(100), DD(100), DDL(100), DTNOW, II, MFA,
 1       MSTOP, NCLNR, NCRDR, NPRNT, NNRUN, NNSET, NTAPE, SS(100),
 2       SSL(100), TNEXT, TNOW, XX(100)
  WRITE(8,*)XX(1),TTAVG(1)
  RETURN
  END
```

.ation.

```
IAL,6/25/86,24,Y,N,Y,N,Y,72;

 1,XX(5)=2.,XX(6)=.3,XX(7)=5.,XX(8)=.1;
```

SAMPLE OUTPUT FOR 0 BIN RUN:

```
000.0000000       782.7802000
000.0000000       773.1516000
000.0000000       783.3796000
     :                 :
     :                 :
```

SAMPLE OUTPUT FOR SIMULATION RUN, 5.0 CASE

```
  50.0000000      793.1085000
  50.0000000      778.0775000
     :                 :
     :                 :
1000.0000000      849.6597000
1000.0000000      856.5233000
     :                 :
     :                 :
```

Appendix C

Contents of Appendix C:

```
STEPWISE REGRESSION
===================

Screen Output will be written to file 2R.OUT

BMDP Instructions will be read from the keyboard


BMDP2R - STEPWISE REGRESSION

Copyright 1977, 1979, 1981, 1982, 1983, 1985, 1987, 1988
         BMDP Statistical Software, Inc.

         BMDP Statistical Software, Inc.| Statistical Software, Ltd.
         1440 Sepulveda Blvd            | Cork Technology Park, Model Farm Rd
         Los Angeles, CA 90025 USA      | Cork, Ireland
                                        |
         Phone (213) 479-7799           | Phone +353 21 542722
         Fax   (213) 312-0161           | Fax   +353 21 542822
         Telex 4972934 BMDP UI          | Telex 75659 SSWL EI

Version: 1988    (IBM PC/DOS)

Manual:  BMDP Manual Vol. 1 (August, 1988); Vol. 2 (December, 1988).
         Use 1983 or 1985 edition until 1988 becomes available.

Digest:  BMDP User's Digest (4th edition), plus addendum.

Updates: State NEWS. in the PRINT paragraph for summary of new features.

08/17/89          AT 17:35:29
/PROB TITLE IS 'STEPWISE REGRESSION OF MID CASE DATA'.
/INPUT FILE = 'MTFR.DAT'.
       VARIABLES = 2.
       FORMAT = FREE.
/VARIABLE NAMES ARE BIN, PROD, CRTB, RTB, SQB, CBB.
          ADD = 4.
/TRANSFORM  CRTB = BIN**(1.0/3.0).
            RTB = SQRT(BIN).
            SQB = BIN * BIN.
            CBB = (SQB*BIN)/1E10.
```

BMDP control code, cont'd next page

```
/REGRESS DEPENDENT = PROD.
/PLOT GRAPH = HIRES.
      RESIDUAL.
      NORMAL.
      FILE = 'MID2R.PLT'.
/END
```

BMDP control code

```
STEP NO.    0
---------------

STD. ERROR OF EST.     81.3205

ANALYSIS OF VARIANCE
                      SUM OF SQUARES      DF    MEAN SQUARE
         RESIDUAL     1897938.0          287   6613.026

       VARIABLES IN EQUATION                      VARIABLES NOT IN EQUATION
--------------------------------------------  --------------------------------
                    STD.ERR          F                 PARTIAL         F
VARIABLE   COEFF.   OF COEFF  TOL. REMOVE(L)  VARIABLE  CORR.    TOL.  ENTER(L)
--------------------------------------------  --------------------------------
(CONSTANT 862.0556)
                                              BIN       0.9610 1.0000 3449.53(1)
                                              CRTB      0.9514 1.0000 2730.16(1)
                                              RTB       0.9803 1.0000 7062.98(1)
                                              SQB       0.8628 1.0000  833.18(1)
                                              CBB       0.7760 1.0000  433.04(1)


STEP NO.    1
---------------
VARIABLE ENTERED     4 RTB

MULTIPLE R              0.9803
MULTIPLE R-SQUARE       0.9611
ADJUSTED R-SQUARE       0.9609

STD. ERROR OF EST.     16.0704

ANALYSIS OF VARIANCE
                      SUM OF SQUARES      DF    MEAN SQUARE       F RATIO
         REGRESSION   1824077.0            1   1824077.          7062.99
         RESIDUAL     73861.880          286   258.2583
```

```
      VARIABLES IN EQUATION                         VARIABLES NOT IN EQUATION
-------------------------------------------  ------------------------------
                    STD.ERR         F                  PARTIAL        F
VARIABLE   COEFF.   OF COEFF  TOL. REMOVE(L) VARIABLE  CORR.    TOL.  ENTER(L)
-------------------------------------------  ------------------------------
(CONSTANT 756.3961)
RTB         3.5619   0.0424 1.0000 7062.99(1) BIN       0.2074 0.0589   12.81(1)
                                              CRTB     -0.4757 0.0272   83.37(1)
                                              SQB      -0.0830 0.2118    1.98(1)
                                              CBB      -0.2264 0.3312   15.40(1)


STEP NO.    2
--------------
VARIABLE ENTERED    3 CRTB

MULTIPLE R            0.9848
MULTIPLE R-SQUARE     0.9699
ADJUSTED R-SQUARE     0.9697

STD. ERROR OF EST.   14.1602

ANALYSIS OF VARIANCE
                  SUM OF SQUARES     DF    MEAN SQUARE        F RATIO
        REGRESSION   1840792.0          2   920396.2           4590.22
        RESIDUAL     57146.050        285   200.5125

      VARIABLES IN EQUATION                         VARIABLES NOT IN EQUATION
-------------------------------------------  ------------------------------
                    STD.ERR         F                  PARTIAL        F
VARIABLE   COEFF.   OF COEFF  TOL. REMOVE(L) VARIABLE  CORR.    TOL.  ENTER(L)
-------------------------------------------  ------------------------------
(CONSTANT 777.1913)
CRTB       -9.1811   1.0055 0.0272   83.37(1) BIN      -0.5237 0.0127  107.34(1)
RTB         5.5992   0.2262 0.0272  612.54(1) SQB      -0.6899 0.1032  257.90(1)
                                              CBB      -0.7489 0.2057  362.76(1)

NOTE THAT VARIABLE     1 BIN       CAN NOT BE ENTERED BECAUSE ITS ENTRY
WOULD LOWER THE TOLERANCE OF VARIABLE     3 CRTB      BELOW THE TOLERANCE LIMIT.
```

STEP NO. 3

VARIABLE ENTERED 6 CBB

MULTIPLE R	0.9934
MULTIPLE R-SQUARE	0.9868
ADJUSTED R-SQUARE	0.9866

STD. ERROR OF EST. 9.3999

ANALYSIS OF VARIANCE

	SUM OF SQUARES	DF	MEAN SQUARE	F RATIO
REGRESSION	1872845.0	3	624281.6	7065.37
RESIDUAL	25093.640	284	88.35790	

VARIABLES IN EQUATION					VARIABLES NOT IN EQUATION			
VARIABLE	COEFF.	STD.ERR OF COEFF	TOL.	F REMOVE(L)	VARIABLE	PARTIAL CORR.	TOL.	F ENTER(L)
(CONSTANT 786.2730)								
CRTB	-19.1092	0.8469	0.0169	509.09(1)	BIN	0.5266	0.0024	108.62(1)
RTB	8.6536	0.2197	0.0127	1551.31(1)	SQB	0.4286	0.0030	63.67(1)
CBB	-6.3133	0.3315	0.2057	362.76(1)				

***** F LEVELS(4.000, 3.900) OR TOLERANCE INSUFFICIENT FOR FURTHER STEPPING

STEPWISE REGRESSION COEFFICIENTS

VARIABLES STEP	0 Y-INTCPT	1 BIN	3 CRTB	4 RTB	5 SQB
0	862.0556*	0.0472	15.3643	3.5619	0.0000
1	756.3961*	0.0083	-9.1811	3.5619*	0.0000
2	777.1913*	-0.0396	-9.1811*	5.5992*	0.0000
3	786.2730*	0.0609	-19.1092*	8.6536*	0.0000

NOTE - 1) REGRESSION COEFFICIENTS FOR VARIABLES IN THE EQUATION ARE INDICATED BY AN ASTERISK
2) THE REMAINING COEFFICIENTS ARE THOSE WHICH WOULD BE OBTAINED IF THAT VARIABLE WERE TO ENTER IN THE NEXT STEP

STEPWISE REGRESSION COEFFICIENTS

VARIABLES	6 CBB
STEP	
0	17.0998
1	-1.7101
2	-6.3133
3	-6.3133*

NOTE - 1) REGRESSION COEFFICIENTS FOR VARIABLES IN THE EQUATION ARE INDICATED BY AN ASTERISK
2) THE REMAINING COEFFICIENTS ARE THOSE WHICH WOULD BE OBTAINED IF THAT VARIABLE WERE TO ENTER IN THE NEXT STEP

SUMMARY TABLE

STEP	VARIABLE		MULTIPLE		CHANGE	F TO	F TO	NO.OF VAR.
NO.	ENTERED	REMOVED	R	RSQ	IN RSQ	ENTER	REMOVE	INCLUDED
1	4 RTB		0.9803	0.9611	0.9611	7062.99		1
2	3 CRTB		0.9848	0.9699	0.0088	83.37		2
3	6 CBB		0.9934	0.9868	0.0169	362.76		3

NUMBER OF INTEGER WORDS OF STORAGE USED IN PRECEDING PROBLEM 2502
CPU TIME USED 409.520 SECONDS

BMDP2R - STEPWISE REGRESSION

08/17/89 AT 17:42:18

```
EXPONENTIAL REGRESSION
======================

Screen Output will be written to file EXP.OUT

BMDP Instructions will be read from the keyboard

BMDPAR--DERIVATIVE-FREE NONLINEAR REGRESSION

Copyright 1977, 1979, 1981, 1982, 1983, 1985, 1987, 1988
        BMDP Statistical Software, Inc.

        BMDP Statistical Software, Inc.| Statistical Software, Ltd.
        1440 Sepulveda Blvd            | Cork Technology Park, Model Farm Rd
        Los Angeles, CA 90025 USA      | Cork, Ireland
                                       |
        Phone (213) 479-7799           | Phone +353 21 542722
        Fax   (213) 312-0161           | Fax   +353 21 542822
        Telex 4972934 BMDP UI          | Telex 75659 SSWL EI

Version: 1988    (IBM PC/DOS)

Manual:  BMDP Manual Vol. 1 (August, 1988); Vol. 2 (December, 1988).
         Use 1983 or 1985 edition until 1988 becomes available.

Digest:  BMDP User's Digest (4th edition), plus addendum.

Updates: State NEWS. in the PRINT paragraph for summary of new features.

08/18/89          AT 13:15:26
/PROB TITLE IS 'EXPONENTIAL REGRESSION OF MID CASE DATA'.
/INPUT FILE = 'MTFR.DAT'.
      VARIABLES = 2.
      FORMAT = FREE.
/VARIABLE NAMES = BIN, PROD.
/REGRESS  DEPENDENT = PROD.
          INDEPENDENT = BIN.
          NUMBER = 1.
          PARAMETERS = 3,
```

BMDP control code, cont'd next page

```
/PARAMETER INIT = -240.00, -0.0005, 1020.0.
          MAXIMUM = -200.00, 0.005, 1100.0.
          MINIMUM = -350.00, -0.5, 1015.0.
/PLOT GRAPH = HIRES.
      RESIDUAL.
      NORMAL.
      FILE = 'MIDEXP.PLT'.
/END
```

BMDP control code

PARAMETERS TO BE ESTIMATED

	P1	P2	P3
MINIMUM	-350.000	-0.500000	1015.00
MAXIMUM	-200.000	0.500000E-02	1100.00
INITIAL	-240.000000	-0.000500	1020.000000

USING THE ABOVE SPECIFICATIONS THIS PROGRAM COULD USE UP TO 1040 CASES IF NO BOUNDARIES ARE ENCOUNTERED, AND 1035 CASES IF ALL BOUNDARIES ARE ENCOUNTERED SIMULTANEOUSLY.

NUMBER OF CASES READ. 288

VARIABLE NO.	NAME	MEAN	STANDARD DEVIATION	MINIMUM	MAXIMUM
1	BIN	1379.166000	1654.321000	0.000000	5000.000000
2	PROD	862.055600	81.320510	756.577600	1001.827000

ITER. NO.	INCR. HALV.	RESIDUAL SUM OF SQUARES	PARAMETERS P1	P2	P3
0	0	581377.55521088	-240.000000	-0.000500	1060.000000
0	0	80257.20417641	-264.000000	-0.000500	1020.000000

0	0	48602.60827384	-240.000000	-0.000550	1020.000000
0	0	31807.18621137	-240.000000	-0.000500	1020.000000
1	0	26822.34886517	-249.600692	-0.000442	1026.979204
2	0	26440.05982073	-248.669010	-0.000459	1025.491303
3	4	26436.53673572	-249.039246	-0.000458	1025.909396
4	0	26429.14343523	-249.925988	-0.000454	1026.703357
5	0	26427.69492654	-249.826999	-0.000455	1026.684377
6	0	26427.62939137	-249.846236	-0.000454	1026.738597
7	2	26427.62521977	-249.840345	-0.000454	1026.733907
8	0	26427.58374156	-249.778641	-0.000455	1026.660819
9	2	26427.58372350	-249.779057	-0.000455	1026.661191
10	2	26427.58369890	-249.779208	-0.000455	1026.661592

THE RESIDUAL SUM OF SQUARES (= 26427.6) WAS SMALLEST WITH THE FOLLOWING PARAMETER VALUES

PARAMETER	ESTIMATE	ASYMPTOTIC STANDARD DEVIATION	COEFFICIENT OF VARIATION
P1	-249.779208	3.258098	-0.013044
P2	-0.000455	0.000016	-0.034263
P3	1026.661592	3.571209	0.003478

ESTIMATE OF ASYMPTOTIC CORRELATION MATRIX

		P1	P2	P3
		1	2	3
P1	1	1.0000		
P2	2	-0.8420	1.0000	
P3	3	-0.9602	0.9324	1.0000

VARIABLE	MEAN	VARIANCE	ESTIMATED MEAN SQUARED ERROR	PSEUDO * R-SQUARE
PROD	862.055600	6613.026000	92.728360	0.9861
		DF= 287	DF= 285	

*NOTE--PSEUDO R-SQUARE=1.0 MINUS THE RATIO OF (WEIGHTED) RESIDUAL SS TO (N-1) TIMES (WEIGHTED) VARIANCE. WHEN THE NONLINEAR MODEL FITS THE DATA LESS WELL THAN THE MEAN, THE PSEUDO R-SQUARE WILL BE NEGATIVE.

```
NON-LINEAR REGRESSION
=====================

Screen Output will be written to file AR.OUT

BMDP Instructions will be read from the keyboard

BMDPAR--DERIVATIVE-FREE NONLINEAR REGRESSION

Copyright 1977, 1979, 1981, 1982, 1983, 1985, 1987, 1988
          BMDP Statistical Software, Inc.

          BMDP Statistical Software, Inc.| Statistical Software, Ltd.
          1440 Sepulveda Blvd            | Cork Technology Park, Model Farm Rd
          Los Angeles, CA 90025 USA      | Cork, Ireland
                                         |
          Phone (213) 479-7799           | Phone +353 21 542722
          Fax   (213) 312-0161           | Fax   +353 21 542822
          Telex 4972934 BMDP UI          | Telex 75659 SSWL EI

Version: 1988    (IBM PC/DOS)

Manual:  BMDP Manual Vol. 1 (August, 1988); Vol. 2 (December, 1988).
         Use 1983 or 1985 edition until 1988 becomes available.

Digest:  BMDP User's Digest (4th edition), plus addendum.

Updates: State NEWS. in the PRINT paragraph for summary of new features.

08/17/89            AT 18:05:55
/PROB TITLE IS 'AR REGRESSION OF MID CASE DATA'.
/INPUT FILE = 'MTFR.DAT'.
       VARIABLES = 2.
       FORMAT = FREE.
/VARIABLE NAMES = BIN, PROD.
/REGRESS  DEPENDENT = PROD.
          INDEPENDENT = BIN.
          PARAMETERS = 4.
/FUNCTION F = B0+B1*SQRT(BIN)-B2*(BIN**(1.0/3.0))-B3*((BIN**3.0)/1.0E10).
```

BMDP control code, cont'd next page

```
/PARAMETER NAME = B0, B1, B2, B3.
          INIT = 786.27, 8.654, 19.11, 6.313.
          MAXIMUM = 790.0.
          MINIMUM = 780.0.
/PLOT GRAPH = HIRES.
     RESIDUAL.
     NORMAL.
     FILE = 'MIDEXP.PLT'.
/END
```

BMDP control code

```
PARAMETERS TO BE ESTIMATED

                  B0              B1              B2
MINIMUM        780.000     -0.324519E+33   -0.324519E+33
MAXIMUM        790.000      0.324519E+33    0.324519E+33
INITIAL         786.270020      8.654000       19.110001

                  B3
MINIMUM      -0.324519E+33
MAXIMUM       0.324519E+33
INITIAL          6.313000


  USING THE ABOVE SPECIFICATIONS THIS PROGRAM COULD USE UP TO  907 CASES
  IF NO BOUNDARIES ARE ENCOUNTERED, AND   904 CASES IF ALL BOUNDARIES ARE
  ENCOUNTERED SIMULTANEOUSLY.

NUMBER OF CASES READ. . . . . . . . . . . . . .    288

 VARIABLE                       STANDARD
 NO. NAME          MEAN         DEVIATION     MINIMUM       MAXIMUM

  1  BIN        1379.166000   1654.321000     0.000000   5000.000000
  2  PROD        862.055600     81.320510   756.577600   1001.827000

ITER. INCR.  RESIDUAL SUM     PARAMETERS
 NO.  HALV.   OF SQUARES         B0            B1            B2
                                 B3

  0     0   322699.85293120   786.270020     9.519400     19.110001
                                6.313000
```

```
  0    0  133924.22309661      786.270020         8.654000          21.021001
                                 6.313000
  0    0   27045.67551593      786.270020         8.654000          19.110001
                                 6.944300
  0    0   26100.41691935      788.135010         8.654000          19.110001
                                 6.313000
  0    0   25094.53046037      786.270020         8.654000          19.110001
                                 6.313000
  1    0   25094.51150890      786.272300         8.653443          19.108789
                                 6.313187
  2    0   25094.51150890      786.272300         8.653443          19.108789
                                 6.313187
  3    2   25094.51150890      786.272300         8.653443          19.108789
                                 6.313187
  4    2   25094.51150890      786.272300         8.653443          19.108789
                                 6.313187
  5    5   25094.51150890      786.272300         8.653443          19.108789
                                 6.313187

***  WARNING: THE MINIMUM SS OCCURRED AT ITERATION    4.
THE VARIANCES AND COVARIANCES ARE BASED ON THE FINAL ITERATION.
TO OBTAIN VARIANCES AND COVARIANCES CORRESPONDING TO MINIMUM SS
RERUN SAME COMMANDS USING ITER=    4. IN THE /REGRESS PARAGRAPH.

THE RESIDUAL SUM OF SQUARES ( =      25094.5      ) WAS SMALLEST WITH THE
FOLLOWING PARAMETER VALUES

PARAMETER        ESTIMATE          ASYMPTOTIC            COEFFICIENT
                                STANDARD DEVIATION       OF VARIATION

B0              786.272300           1.833260              0.002332
B1                8.653443           0.219709              0.025390
B2               19.108789           0.846931              0.044322
B3                6.313187           0.331478              0.052506

ESTIMATE OF ASYMPTOTIC CORRELATION MATRIX


               B0          B1          B2          B3

                   1           2           3           4
B0         1     1.0000
```

B1	2	0.7006	1.0000			
B2	3	0.8101	0.9806	1.0000		
B3	4	0.2601	0.7299	0.6155	1.0000	

VARIABLE	MEAN	VARIANCE	ESTIMATED MEAN SQUARED ERROR	PSEUDO * R-SQUARE
PROD	862.055600	6613.026000	88.360950	0.9868
		DF= 287	DF= 284	

*NOTE--PSEUDO R-SQUARE=1.0 MINUS THE RATIO OF (WEIGHTED) RESIDUAL SS TO (N-1) TIMES (WEIGHTED) VARIANCE. WHEN THE NONLINEAR MODEL FITS THE DATA LESS WELL THAN THE MEAN, THE PSEUDO R-SQUARE WILL BE NEGATIVE.

```
REGRESSION OF RANDOM SAMPLE OF SIMULATION OUTPUT.
===============================================

Screen Output will be written to file RND80.OUT

BMDP Instructions will be read from the keyboard

BMDPAR--DERIVATIVE-FREE NONLINEAR REGRESSION

Copyright 1977, 1979, 1981, 1982, 1983, 1985, 1987, 1988
        BMDP Statistical Software, Inc.

        BMDP Statistical Software, Inc.| Statistical Software, Ltd.
        1440 Sepulveda Blvd            | Cork Technology Park, Model Farm Rd
        Los Angeles, CA 90025 USA      | Cork, Ireland
                                       |
        Phone (213) 479-7799           | Phone +353 21 542722
        Fax   (213) 312-0161           | Fax   +353 21 542822
        Telex 4972934 BMDP UI          | Telex 75659 SSWL EI

Version: 1988    (IBM PC/DOS)

Manual:  BMDP Manual Vol. 1 (August, 1988); Vol. 2 (December, 1988).
         Use 1983 or 1985 edition until 1988 becomes available.

Digest:  BMDP User's Digest (4th edition), plus addendum.

Updates: State NEWS. in the PRINT paragraph for summary of new features.

09/14/89          AT 16:30:15
/PROB TITLE IS '50% SUBSAMPLE OF EXPONENTIAL REGRESSION OF MID CASE DATA'.
/INPUT FILE = 'MTFR.DAT'.
       VARIABLES = 2.
       FORMAT = FREE.
/VARIABLE NAMES = BIN, PROD.
/TRANSFORM
         IF (RNDU(7892911) LT .50) THEN USE = 0.
/REGRESS  DEPENDENT = PROD.
          INDEPENDENT = BIN.
```

```
         NUMBER = 1.
         PARAMETERS = 3,
/PARAMETER INIT = -240.00, -0.0005, 1020.0.
          MAXIMUM = -200.00, 0.005, 1100.0.
          MINIMUM = -350.00, -0.5, 1015.0.
/PLOT GRAPH = HIRES.
      RESIDUAL.
      NORMAL.
      FILE = 'JUNK.PLT'.
/END
```

REGRESSION TITLE
50% SUBSAMPLE OF EXPONENTIAL REGRESSION OF MID CASE DATA

PARAMETERS TO BE ESTIMATED

	P1	P2	P3
MINIMUM	-350.000	-0.500000	1015.00
MAXIMUM	-200.000	0.500000E-02	1100.00
INITIAL	-240.000000	-0.000500	1020.000000

USING THE ABOVE SPECIFICATIONS THIS PROGRAM COULD USE UP TO 1037 CASES IF NO BOUNDARIES ARE ENCOUNTERED, AND 1032 CASES IF ALL BOUNDARIES ARE ENCOUNTERED SIMULTANEOUSLY.

NUMBER OF CASES READ.	288
CASES WITH USE SET TO ZERO	145
REMAINING NUMBER OF CASES	143

VARIABLE NO.	NAME	MEAN	STANDARD DEVIATION	MINIMUM	MAXIMUM
1	BIN	1375.525000	1641.741000	0.000000	5000.000000
2	PROD	861.867500	80.573350	760.031300	1001.827000

ITER. NO.	INCR. HALV.	RESIDUAL SUM OF SQUARES	PARAMETERS P1	P2	P3
0	0	298647.31964965	-240.000000	-0.000500	1060.000000

0	0	38047.67220091	-264.000000	-0.000500	1020.000000
0	0	27057.89290648	-240.000000	-0.000550	1020.000000
0	0	17450.58980048	-240.000000	-0.000500	1020.000000
1	0	14055.36064568	-249.999369	-0.000440	1026.826174
2	0	13844.42320444	-248.291995	-0.000459	1024.489706
3	2	13834.01659055	-248.980339	-0.000457	1025.267448
4	0	13826.58640450	-250.377679	-0.000452	1026.562439
5	0	13825.66467619	-250.275641	-0.000452	1026.560071
6	0	13825.64766992	-250.285599	-0.000452	1026.588529
7	2	13825.64756936	-250.285548	-0.000452	1026.588537
8	0	13825.61205538	-250.204729	-0.000452	1026.504030
9	2	13825.61204241	-250.205471	-0.000452	1026.504714
10	2	13825.61203781	-250.205873	-0.000452	1026.505400

THE RESIDUAL SUM OF SQUARES (= 13825.6) WAS SMALLEST WITH THE FOLLOWING PARAMETER VALUES

PARAMETER	ESTIMATE	ASYMPTOTIC STANDARD DEVIATION	COEFFICIENT OF VARIATION
P1	-250.205873	4.725527	-0.018887
P2	-0.000452	0.000022	-0.048278
P3	1026.505400	5.148671	0.005016

ESTIMATE OF ASYMPTOTIC CORRELATION MATRIX

		P1	P2	P3
		1	2	3
P1	1	1.0000		
P2	2	-0.8340	1.0000	
P3	3	-0.9586	0.9288	1.0000

VARIABLE	MEAN	VARIANCE	ESTIMATED MEAN SQUARED ERROR	PSEUDO * R-SQUARE
PROD	861.867500	6492.065000	98.754370	0.9850
		DF= 142	DF= 140	

*NOTE--PSEUDO R-SQUARE=1.0 MINUS THE RATIO OF (WEIGHTED) RESIDUAL SS TO (N-1) TIMES (WEIGHTED) VARIANCE. WHEN THE NONLINEAR MODEL FITS THE DATA LESS WELL THAN THE MEAN, THE PSEUDO R-SQUARE WILL BE NEGATIVE.

CPU TIME USED 178.060 SECONDS

BMDPAR--DERIVATIVE-FREE NONLINEAR REGRESSION

09/14/89 AT 16:33:12

PROGRAM TERMINATED

BIN	PROD
0	782.7802
0	773.1516
0	783.3796
0	791.316
0	789.6464
0	784.4707
0	783.5894
0	756.5776
0	783.0757
0	760.0313
0	789.7604
0	789.7618
0	781.4619
0	786.5802
0	770.7352
0	774.7625
0	776.5016
0	788.0791
0	791.6599
0	781.2452
0	781.101
0	774.7738
0	774.5035
0	794.3915
50	793.1085
50	778.0775
50	787.8795
50	799.8044
50	782.8888
50	789.0178
50	788.2644
50	787.6731
50	775.8854
50	789.4554
50	783.4858
50	785.9789
50	785.9103
50	787.4026
50	778.3536
50	787.7987
50	790.4625
50	792.9443
50	784.584
50	781.152
50	783.539
50	791.2208
50	789.0256
50	796.3348
100	795.1037
100	785.2184
100	798.2462
100	806.5415
100	785.4111
100	784.269
100	793.2587

100	788.6879
100	781.0127
100	794.6133
100	780.0931
100	793.9918
100	785.4764
100	784.6072
100	789.4303
100	790.5417
100	798.9287
100	784.4974
100	782.0414
100	794.989
100	796.5459
100	797.4926
100	794.4417
200	803.0319
200	795.434
200	800.7349
200	806.2567
200	796.9622
200	794.6693
200	807.444
200	797.9331
200	797.2924
200	809.6174
200	797.3965
200	794.7546
200	803.7197
200	787.3024
200	804.8665
200	805.1223
200	800.7303
200	801.5194
200	789.5724
200	792.9995
200	793.8251
200	811.5396
200	812.0254
200	804.2516
300	809.4833
300	801.4246
300	811.0372
300	813.2736
300	810.9973
300	803.2118
300	811.2881
300	797.9594
300	807.07
300	819.3198
300	797.1348
300	798.299
300	806.7531
300	805.7761

300	804.0015
300	810.7507
300	800.8432
300	802.4337
300	800.2136
300	802.4004
300	797.2116
300	812.8367
300	811.395
300	799.0856
400	815.5559
400	807.3851
400	820.2181
400	819.2941
400	815.4408
400	815.582
400	810.6449
400	808.5908
400	827.376
400	821.1628
400	808.3223
400	816.485
400	814.2057
400	804.0566
400	822.082
400	806.4
400	814.8368
400	822.7828
400	815.1268
400	810.4317
400	813.7427
400	828.8154
400	809.2522
400	792.952
500	819.9606
500	813.8812
500	828.3419
500	839.8479
500	812.2861
500	817.4349
500	817.1127
500	814.9158
500	832.2291
500	824.2424
500	829.3828
500	814.1616
500	831.2084
500	814.7324
500	819.9907
500	818.6912
500	809.7858
500	831.8927
500	814.8498

500	815.1536
500	810.4762
500	847.4508
500	829.0208
500	812.1351
1000	849.6597
1000	856.5233
1000	870.7448
1000	862.4464
1000	854.4501
1000	844.4004
1000	863.9838
1000	847.5239
1000	854.4756
1000	848.5557
1000	858.3412
1000	863.2211
1000	859.176
1000	876.7823
1000	862.7052
1000	856.3215
1000	859.074
1000	853.8702
1000	866.1852
1000	854.7687
1000	861.111
1000	859.3833
1000	860.0236
1000	855.0546
2000	928.3699
2000	931.1099
2000	930.4238
2000	942.4889
2000	928.87
2000	931.2272
2000	938.1504
2000	920.9135
2000	933.111
2000	922.7126
2000	933.6776
2000	929.8521
2000	942.2642
2000	924.1339
2000	921.6668
2000	929.6343
2000	938.5659
2000	931.237
2000	933.6493
2000	945.3303
2000	917.0907
2000	922.5084
2000	934.8816
2000	927.9903

3000	968.8617
3000	971.6637
3000	989.4139
3000	989.4406
3000	978.4316
3000	979.2277
3000	965.4062
3000	985.6754
3000	974.1453
3000	977.8297
3000	987.1377
3000	968.8516
3000	985.3895
3000	972.9922
3000	981.6931
3000	971.1444
3000	963.4821
3000	979.3557
3000	976.5243
3000	968.6918
3000	965.0892
3000	975.7776
3000	972.6005
3000	968.0421
4000	986.8931
4000	989.8858
4000	999.069
4000	993.54
4000	994.024
4000	984.9141
4000	977.8368
4000	987.8377
4000	988.4449
4000	987.6884
4000	994.8005
4000	982.8914
4000	991.8788
4000	982.4254
4000	997.5474
4000	983.7496
4000	982.4212
4000	998.0947
4000	992.8417
4000	978.1265
4000	986.9221
4000	983.5797
4000	978.2834
4000	980.0798
5000	988.3937
5000	994.9146
5000	1000.029
5000	994.5615
5000	995.2272

5000	990.9604
5000	986.96
5000	995.9687
5000	993.5308
5000	989.045
5000	995.0168
5000	993.4951
5000	995.041
5000	988.5618
5000	992.9546
5000	987.7101
5000	989.2824
5000	1001.827
5000	992.9337
5000	981.0148
5000	995.6024
5000	988.5024
5000	982.5032
5000	987.3751