LEAD TIME PREDICTION FOR A JOB SHOP

By

WEI-SHIN YU, B.E.(HONS.)

A thesis submitted for the

Degree of Doctor of Philosophy

in

Mechanical Engineering

in the

University of Canterbury,

Christchurch, New Zealand.

MARCH, 1975.

CONTENTS

CHAPTER PAGE

APPENDICES

ABSTRACT

For a job shop, there has been very little research done on the accurate prediction of lead times, despite of the importance of and the advantages to be gained from the ability to predict lead times accurately.

In job shop scheduling using integer programming method, etc., to produce fixed schedules, the lead time estimates can be obtained directly from the schedules.

But none of these methods is of any appreciable use to industry.

Heuristic job shop scheduling, on the other hand, has been implemented and performs satisfactorily.

However, the means of predicting the lead times has to be formulated separately.

This work investigates existing methods of predicting lead times, for a job shop employing heuristic scheduling.

It evaluates their stability, system response rates, and their accuracy under steady state conditions.

Before a method can be implemented for a real life job shop, it must be tested under dynamic conditions extensively, and found to be stable.

The accuracy of the lead time predicted under such condition, must also be acceptable.

Hence, the method with the best accuracy from the evaluation was subjected to such dynamic tests.

The results of the tests showed that this method was stable under all the dynamic conditions tested, and predicted lead time with very good accuracy.

A new version of this method is formulated.

Testing under similar steady state and dynamic conditions, showed that it was superior to the original version.

ACKNOWLEDGEMENTS

The author gratefully acknowledges the following:

Professor H. McCallion, Dean, School of Engineering, and Professor of Technology, Department of Mechanical Engineering, University of Canterbury, the author's supervisor, for his excellent supervision and encouragement throughout this project.

Professor D.C. Stevenson, for placing the facilities of the Department of Mechanical Engineering, University of Canterbury, at the author's disposal.

Todd Motors Limited, for providing financial support during the research.

Mrs P. Thomas, Mrs J. Brown and Mrs J. Ritchie, for their assistance in preparing this thesis.

CHAPTER 1

INTRODUCTION

The success of a manufacturer depends on his ability to produce high quality goods at a competitive price, as well as his ability to keep up with delivery promises.

Not only are customers "becoming increasingly particular about the capability of contracting firms to meet their promised delivery dates"[1], "in some cases, a reliable guarantee of delivery can win an order for a supplier in the face of competition from lower-priced competitors, even where these can demonstrate equal technical competence"[2].

Lateness in the completion of goods not only causes irreparable damage to the customer goodwill, it also results in unwelcome additions to fixed manufacturing expenses, with space, plant and men committed beyond the economic timespan.

Yet manufacturers are far less successful in delivering the goods consistently on time, although they may be very effective in order taking[3].

The situation in batch production of the job shop nature is the worst, due to the inherent complexity arising from the wide variety of product mix and material.

The failure to meet due dates may be caused by the unexpected breakdown of machines or the delay of raw material from suppliers. But more often, it is caused by the lack of a sound basis on which to predict the lead times accurately.

Lead time is defined as the time elapsed between placing an order and taking delivery of the goods.

In batch production, this is very obvious. Some firms estimate lead times by "intuition". Others estimate the lead time of a part as the total operation time multiplied by a constant. Still others allocate a week towards lead time for each operation. The last two methods are invariant with respect to the state of congestion in the machine shop, and are optimistic when the load in the shop is light.

The problem is: how can lead times be predicted accurately, in a job shop in particular?

There has been very little research done on lead time prediction.

In job shop scheduling using integer programming method, branch and bound method or network method, the information on lead times can be obtained directly from the schedules produced. But none of these methods can optimise efficiently large-sized problems, and the reported progress is not sufficient to be of any appreciable use to industry[4].

In heuristic job shop scheduling, where a priority rule is used to select, from a queue, a batch to be loaded onto the corresponding machine, no schedules are produced. Hence a means of predicting lead time has to be formulated. Unlike the integer programming method, the branch and bound method or the network method, the heuristic scheduling system is quite feasible for implementation.

The El Segundo Division of Hughes Aircraft Company, a firm who had a heuristic scheduling system installed in its job shop reported a significant reduction in work-in-progress inventory and the number of orders late, and a significant increase in manpower and machine utilisation[5].

In view of the success achieved by the heuristic scheduling system, this work proposes to investigate the prediction of lead times for a job shop employing such a scheduling system.

Eilon and Hodgson[6] suggested a method of "quoting more realistic lead time", in a paper entitled "Job Shop Scheduling With Due Dates", 1967. But their investigation dealt with a machine shop with only two identical machines. The method derived has very limited application.

At Nottingham University, McCallion, Horsnell, Davies and Brittain[7,8,9] investigated a few methods of lead time prediction, for a job shop with heuristic scheduling. The methods involved "learning", achieved through the use of exponential smoothing. Some degree of success was attained.

The research work presented here, is a continuation of the investigation at Nottingham University. It examined the work done at Nottingham University, and attempted to find better means of predicting job lead times for a job shop with heuristic scheduling.

CHAPTER 2

LITERATURE SURVEY

A survey of the current literature shows that no published work has done much research on the lead time prediction for a job shop with heuristic scheduling.

Eilon and Hodgson (1967)[6], in their investigation into the performance of priority rules in scheduling with due dates, suggested how lead times could be determined more realistically.

They used a job shop consisting of two identical machines operating in parallel. Each job required one operation only.

The lead time of a job was assigned as the process time multiplied by a constant K. For a particular loading, the optimum value, K_o, of K was one that gave zero as the mean of the "missed due date distribution".

The missed due date distribution is a distribution of the amount of time a job is late or early, with respect to the assigned due date. Hence it included jobs which were finished late, as well as those early.

For a particular loading (i.e. average utilisation) of the job shop and a priority rule used, a full scale simulation of the job shop was made for each value of K.

The means of the missed due date distributions obtained were plotted against the K values to give K_o, by interpolation.

The model used was far too simple for the average job shop. Consequently, the result obtained, with regard to lead time prediction, is of limited use.

Davies (1969)[7], in his Ph.D. thesis, investigated two methods of lead time prediction:

(a) The adaptive lead time method and

(b) The adaptive queueing time method.

(a) <u>The Adaptive Lead Time Method</u>

This method was based on the following reasoning:

If a human scheduler loads a part onto a machine shop and it emerges either grossly late or grossly early, he will alter the lead time accordingly.

The method of adjusting the lead times could be formalised and built into a scheduling system.

For this, he used the exponential smoothing prediction formula:

$$\text{new estimated lead time} = (1-\alpha) \times \text{old estimated lead time} + \alpha \times \text{actual manufacturing time}$$

where α is the exponential constant, and $0 \leq \alpha \leq 1$

This method would automatically adjust the lead times to the best value consistent with the machine shop utilisation.

(b) The Adaptive Queueing Time Method

Lead time of a part may be defined as the sum of the service times plus the sum of the queueing times. Assuming the service time distribution of a part is known. Then, the accuracy of the lead time predicted depends largely on the queueing time prediction.

To formalise the queueing time prediction, the exponential smoothing prediction formula was again used.

A scheduling system using such an approach would update the queueing time estimate at a machine group for lead time prediction as soon as a batch was loaded onto a machine of the group. Whereas in the previous approach, it was necessary for a batch to be completely finished, before information could be fed back to modify the lead time estimate. The system response is therefore increased.

The job shop model used by Davies consisted of five functionally different machine groups. Four of the groups had one machine each, and the remaining group had two identical machines.

The heuristic scheduling system he used for the lead time investigation employed the minimum float rule as the queueing discipline.

To obtain a steady initial state of the machine shop, he used a procedure developed by Brittain[9]. This procedure will be described in Chapter 4.

Twenty different parts were produced by the job shop.

Orders for parts were loaded into the job shop on the basis of minimising shortages: Whenever the quantity demanded of a part during the estimated lead time period was expected to exceed the stock level plus the quantity in work in progress, a standard batch of the part was introduced into the job shop.

To vary the utilisation of the job shop the shift length was varied.

To form a basis for comparison, a method with fixed lead times was used.

In this method, the predicted lead time of a part was assigned as the total service time required multiplied by a constant.

The constant used was 2.5.

This method was referred to as the Constant Lead Time Method.

Graphs 2-1 to 2-4 and Tables 2-1 to 2-3 are reproduced from Davies' thesis.

Davies' findings are summarised as below:

(a) The Adaptive Lead Time Method Vs. The Constant Lead Time Method:

With reference to Graphs 2-1 and 2-2:-

Over the range of job shop utilisation investigated, the adaptive lead time method improved the standard deviations of the "lead time error" histograms considerably.

In addition, above 86% shop utilisation, there were varying degrees of reduction in the "objective function",

*TABLE 2-1.

Minimum Float Rule - values of objective function for various utilisations - stock buffer level for each part 2 hours usage per operation on each part

Shift length (hours)	3.0	2.8	2.6	2.5	2.4	2.3	2.2
Utilisation per cent	72.5	77.7	83.0	86.0	89.4	93.0	96.0
Unused Machine Time cost	111.0	84.0	60.0	47.0	34.0	23.0	11.0
Work in progress cost	4.4	4.4	4.1	4.0	3.8	3.6	3.3
Shortage cost	6.8	7.2	9.9	8.1	23.9	58.5	149.3
Total cost (£/wk)	122.2	95.6	74.0	59.1	61.7	85.1	163.6
Standard deviation of lead time error histograms(shifts)	11.39	11.09	10.76	9.63	8.96	6.92	5.16

*Reproduced from Davies's Thesis[7].

*TABLE 2-2.

Minimum Float Rule - objective function versus utilisation - stock buffer level 2 hours usage per operation per part, exponentially smoothed lead time prediction $\alpha = 0.1$.

Shift length (hours)	3.0	2.8	2.6	2.5	2.4	2.3	2.2
Utilisation per cent	71.3	76.2	82.6	85.6	89.5	94.0	-
Unused Machine Time cost	115.0	89.0	61.0	48.0	38.0	20.0	-
Work in progress cost	3.7	3.7	3.7	3.7	3.9	3.9	-
Shortage cost	6.0	6.3	8.4	8.2	13.4	25.5	-
Total cost (£/wk)	124.7	99.0	73.3	59.9	55.3	49.4	-
Standard deviation of lead time error histogram(shifts)	3.18	3.18	3.47	4.58	3.31	3.63	-

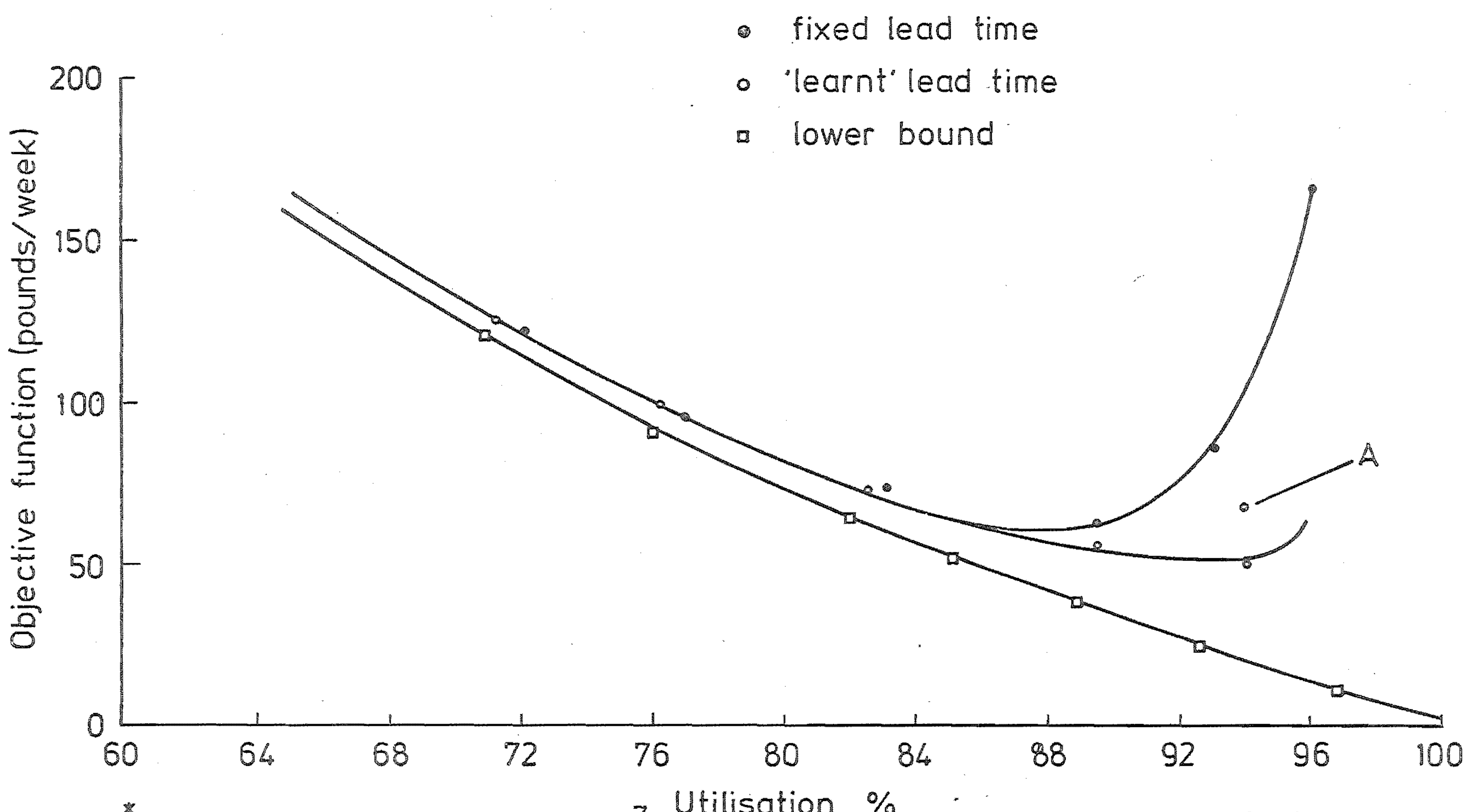
*Reproduced from Davies's Thesis[7].

*TABLE 2-3.

Minimum Float Rule - objective function versus utilisation, prediction of machine queueing time by exponential smoothing, smoothing constant 0.1, fixed stock buffer level of 2 hours usage per operation per part.

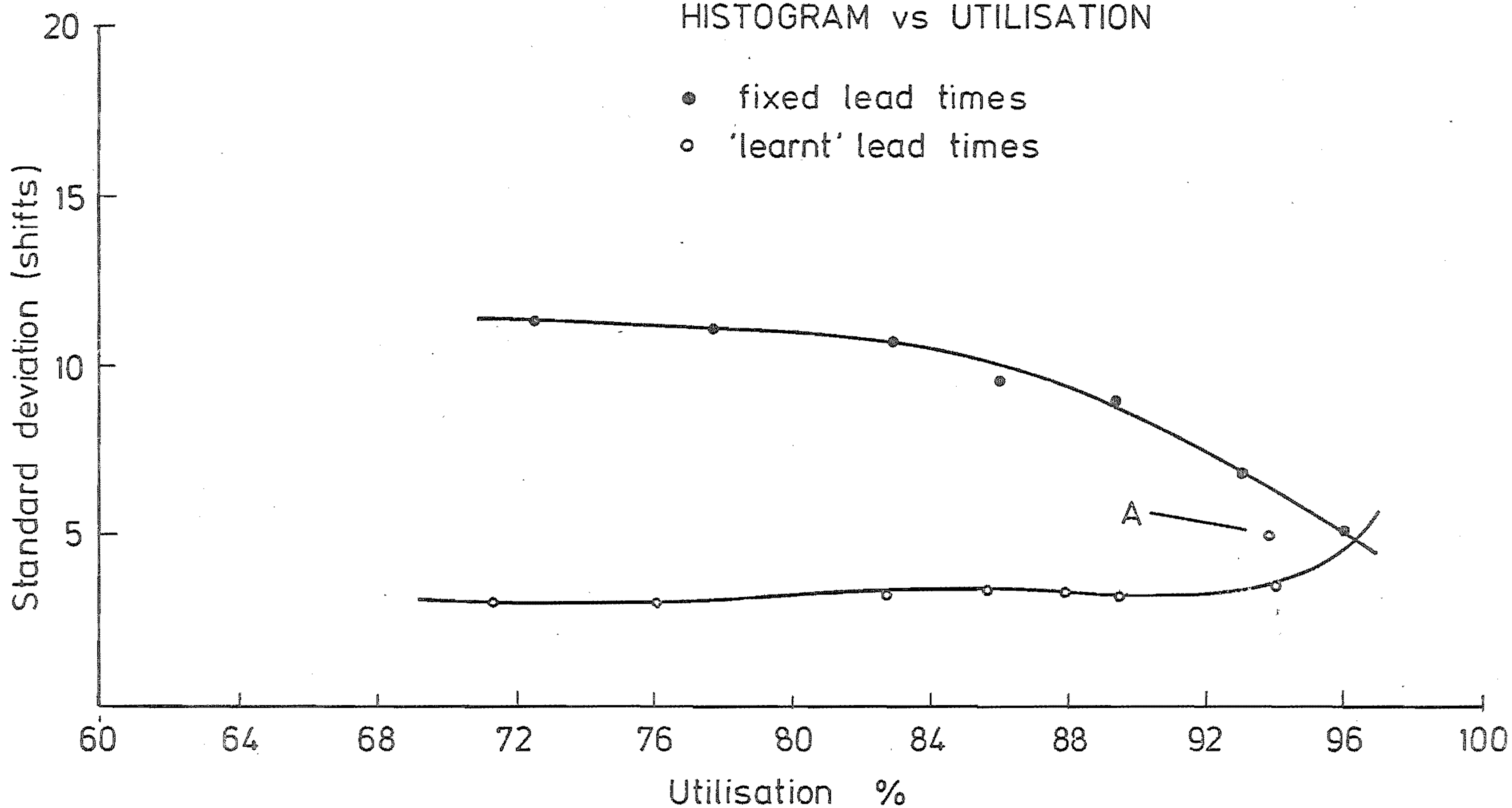
Shift length (hours)	3.0	2.8	2.6	2.5	2.4	2.3	2.2
Utilisation per cent	70.0	75.0	80.9	84.4	88.4	92.8	94.5
Unused Machine time cost	121.5	94.5	67.2	52.8	38.5	24.9	17.0
Work in Progress cost	3.8	4.1	4.4	4.5	4.7	5.3	5.5
Shortage cost	0.3	2.7	2.9	3.2	1.3	4.2	1.9
Total cost (£/wk)	125.6	101.3	74.5	60.5	44.5	34.4	24.4
Standard deviation of lead time error histogram(shifts)	3.004	3.154	3.103	3.038	3.069	3.985	3.849
Average queueing time (hours)	1.77	2.99	4.18	4.42	4.04	5.82	10.5

*Reproduced from Davies's Thesis[7].



*GRAPH 2·1
OBJECTIVE FUNCTION vs UTILISATION

*Reproduced from Davies's thesis[7]



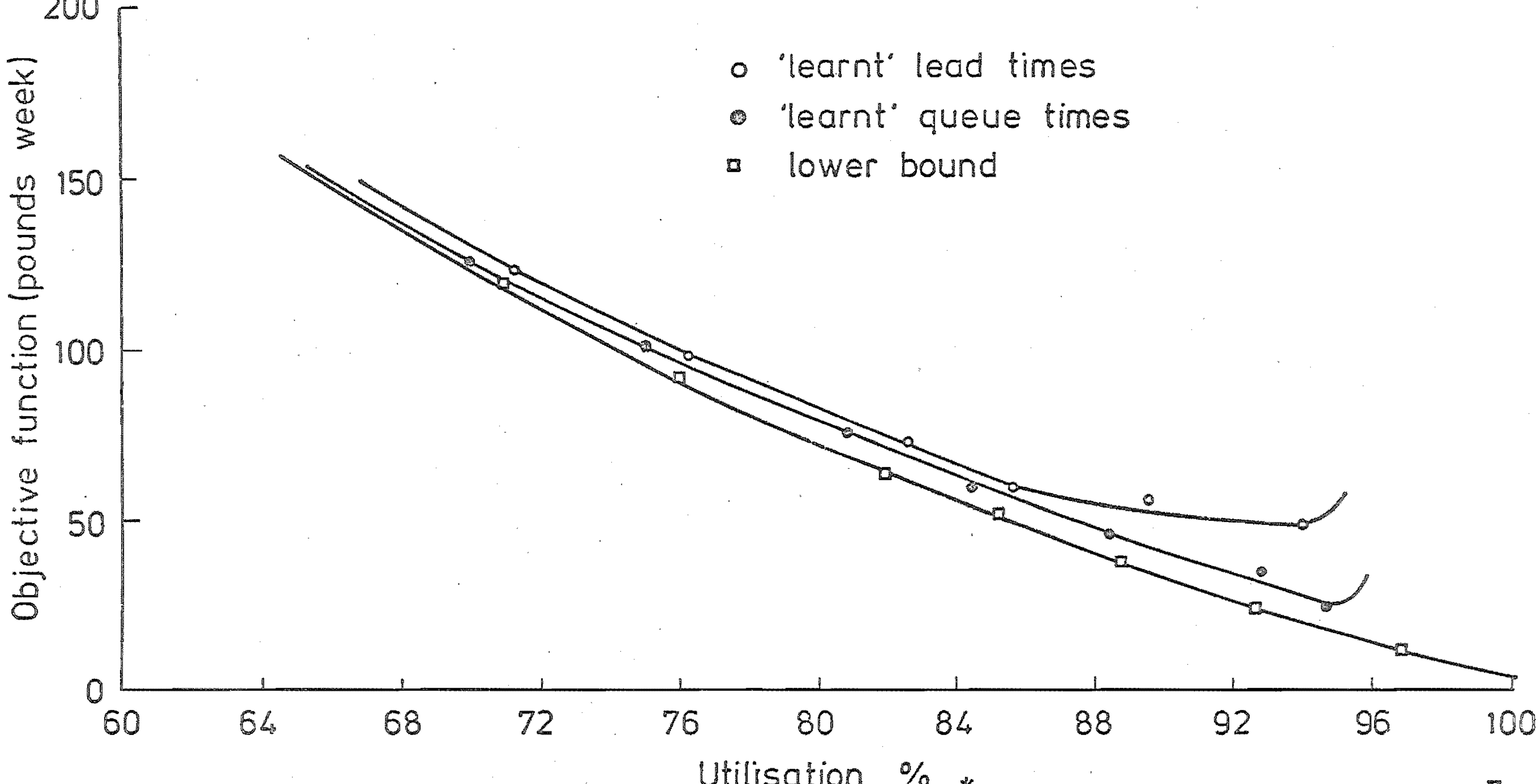
*GRAPH 2·2

STANDARD DEVIATION OF LEAD TIME ERROR HISTOGRAM vs UTILISATION

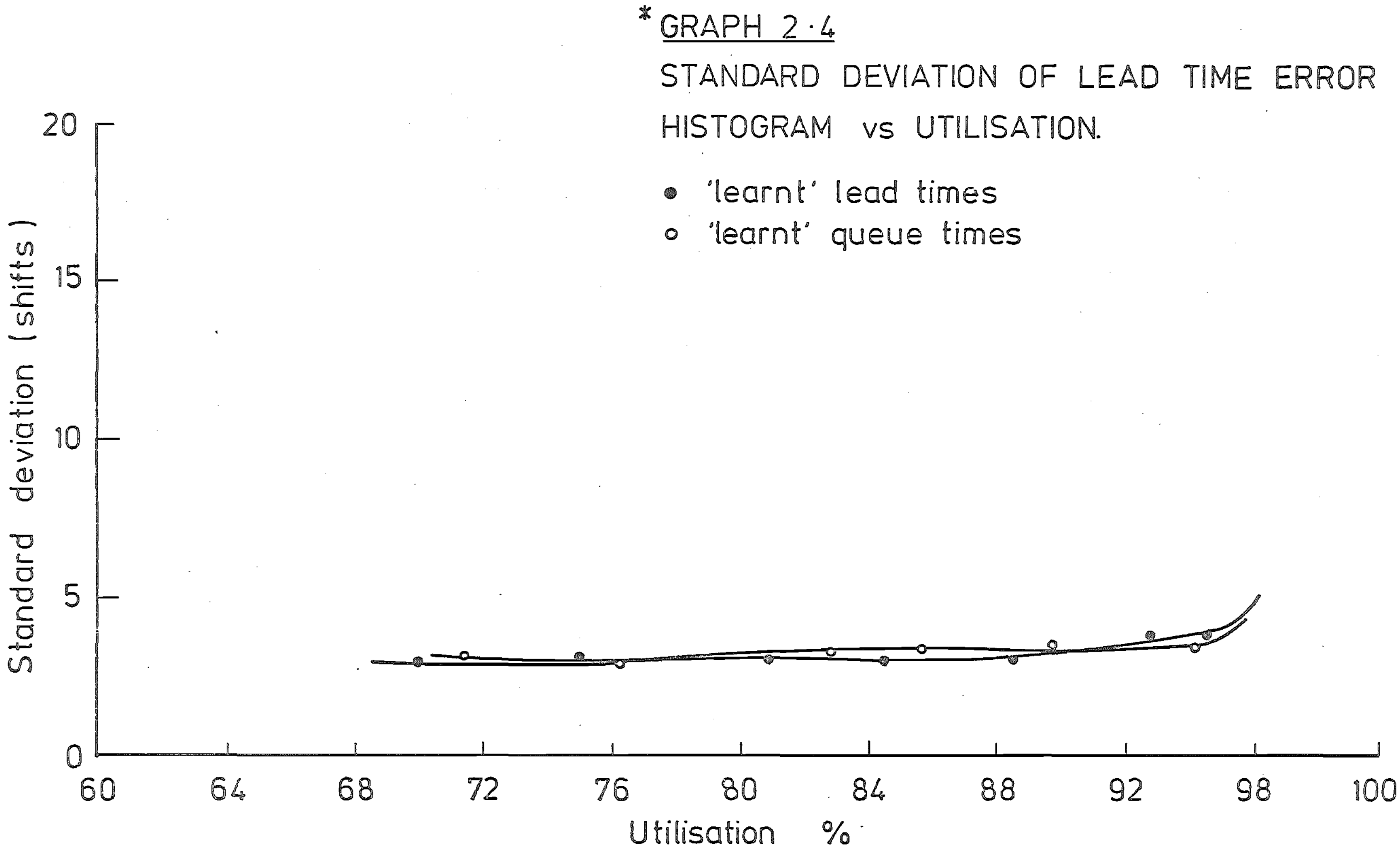
**Reproduced from Davies's thesis*[7]

*GRAPH 2 · 3

OBJECTIVE FUNCTION vs UTILISATION



*Reproduced from Davies 's thesis[7]



*Reproduced from Davies's thesis[7]

and the reduction increased with increasing shop utilisation. But below this shop utilisation, the two objective function curves coincided.

(b) The Adaptive Queueing Time Method Vs. The Adaptive Lead Time Method:

With reference to Graphs 2-3 and 2-4 and Tables 2-2 and 2-3:-

Over the range of shop utilisation tested, the two methods produced very similar standard deviations of the lead time error histograms.

However, the adaptive queueing time method reduced the objective function significantly, throughout the same range of shop utilisation. Of the three costs that constituted the objective function, the shortage cost was affected most. At 94% shop utilisation, it was reduced from 25.5 to 1.9 units.

Lead time error is defined as the predicted lead time minus the actual lead time. A positive lead time error signifies that a batch was finished earlier than expected. And a negative value corresponds to a batch which is late.

The lead time error histogram contains the distribution of the amount of time a batch is late or early.

The objective function, a performance criterion used by Davies in his research, is defined as the sum of unused machine time cost, shortage cost and work in progress cost.

Davies made two deductions and used them to analyse the results to reach conclusions:

1) "The standard deviation of this (lead time error) histogram reflects the degree of control that exists on the simulation model. A tight histogram represents a high degree of control and vice versa". Therefore a lower standard deviation corresponds to more accurate lead time prediction.

2) In view of 1), "when the standard deviation is low, the contribution of shortage cost and work in progress cost to the objective function is also low".

In other words, a method that gives lower standard deviation will also give lower shortage cost and work in progress cost.

He concluded implicitly that:

1) The adaptive lead time was predicting lead times more accurately than the constant lead time method, (since the standard deviations were reduced);

2) The adaptive queueing time method did not predict lead times more accurately than the adaptive lead time method, (since the standard deviations were very similar).

He did not account adequately the reductions in the objective function, particularly the shortage cost.

Perhaps, this is because the results contradict his second deduction:

1) With reference to Graphs 2-1 and 2-2, and Tables 2-1 and 2-2:-

Below 82% shop utilisation, substantially lower standard deviations only gave rise to insignificant reduction in the shortage cost and the work in progress cost.

2) With reference to Graphs 2-3 and 2-4, and Tables 2-1 and 2-2:-

The standard deviations produced by the two methods were not significantly different, over the range of shop utilisation tested.

Yet with the adaptive queueing time method, the reductions in the shortage cost was quite considerable.

A review of his thesis showed that he had not recorded or taken into account the mean of the lead time error histogram.

The mean may not be zero.

Depending on the constant used, the constant lead time method will give a positive mean at a low utilisation, and a negative value at a high utilisation. Therefore, in general, both the mean and the standard deviation have to be used, to assess the accuracy of the lead time predicted.

Since he considered the standard deviations only, his conclusions on the relative accuracies of the methods could not be held valid.

And in the absence of any mean figures in his thesis, the relative accuracies cannot be assessed.

In Figures 2-2, the shape of the standard deviation curve produced by the constant lead time method differs completely from the general shape according to classical queueing theory.

The constant lead time produces zero mean at a particular utilisation depending on the value of the constant used, and positive and negative means below and above that utilisation respectively.

Hence, the shape of the curve may be explained by assuming the means of the lead time error histograms to be zero, and calculate the standard deviations accordingly.

If this procedure was adopted, a distribution with majority of the batches completed late could be given the same standard deviation value as one with majority of the batches early.

But the shortage cost associated with the former is expected to be larger.

And since, in the thesis, the standard deviations due to a late majority distribution could not be distinguished from the ones due to the other, the standard deviation figures produced should not be used for deductions on shortage cost reductions.

This work will re-investigate the lead time prediction methods reported by Davies.

Davies used buffer stocks in his simulations.

It is not clear how buffer stocks affected his results.

To carry out a more fundamental research, buffer stocks will not be used in this research.

McCallion, Horsnell, Davies and Brittain (1970)[8] investigated a lead time prediction method which employed, in a modified form, the classical queueing theory.

Essentially, the method involved adjusting the queueing time estimate at a machine group for lead time prediction, using the expected utilisation of the facility.

In batch production, the demand upon the finished parts is often known for several months into the future. With this information, the expected utilisation of each machine group over that time could be computed, using the current expected queueing time to allocate approximately the resources over that time.

The queueing time estimate W was adjusted according to the expression:

$$W_1 = W_0 \times \frac{U_1(1-U_0)}{U_0(1-U_1)} \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (1)$$

where: U was the average utilisation of the machine group; the subscripts "0" and "1" referred to times t_0 and t_1 respectively; and t_0 being the reference time and t_1 being the future time at which the queueing time will apply.

For their investigation, they used an actual light machine shop with 39 groups of machines. Altogether, there were 57 machines: 7 operating on the day shift only, 13 on the night shift only, and 37 on both shifts. 226 individual parts in predetermined batch quantities were produced.

Like Davies, the queueing discipline used for the heuristic scheduling system was the minimum float rule.

And orders for parts were loaded into the job shop on the basis of minimising shortages.

To obtain a steady initial state of the machine shop, Brittain's procedure was again used.

In the simulation, the load on each machine group was predicted using the reference waiting times W_0 and the known future demand for finished parts. This "forward load" was averaged over an eight week period to obtain the average utilisation expected of each machine group.

Using Equation (1), the corresponding values of W_1, were calculated, for use in placing orders in the first four weeks of the above eight week period.

This process was repeated every four weeks.

The preliminary tests showed an undesirable behaviour.

When this was alleviated, - by a biased exponential smoothing of W_1, - this method was compared with one under the same load pattern, but using the reference waiting time W_0 for load time prediction. The comparison was made at an average shop utilisation of 76%.

It was found that for the two hundred weeks of shop operation simulated, the system based on smoothed variable waiting times consistently had lower overall costs than the one based on reference waiting times.

But the reductions in shortage cost was not mentioned. Neither were the means and standard deviations of the load time error histograms.

The undesirable behaviour was that: when the utilisation dropped, the allowed lead times shortened, and the work load pattern was disturbed to the extent that highly utilised machines became idle for prolong periods. This loss of production could not be made up, and large shortage costs resulted later.

To alleviate that, the W_1 values were exponentially smoothed, and biased to respond more rapidly to increases in the expected utilisation than to decreases.

A careful examination of the computer program used reveals the following:

(a) The subroutine that compiled the forward load information did not take the current work load (i.e. work in progress) into account, and therefore would under-estimate the utilisations of the machines.

A lower predicted utilisation gives a lower predicted queueing time according to Equation (1), and the allowed lead times are shorter than the actual.

This under-allocation of lead times amounted to the batches being late, and shortages, incurred.

(b) The subroutine that compiled the forward load information also failed to take the shortages in the forward period into account.

This reduced the number of batches loaded, and further under-estimated the utilisations of the machines.

The end result was the same as in (a).

These two factors acting together, made the shortages soar.

By biasing the exponential smoothing process to respond more rapidly to increases in demanded utilisation than decreases, the queueing time estimates become larger than otherwise.

And this was found sufficient to remedy the situation.

The version to be tested in this research will not use a biased exponential smoothing, but will instead have the situations (a) and (b) corrected.

It will be seen that this method, after the corrections, does not experience the undesirable behaviour mentioned.

CHAPTER 3

THE SIMULATION PROGRAM

3.1 INTRODUCTION

In this chapter, the machine shop model and the simulation program used are described.

3.2 THE MACHINE SHOP MODEL

The simulation program was written by Brittain (1969)[9], in Atlas Antocode.

In constructing the machine shop model for the simulation program, Brittain made the following assumptions:

(a) No machine may perform more than one operation at a time.

(b) Machines never breakdown and manpower of uniform ability is always available.

(c) For each operation, all materials, jigs and tools are available when required.

(d) Each operation takes a known finite time to perform.

(e) An operation, once started, must be performed to completion (no pre-emptive priorities).

(f) Each batch is an entity and may not be processed by more than one machine at a time.

(g) Each batch has only one route through the machine groups, alternative routes being prohibited.

(h) A batch, once started, must be finished (no order cancellations).

(i) Batches may be placed in a queue before any machine group.

(j) Perfect parts are consistently produced.

(k) Transport times between machines are negligible.

(l) Due dates for each batch, once calculated, remain fixed.

(m) There is no overtime working.

These assumptions are typical of those usually made by workers in the machine shop simulation field.

They are over simplification of real life, and some are questionable. However, they were introduced to make the model easier for programming.

For this investigation, the assumptions are considered to be appropriate, and no changes are made.

The machine shop model constructed by Brittain consists of machines formed into groups.

Machines in a machine group are functionally identical, but machines from different groups are not. This means that a particular operation that can be performed by a machine of a group, can also be performed by other machines of the group, but not by any machine from other group.

Batches entering the machine shop are put into a queue at the machine group of their first operation. When a machine finishes a job, the highest priority batch is selected from the queue at the group of which the machine is a member, and loaded to the machine.

The finished job joins the queue appropriate to its next operation.

Thus, a batch will be loaded in the correct sequence to every machine group associated with the operations required of it and will eventually complete its final operations and leaves the machine shop.

A batch, when completely manufactured, is added to the stores quantity of the corresponding part.

At weekly intervals, a quantity equals to the average weekly usage is subtracted from the stores quantity of the part.

Whenever the quantity of a part in work plus the quantity in stores drops to the quantity that will be used during the expected lead time, the "batch loader" introduces a new batch of the part into the machine shop.

No safety stocks are held, and failure of a batch to meet its due date will result in a shortage.

3.3 THE SIMULATION PROGRAM

Brittain's program was later translated into Fortran.

The Fortran version which appeared in Aswed's thesis[10], was modified and simplified extensively for this research. Its present size is about half of its original size.

It is run on both IBM 360/44 and Burroughs B6700. Besides the main program, there are 13 subroutines and three functions.

Total core storage required is about 8500 words: 1500 for program code and 7000 for arrays.

The flow chart of the main program is shown in Figure 3-1. The key and a copy of the simulation program are given in the Appendix 2.

3.3.1 The Main Program

At the beginning of a simulation run, the program reads in: the information about the parts to be produced by the machine shop, the size and the initial state of the machine shop, as well as the number of weeks to be simulated, the number of shifts in each week and the length of each shift.

Batches are loaded into the shop to avoid shortage, and progressed through the machine shop in the manner described. Completed batches are added to the stores. And at weekly intervals, parts are drawn to meet demand.

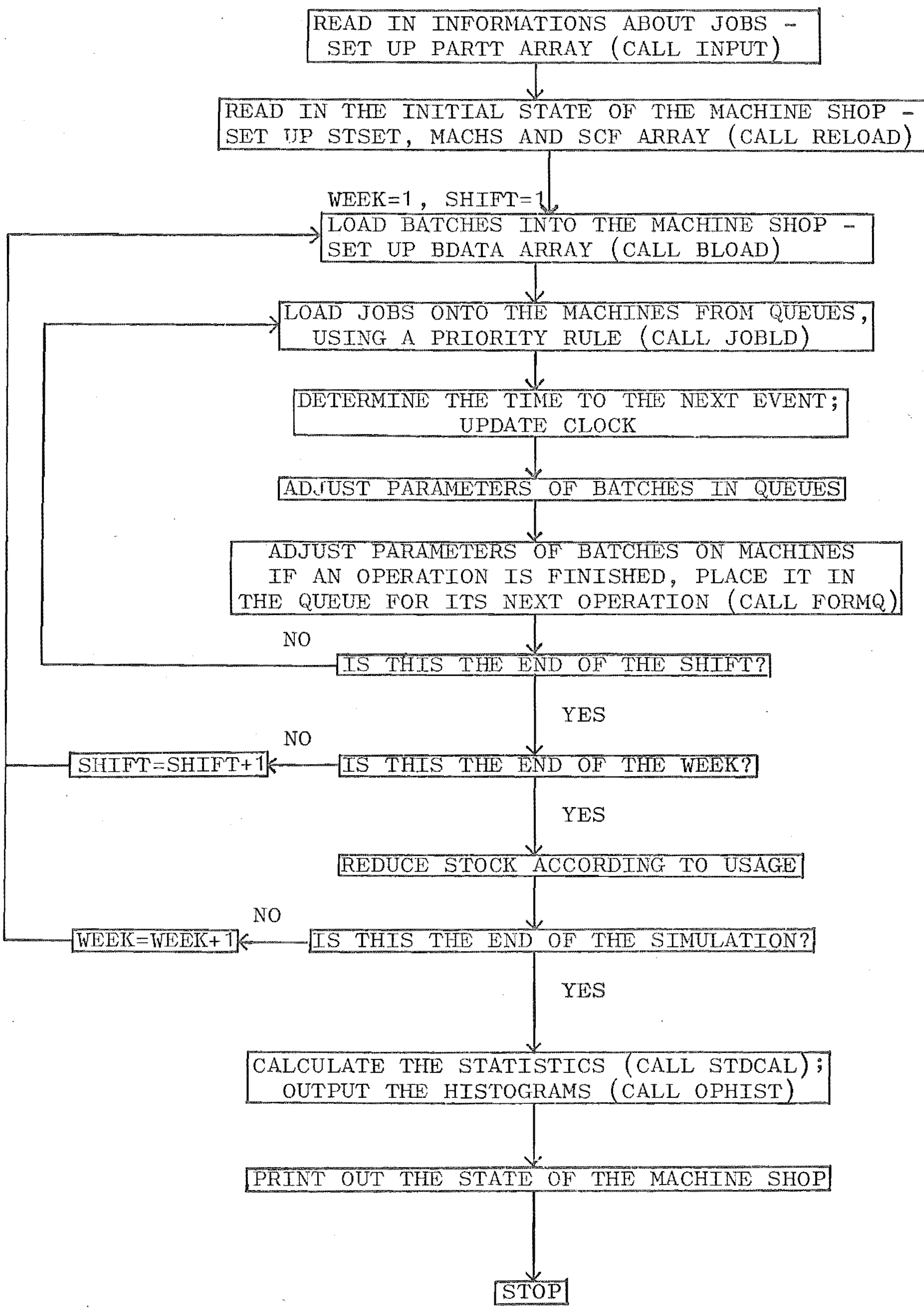
Before the termination of the simulation, the state of the machine shop is printed out.

The program also outputs the histograms of actual

FIGURE 3-1.

FLOWCHART FOR THE MACHINE SHOP SIMULATION

MAIN PROGRAM



waiting times, actual lead times, and the histogram of the lead time errors.

3.3.2 The Subroutines

(a) SUBROUTINE BLOAD

This is called at the beginning of each shift by the main program.

It loads a new batch of fixed quantity of a part if the quantity in stores plus the quantity in work in progress drops to the expected demand during the lead time period.

For a batch loaded, it calls FUNCTION BCHADD to allocate storage area in BDATA array for informations relating to the batch. After updating the stock control file, its calls SUBROUTINE FORMQ to place the batch in the queue for its first operation.

(b) SUBROUTINE FILTIM

Called by SUBROUTINE FWLOAD, this subroutine allocates the resources demanded by a batch.

(c) SUBROUTINE FORMQ

For a batch that has just completed its last operation, this subroutine places it into the stores.

For a batch that has just completed an operation other than the last, this subroutine places it in the queue of the appropriate machine group.

(d) SUBROUTINE FWLOAD

This subroutine predicts the distribution of demand on each of the machine groups, for a given length of time.

(e) SUBROUTINE INPUT

Called at the beginning of a simulation, it reads in the information of the parts to be produced.

(f) SUBROUTINE JOBLD

When a machine becomes available, this subroutine loads onto it a job selected from the queue.

(g) SUBROUTINE OPHIST

It prints out the histograms with their means and standard deviations.

(h) SUBROUTINE OUTPUT

This subroutine prints out the state of the machine shop.

For each machine, it prints out the total machining time performed, and the current job on the machine.

For each queue, it prints out the number of batches, and the informations associated with each batch.

(i) SUBROUTINE PRIOR

Called by SUBROUTINE JOBLD, it calculates the priority parameters, and selects the batch with the highest priority from a given queue, according to a priority rule specified.

(j) SUBROUTINE RELOAD

This subroutine reads in the initial state of the machine shop and of the stores, at the beginning of a simulation.

(k) SUBROUTINE STOCAL

It calculates the mean and standard deviation of a histogram.

(l) SUBROUTINE STORES

Called by SUBROUTINE FORMQ when a batch has its final operation completed, this subroutine updates the stock control file.

(m) SUBROUTINE STRLQE

This is used to store, in the LQE array, the actual waiting times, the actual lead times and the lead time errors.

3.3.3 The Functions

(a) INTEGER FUNCTION BCHADD

This is called by SUBROUTINE BLOAD to allocate areas, in the BDATA array, for informations relating to a batch that has just been loaded into the machine shop.

(b) INTEGER FUNCTION LOCATE

It locates the position of a part in the stock control file, given the position in the PARTT array.

(c) INTEGER FUNCTION OPTIME

For a given operation of a batch, it calculates the total time required.

CHAPTER 4

THE ENVIRONMENT OF THE SIMULATION MODEL

4.1 INTRODUCTION

This chapter describes the data used for the simulation program outlined in the last chapter.

It also describes a method of setting up the initial steady-state condition for the simulation.

4.2 DATA

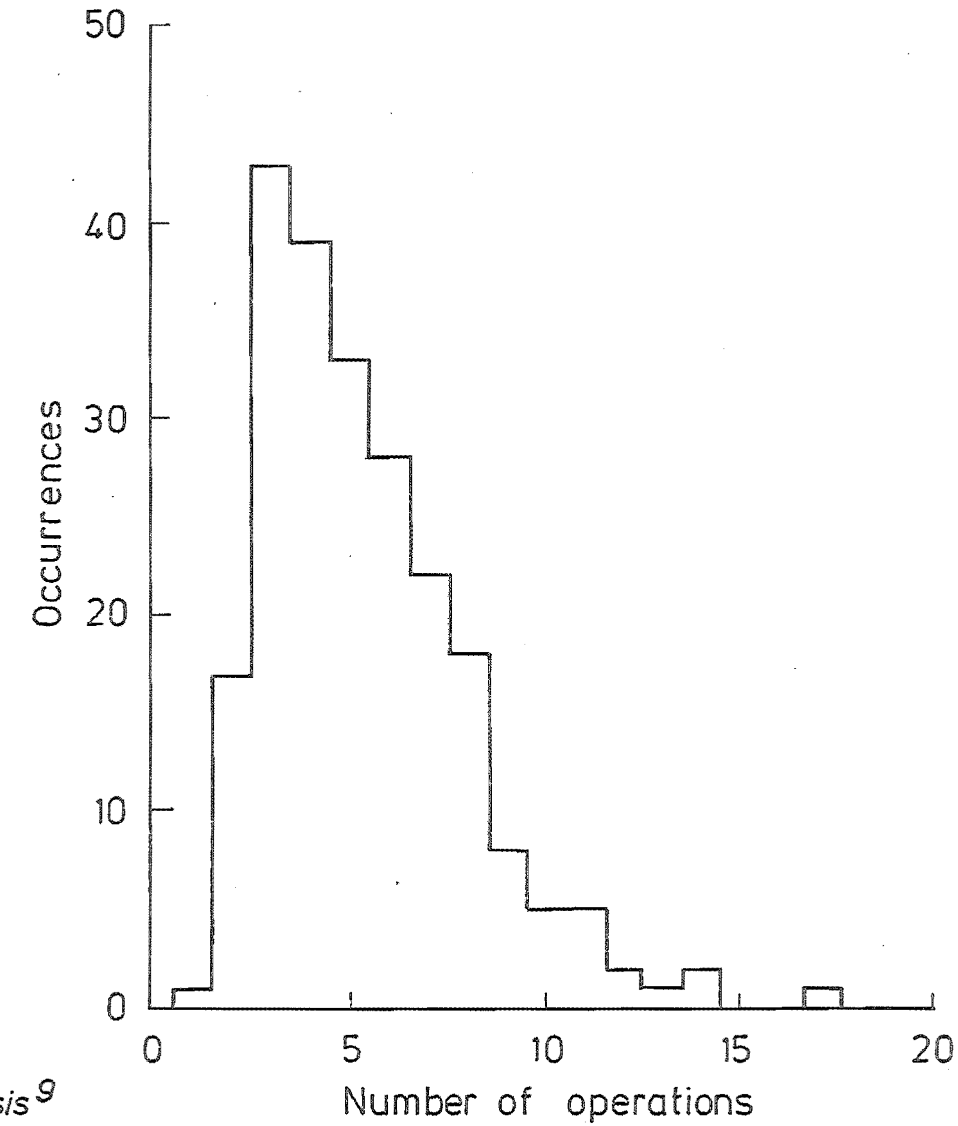
For this research, all the data used were generated from the distributions of actual shop operating data obtained by Brittain.

The actual shop operating data consisted of 226 engine parts manufactured in the light machine shop of Mirrlees-National Ltd., using non-specialised machines.

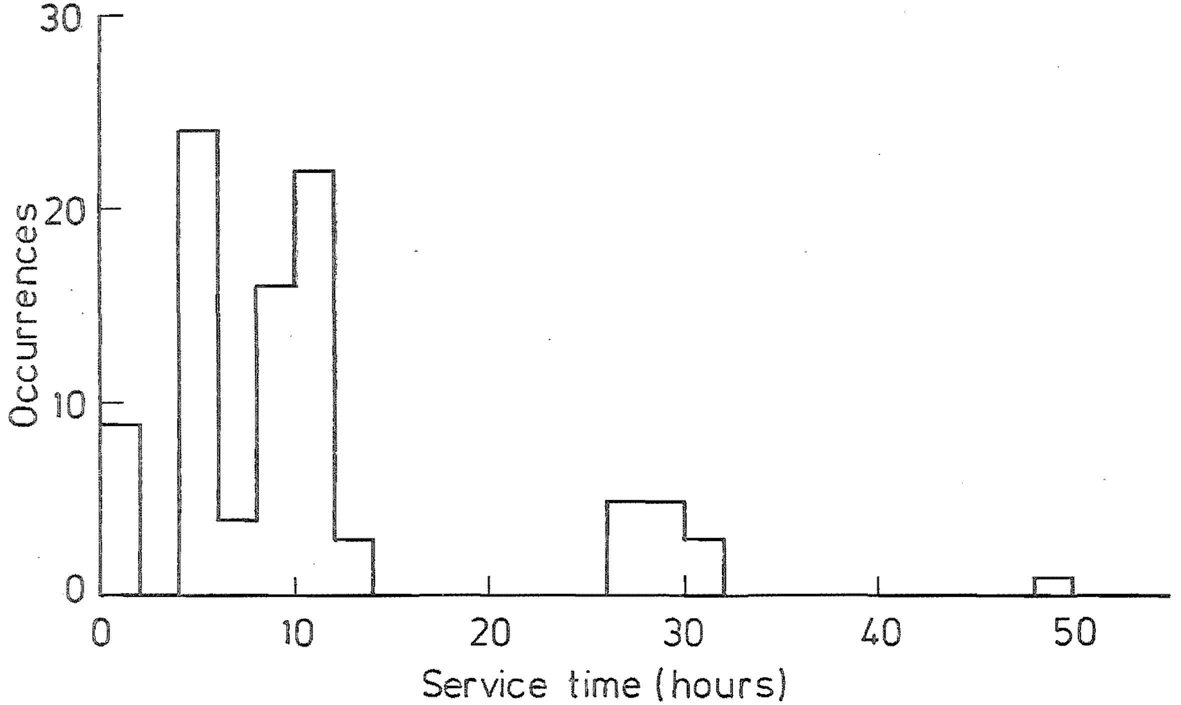
The distributions which appeared in Brittain's thesis, are reproduced here. Figure 4-1 is the distribution of the total number of operation of a part, and Figure 4-2 is the distribution of the service time of an operation.

No information was available on the quantity in the standard batch of each part. So this was generated randomly, assuming a uniform distribution with a minimum of 20 and a maximum of 100.

GRAPH 4·1 HISTOGRAM OF TOTAL NUMBER OF OPERATIONS FOR A PART



Reproduced from Brittain's thesis[9]



GRAPH 4·2 SERVICE TIME DISTRIBUTION

Reproduced from Brittain's thesis[9]

There was no information on the weekly demand of each part either. This was also generated randomly, but adjusted later with liberty to achieve a realistic and desired level of shop utilisation. Once this is achieved, the weekly demand figures remain fixed.

For this research, the size of the machine shop was limited to 10 machine groups, with one machine in each group.

Three sets of data were generated.

Data Set 1 was used for majority of the tests.

Data Sets 2 and 3 were used to examine the effect of job mix on the mean and the standard deviation of the lead time error histogram.

All the data sets gave rise to unevenly loaded machine shops.

Table 4-1 gives, for each data set, the average utilisations of the machine groups, at a shift length of 8.00 hours.

Appendix 1 describes the manner, in which the total number of operations for a part and the operation time of each operation are generated from Brittain's distributions.

TABLE 4-1 THE AVERAGE UTILISATION OF THE MACHINE GROUPS AT A SHIFT LENGTH OF 8.00 HOURS.

MACHINE	AVERAGE UTILISATION		
GROUP NO.	DATA SET 1	DATA SET 2	DATA SET 3
1	60%	65%	45%
2	26%	38%	46%
3	30%	34%	39%
4	18%	23%	32%
5	37%	52%	45%
6	56%	61%	56%
7	48%	50%	30%
8	51%	31%	47%
9	32%	52%	28%
10	49%	47%	35%

4.3 THE INITIAL STATE OF THE MACHINE SHOP

The disturbance due to starting a simulation with an empty machine shop takes a considerable time to settle down.

To overcome this long and unproductive simulation before the steady condition could be arrived at, Brittain developed a forward load predictor procedure, described as below.

In this procedure, the lead time of a part is assumed to be 2.5 times the total processing time. The initial store quantity is assigned as the usage during the estimated lead time plus a randomly generated amount of up to 10 weeks' usage. Initially, there is no work in progress, i.e. the machine shop is empty.

From the weekly usage of a part and its estimated lead time, the dates, on which new batches have to be loaded to avoid shortage, can be determined.

Assuming each batch has to wait at a machine group for a length of time equal to the float per operation, and unlimited capacity on each machine group, the starting times of operations can be determined.

With these assumptions all orders are kept exactly to schedule, and new orders are placed in time to meet the future demand.

This forward load predictor is run for 100 week period. The distribution of batches at the end of the period are modified, if necessary, to account for the finite capacity of each machine group. The modification is such that: any batch loaded to a machine group which is above its capacity, is returned to the queue.

In this thesis, the state of machine shop obtained will be referred to as the First Generation Initial Steady-State condition.

A simulation run using this initial steady-state condition is performed over a 50 week period, to give a proper estimate of the waiting times.

In this thesis, the simulation run will be referred to as the First Generation Simulation Run.

The forward load predictor is modified slightly, and re-run for 100 weeks. Instead of assuming the waiting time for each operation as the average float per operation, the waiting time estimates obtained from the first generation simulation run are used.

The usual adjustment to account for the finite machine capacity is also carried out.

In this thesis, the state of machine shop thus obtained is referred to as the Second Generation Initial Steady-State Condition.

It is used as the initial state for the simulation proper, referred to as the Second Generation Simulation Run in this thesis.

It will be seen that steady-state conditions are achieved in the Second Generation Simulation Runs.

CHAPTER 5

EVALUATION OF THE FOUR LEAD TIME PREDICTION METHODS

PART I

5.1 INTRODUCTION

The four lead time prediction methods described in Chapter 2 are:

1) The constant lead time method
2) The adaptive lead time method
3) The adaptive queueing time method, and
4) The queueing theory with forward load method.

Davies showed that Method 2) gave a lower objective function than Method 1), and Method 3) gave a lower objective function than Method 2).

But he did not account for the reductions adequately. Also, he did not produce or mention the means of the lead time error histograms. And the standard deviations he produced are questionable.

It is proposed to re-simulate and re-investigate the three methods in this and the next chapter.

McCallion *et al.* reported that Method 4) gave a lower objective function than a method using fixed waiting times.

But no information on the means and the standard deviations was given, and the original version of Method 4) gave a soaring shortage cost.

It is also proposed to re-simulate and re-investigate Method 4) in this and the next chapter, and to compare its performance with that of Methods 1), 2) and 3).

It is decided to test the original version of Method 4) with the errors, described in Chapter 2, rectified, instead of the version used by McCallion *et al*.

Exponential smoothing will be applied to the queueing times obtained by using Equation 2-1, but only to reduce the variability.

The method with reference queueing times, used by McCallion *et al*. merely as a basis of comparison, will not be investigated in this research.

In the investigations that follow, the priority rule to be used will be the minimum float rule, the same as that used by Davies and McCallion *et al*.

This rule states that: load next the batch with the minimum float. And float is defined as the time to due date less the time of operations not yet performed.

In this chapter, the system response rate and the stability are examined.

In the next chapter, the accuracy of the lead times predicted by each method is examined.

5.2 SYSTEM RESPONSE RATE

The system response rate concerns the rapidity with which changes in the state of a machine shop are monitored for lead time prediction.

In the constant lead time method, the lead time estimates are not updated at all. The system response rate may be considered to be infinitely slow.

In the adaptive lead time method, the influence on the lead time of a part, due to the changes in the machine shop, can only be monitored when a batch of the part has been finished.

Thus, a considerable length of time may have elapsed before the changes are monitored and the system response is considered sluggish.

In the adaptive queueing time method, any change in the state of a machine shop is monitored when a job is loaded onto a machine.

The system response rate is therefore increased.

However, like all forecasting systems using exponential smoothing, the last two methods give predicted values that lag the actual.

In the queueing time with forward load method, changes in the state of a machine shop are monitored in advance.

The lead time estimates are expected to keep pace with, rather than to lag, the actual lead times. Thus, the system response rate can be considered to be further increased.

But, any local deviation of the load pattern from the predicted cannot be fed back for adjusting the queueing time estimates, until the next forward load prediction is performed.

This inability to adapt immediately to any deviation may be regarded as a reduction on the system response rate. Hence, the overall system response rate may not be faster than that of the last two methods.

Also, the accuracy of the lead times predicted in the periods where deviation occurs, will be reduced.

5.3 TESTS

To show the relative response rates of methods 2), 3) and 4), a series of first generation simulation runs was carried out.

The lead time estimates in the starting condition of the above simulation runs are generally not the steady-state values. But provided the methods are stable, the lead time estimates will reach their steady-state values.

The tests were carried out at a shift length of 6.00 hours. Data Set 1 was used.

For each method, two values of the exponential smoothing constant, α , were used.

5.4 RESULTS AND DISCUSSION OF RESULTS

Graphs 5-1 to 5-4 show the actual and the predicted lead times of two typical parts, using the adaptive lead time method, for two values of the exponential smoothing constant .

Graphs 5-5 to 5-8 show the same, using the adaptive queueing time method.

And Graphs 5-9 to 5-12 show the same, using the queueing theory with forward load method.

In each graph, the lead time estimate goes through a transition period and arrives at a steady-state level.

Examination of the graphs at the steady-state regions shows that in general, $\alpha = 0.10$ gives smoother operations than $\alpha = 0.30$. This value, $\alpha = 0.10$, will therefore be used for further research and analysis.

Note that the value $\alpha = 0.10$ for both the adaptive lead time method and the adaptive queueing time method is in agreement with Davies.

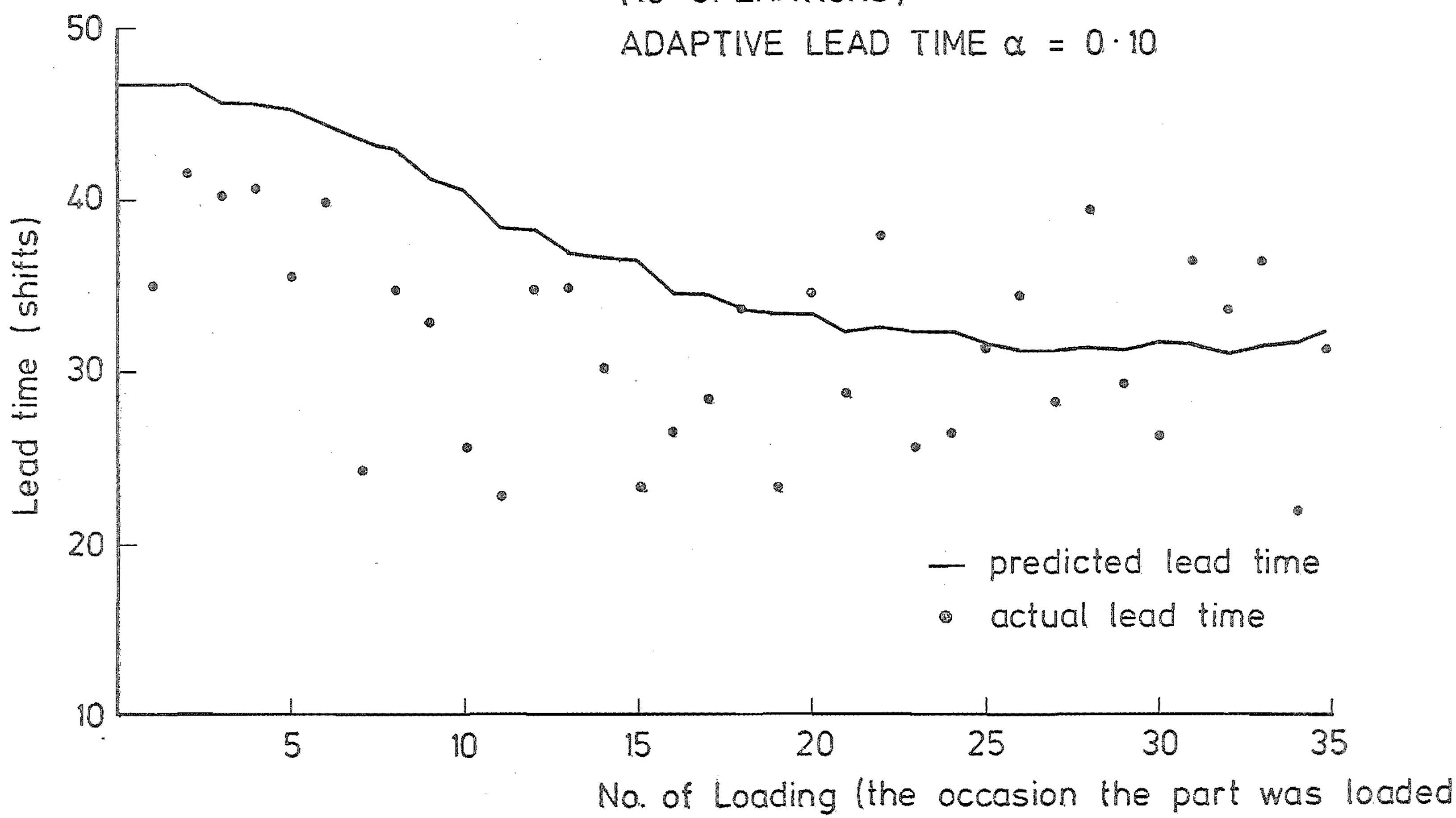
With reference to Graphs 5-1 and 5-2, for the adaptive lead time method:-

The lead time of part A is estimated correctly at the 18th attempt, and that of part B at the 16th attempt.

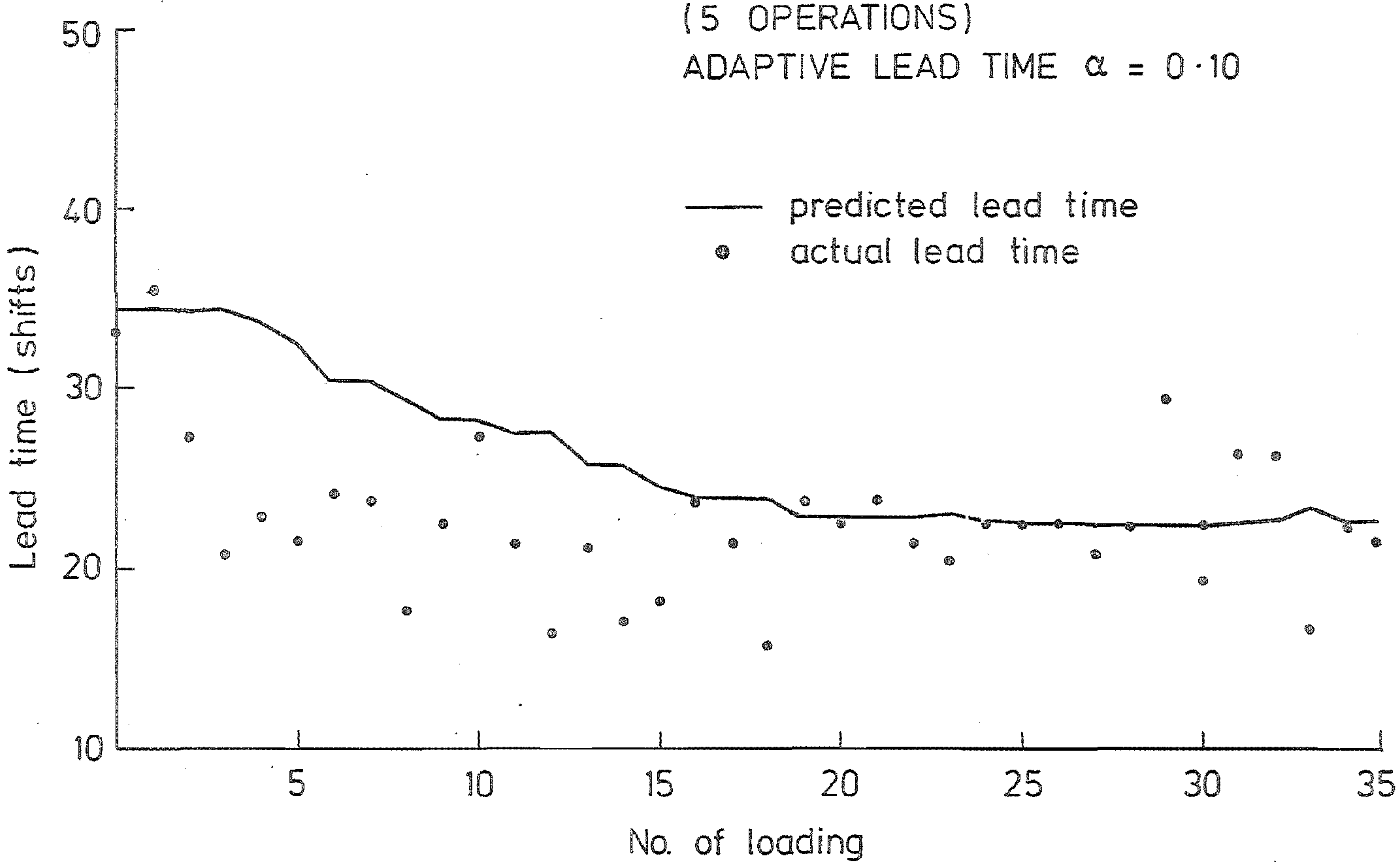
With reference to Graphs 5-5 and 5-6, for the adaptive queueing time method:-

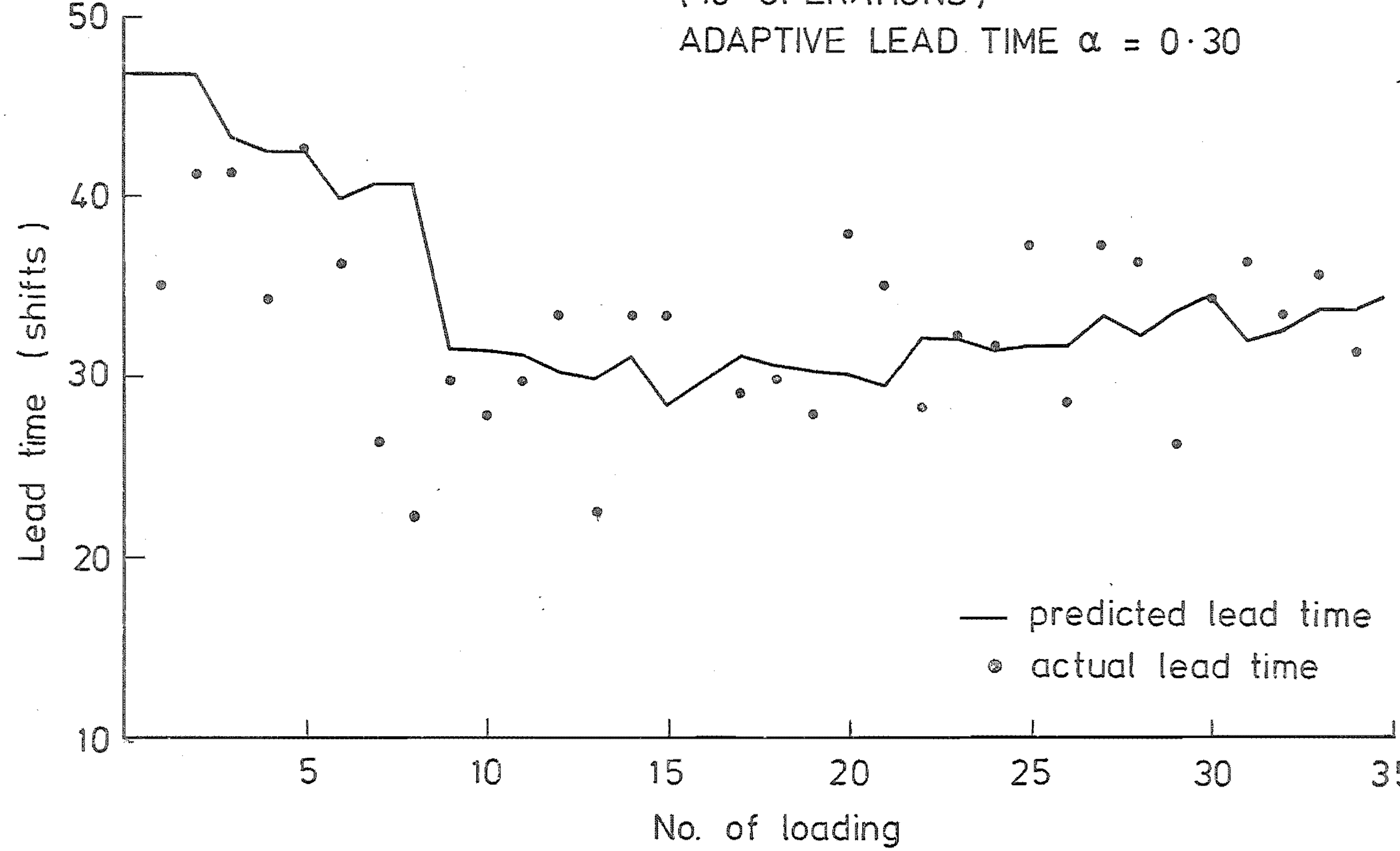
The lead time of part A is estimated correctly at the 8th attempt, and that of part B at the 9th attempt.

GRAPH 5·1 PART A
(10 OPERATIONS)
ADAPTIVE LEAD TIME α = 0·10

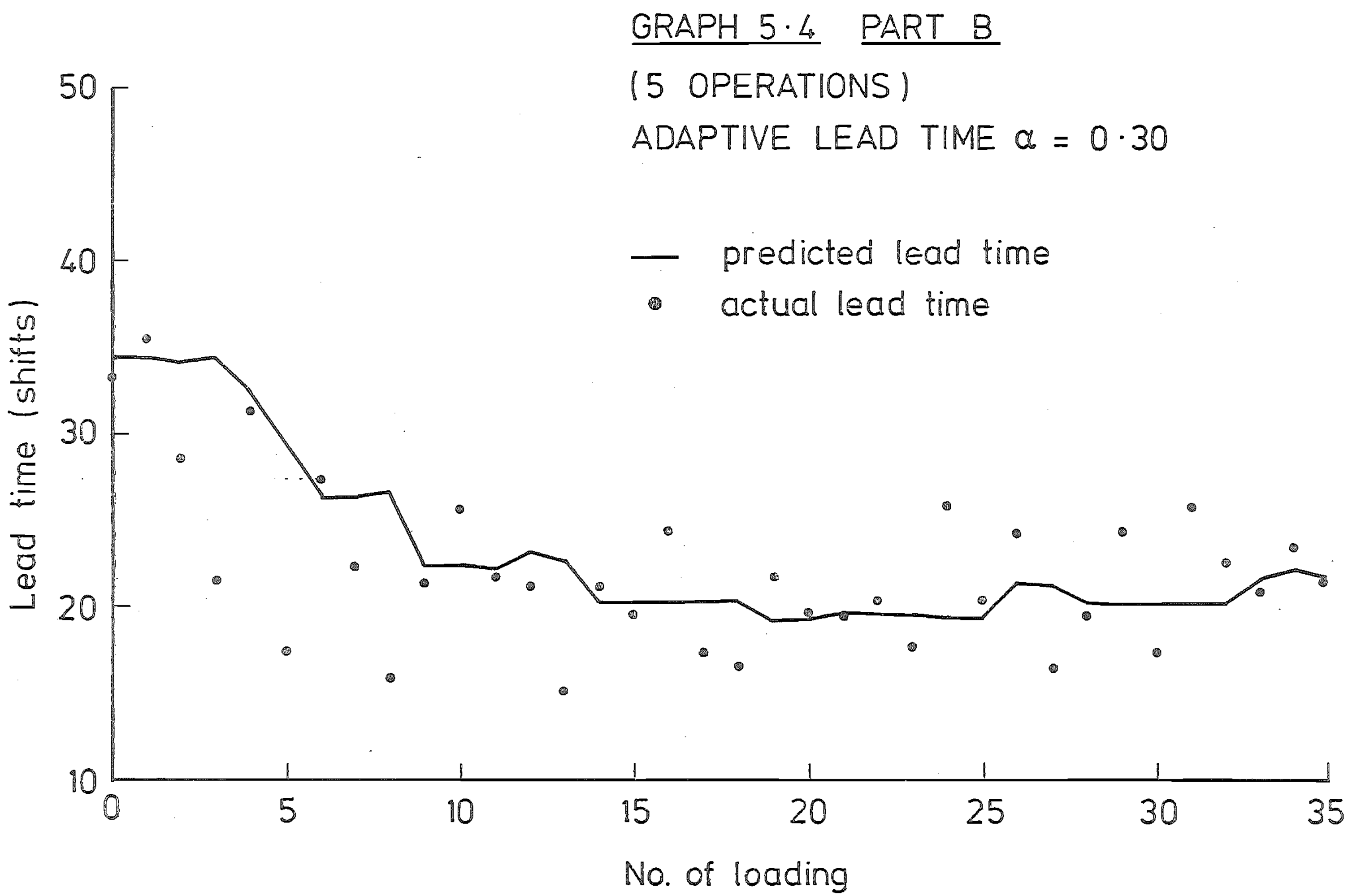


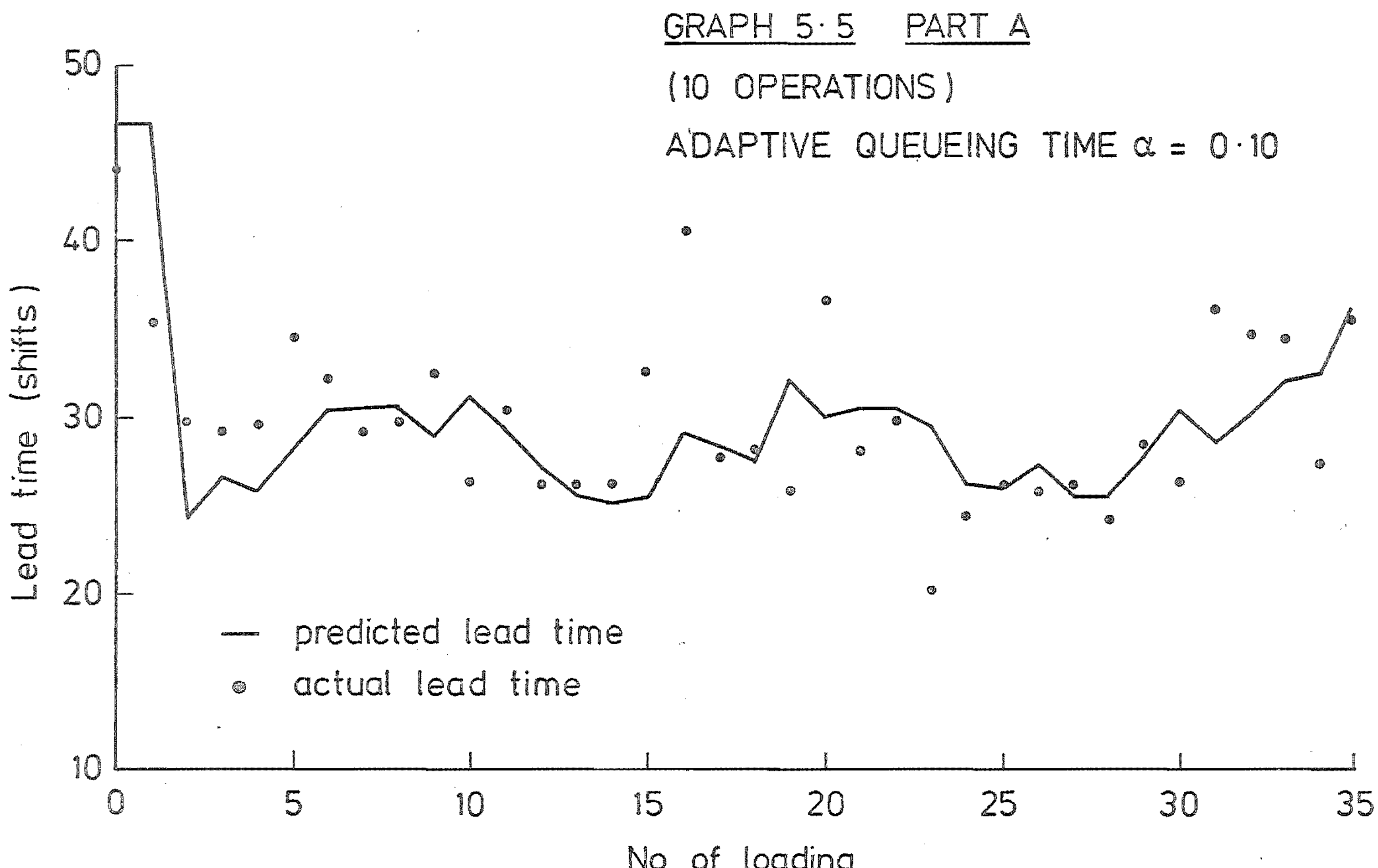
GRAPH 5·2 PART B
(5 OPERATIONS)
ADAPTIVE LEAD TIME $\alpha = 0 \cdot 10$

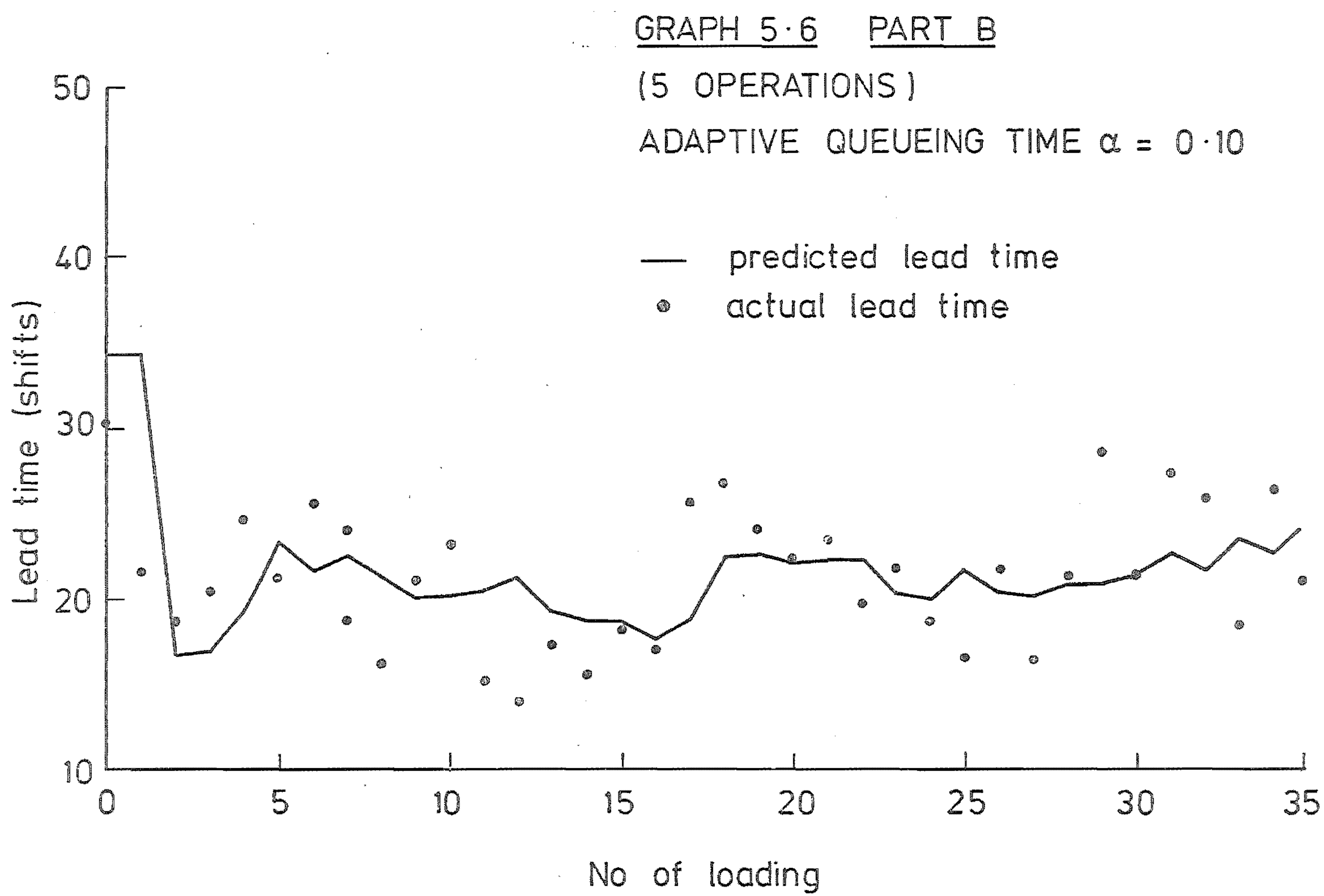




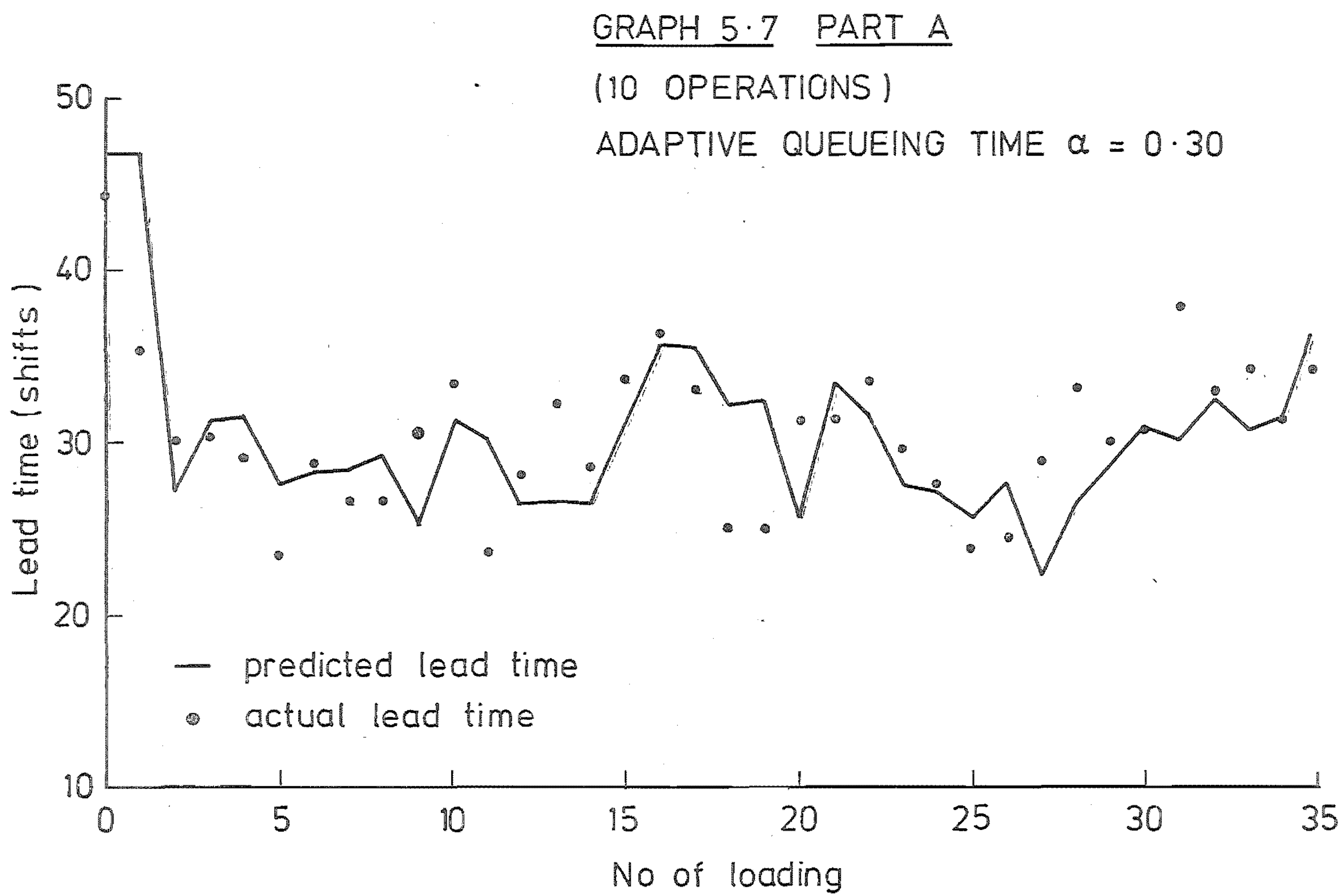
GRAPH 5·3 PART A
(10 OPERATIONS)
ADAPTIVE LEAD TIME $\alpha = 0 \cdot 30$


GRAPH 5·4 PART B
(5 OPERATIONS)
ADAPTIVE LEAD TIME α = 0·30
predicted lead time
actual lead time
Lead time (shifts)
No. of loading
50
40
30
20
10
0
5
10
15
20
25
30
35


GRAPH 5·5 PART A
(10 OPERATIONS)
ADAPTIVE QUEUEING TIME α = 0·10
Lead time (shifts)
50
40
30
20
10
0
5
10
15
20
25
30
35
No of loading
predicted lead time
actual lead time



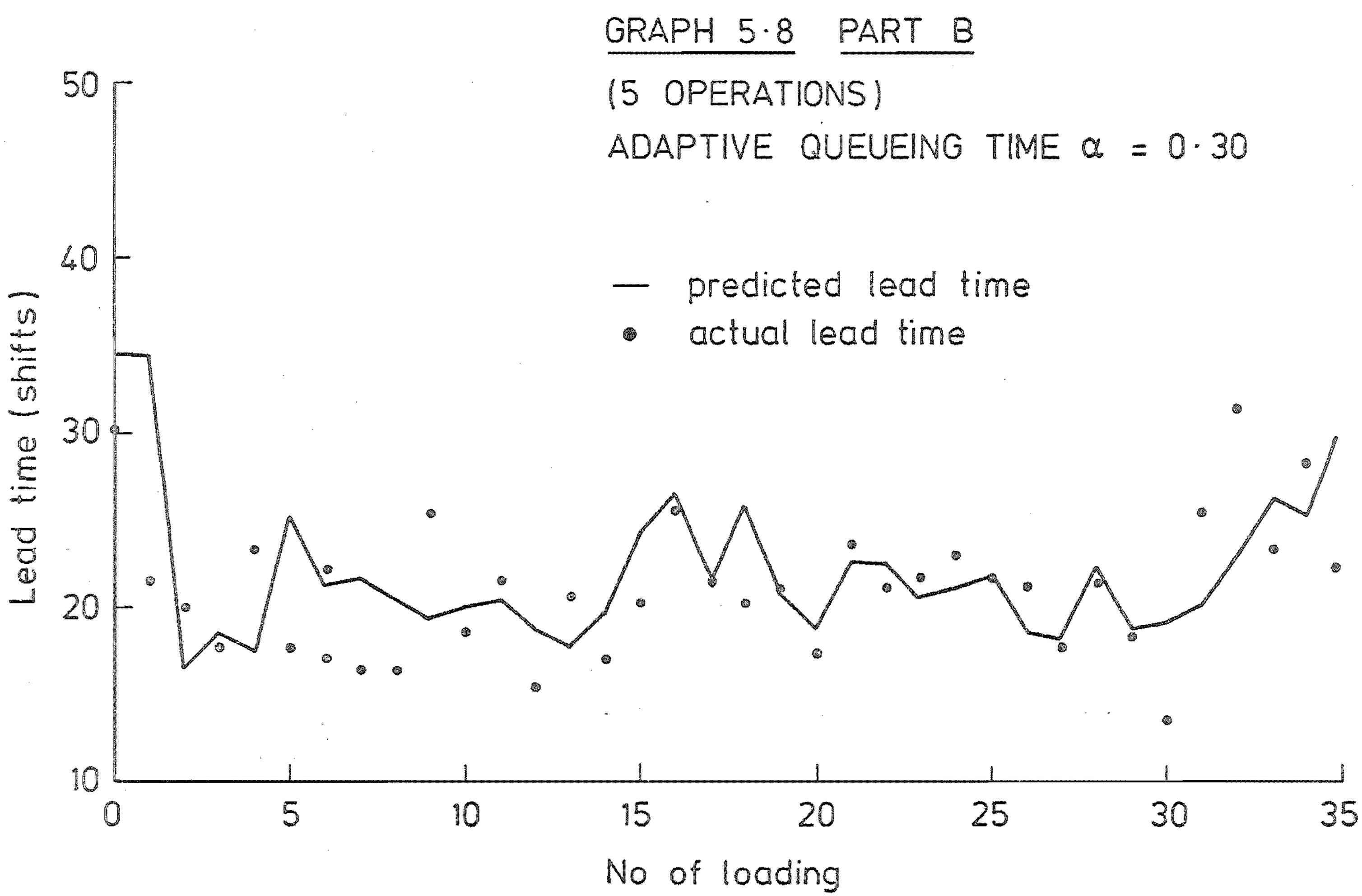
GRAPH 5·6 PART B
(5 OPERATIONS)
ADAPTIVE QUEUEING TIME α = 0·10
predicted lead time
actual lead time
Lead time (shifts)
50
40
30
20
10
0
5
10
15
20
25
30
35
No of loading

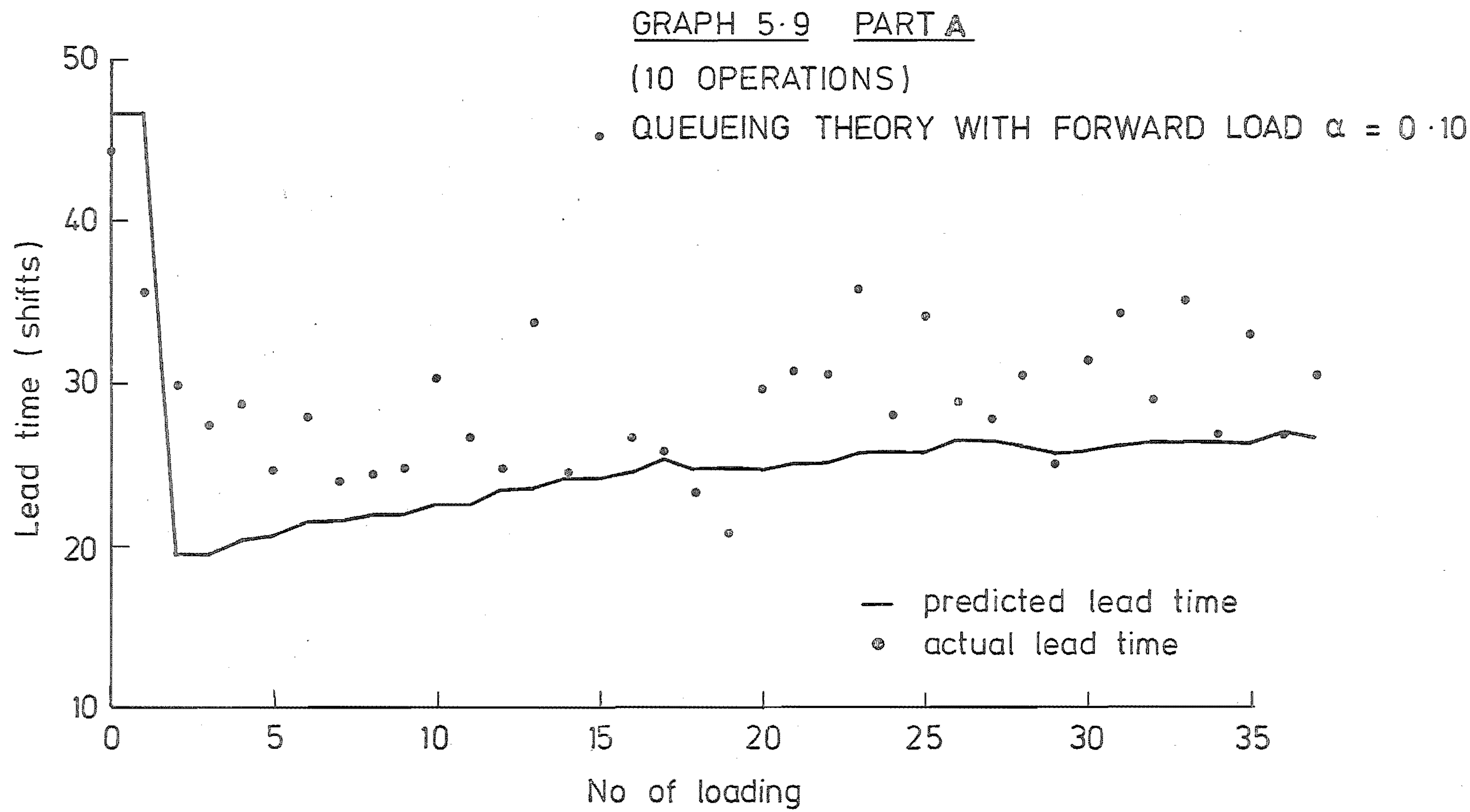


GRAPH 5·7 PART A

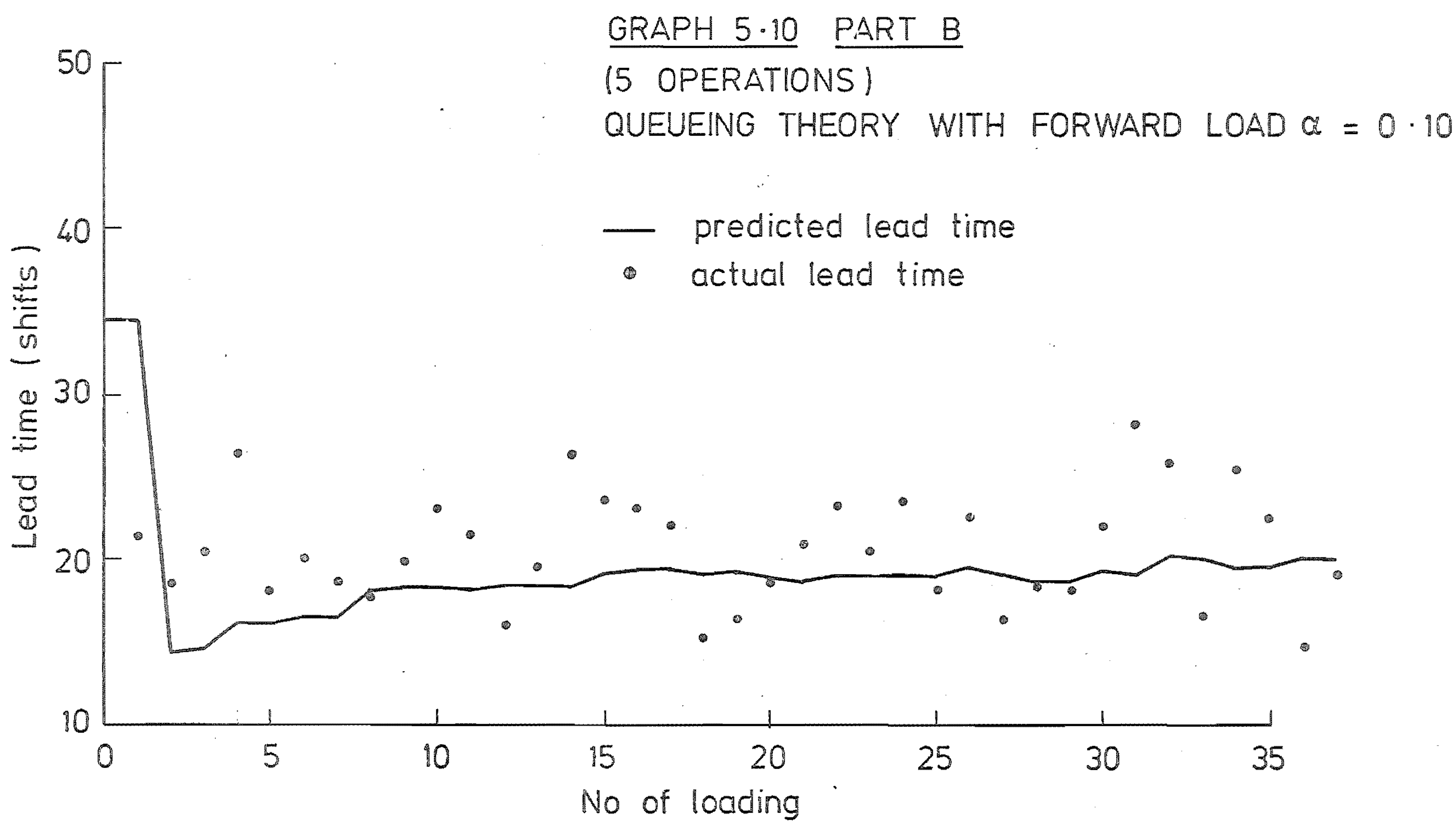
(10 OPERATIONS)

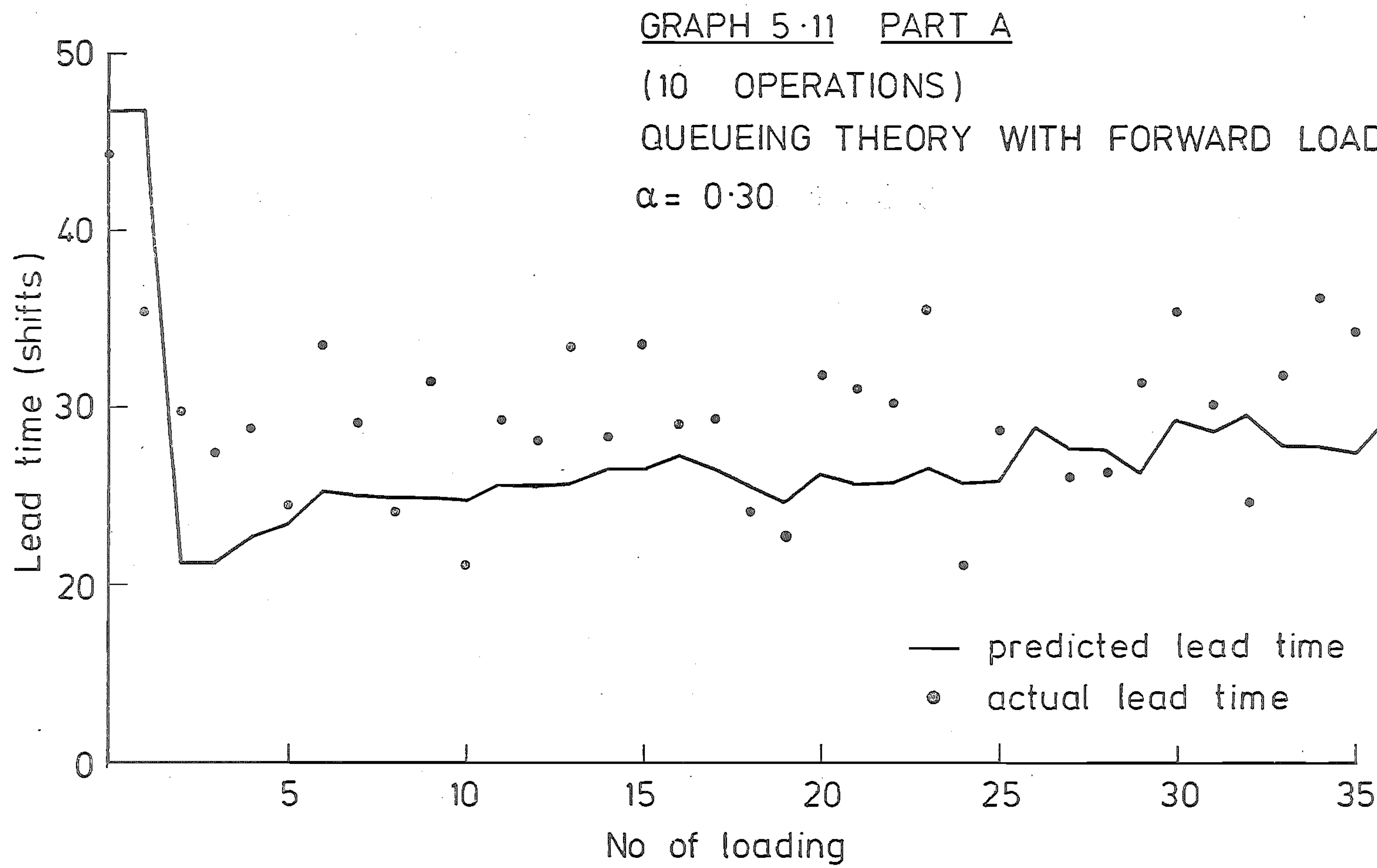
ADAPTIVE QUEUEING TIME $\alpha = 0\cdot30$

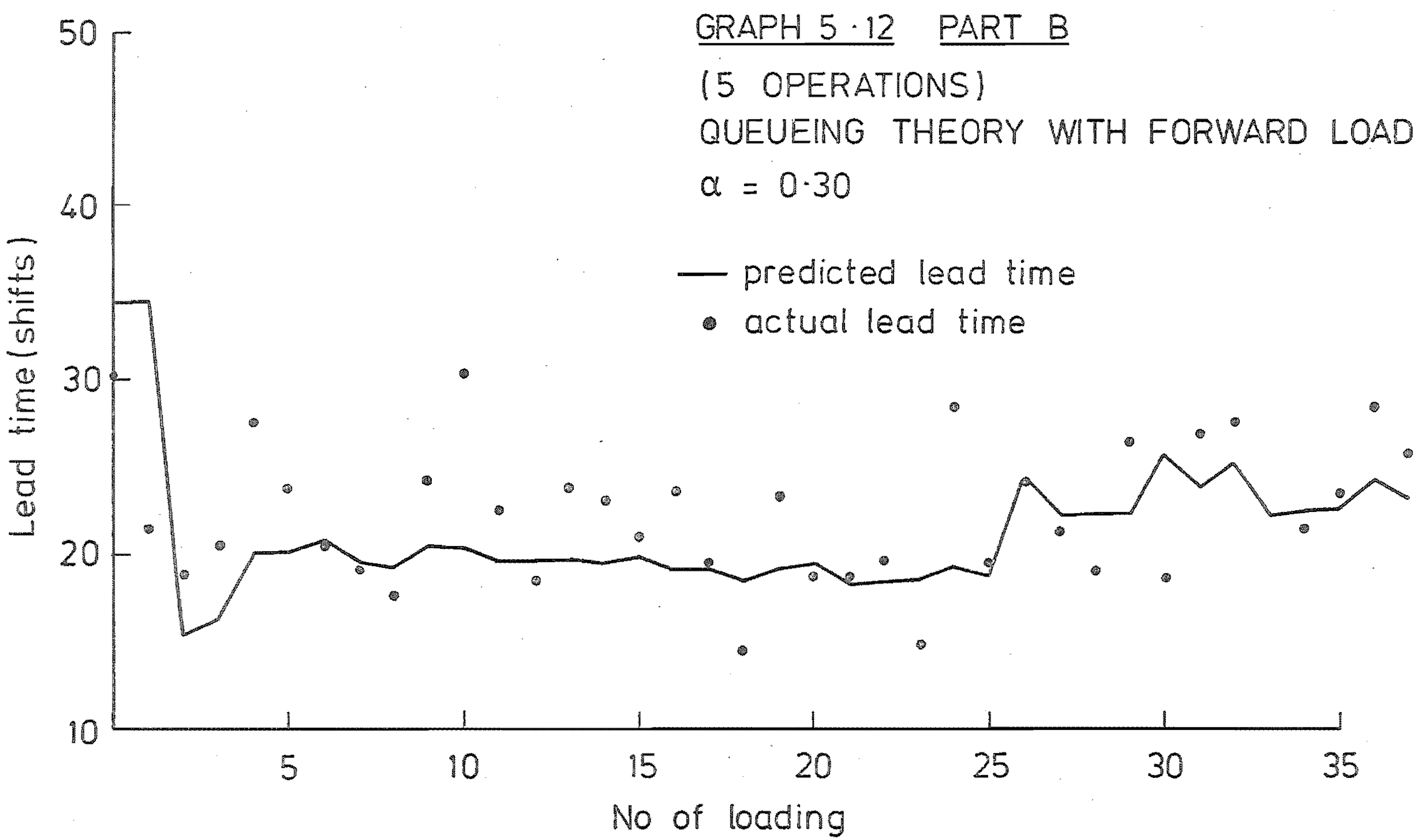

GRAPH 5·8 PART B
(5 OPERATIONS)
ADAPTIVE QUEUEING TIME α = 0·30
— predicted lead time
● actual lead time
Lead time (shifts)
50
40
30
20
10
0
5
10
15
20
25
30
35
No of loading



GRAPH 5·9 PART A
(10 OPERATIONS)
QUEUEING THEORY WITH FORWARD LOAD $\alpha = 0 \cdot 10$


GRAPH 5·10 PART B
(5 OPERATIONS)
QUEUEING THEORY WITH FORWARD LOAD α = 0·10
predicted lead time
actual lead time
Lead time (shifts)
50
40
30
20
10
0
5
10
15
20
25
30
35
No of loading


GRAPH 5·11 PART A
(10 OPERATIONS)
QUEUEING THEORY WITH FORWARD LOAD
α = 0·30
50
40
30
20
0
Lead time (shifts)
5
10
15
20
25
30
35
No of loading
predicted lead time
actual lead time


GRAPH 5·12 PART B
(5 OPERATIONS)
QUEUEING THEORY WITH FORWARD LOAD
α = 0·30
predicted lead time
actual lead time
Lead time (shifts)
50
40
30
20
10
5
10
15
20
25
30
35
No of loading

With reference to Graphs 5-9 and 5-10, for the queueing theory with forward load method: -

The lead time of part A is predicted correctly at the 14th attempt, but that of part B at the 8th attempt.

The above results confirm that the adaptive queueing time method has a faster system response than the adaptive lead time method.

The results also show that the system response rate of the queueing theory with forward load method is faster than that of the adaptive lead time method, but it may not be faster than that of the adaptive queueing time method.

5.5 SYSTEM STABILITY

For an iterative system of this kind, the question of stability must be examined.

The lead time of a part may be regarded as a random variable, because machining times vary, setting-up times vary and queueing times before machines vary. And according to the Central Limit Theory, the distribution of the lead time can be assumed to be normal.

Assume that the machining times, etc., are normally distributed, and that the machines have identical standard deviations. Then the standard deviation of the lead time distribution is proportional to the square root of the total number of operations of the part, as shown below:

$$LT = \sum_{i=1}^{N} QT_i + \sum_{i=1}^{N} MCT_i + \sum_{i=1}^{N} SUT_i \qquad \ldots\ldots\ldots\ldots(2)$$

$$\sigma^2_{LT} = \sum_{i=1}^{N} \sigma^2_{QT_i} + \sum_{i=1}^{N} \sigma^2_{MCT_i} + \sum_{i=1}^{N} \sigma^2_{SUT_i} \qquad \ldots\ldots\ldots(3)$$

$$= N\sigma^2_{QT} + N\sigma^2_{MCT} + N\sigma^2_{SUT}$$

(Assume: $\sigma_{QT_i} = \sigma_{QT}$, $\sigma_{MCT_i} = \sigma_{MCT}$, $\sigma_{SUT_i} = \sigma_{SUT}$, for $i \leq N$)

$$\therefore \sigma_{LT} = \sqrt{N}\sqrt{\sigma_{QT}^2 + \sigma_{MCT}^2 + \sigma_{SUT}^2} \qquad \ldots\ldots\ldots\ldots(4)$$

N = total number of operations;

where: LT = lead time of the part;

QT = queueing time;

MCT = machining time;

SUT = Setting up time;

and subscript "i" denotes the machine concerned.

Thus, given the total number of operations of a part, the mean lead time and the standard deviations of queueing times, etc., a set of normally distributed lead times of the part can be generated.

By comparing this to a similar set generated by a lead time prediction method, one may conclude on the stability of the method: if the two sets lie close to each other, the method is stable.

In the investigation of the stability of the adaptive lead time method, Davies found that the two sets of lead times "exhibited very similar behaviour".

In this research, however, the stability of the three adaptive systems were tested by running the respective systems for 200 shifts of steady-state operation, and applying a step increase of about 37% in the weekly demand of the parts at the 201th shift and maintaining the increased demand onwards.

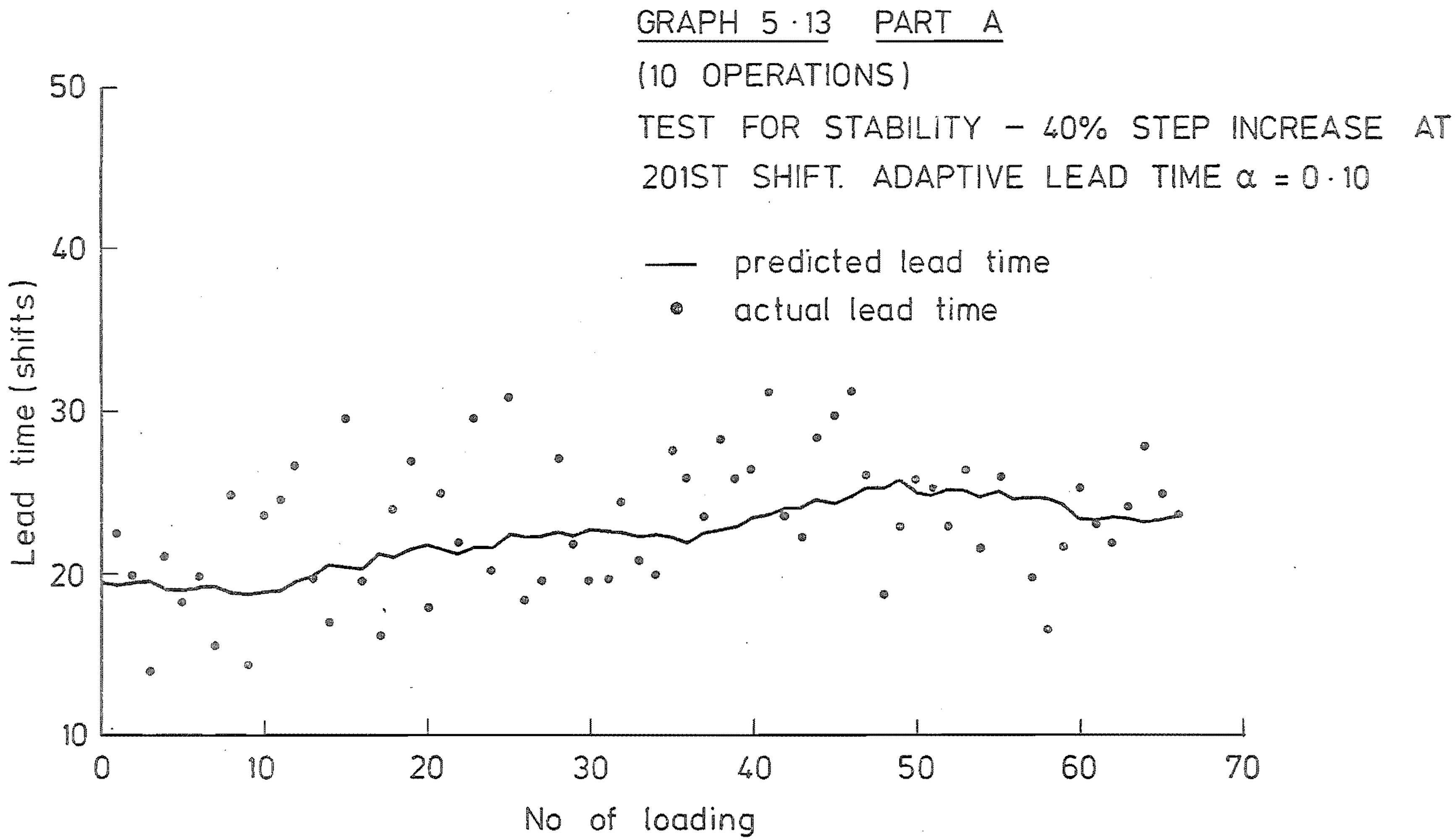
Graphs 5-23, 5-25 and 5-27 showed the lead times of a part of 10 operations (Part A) plotted against the number of loading, i.e. the occasion the part was loaded, for the various adaptive systems.

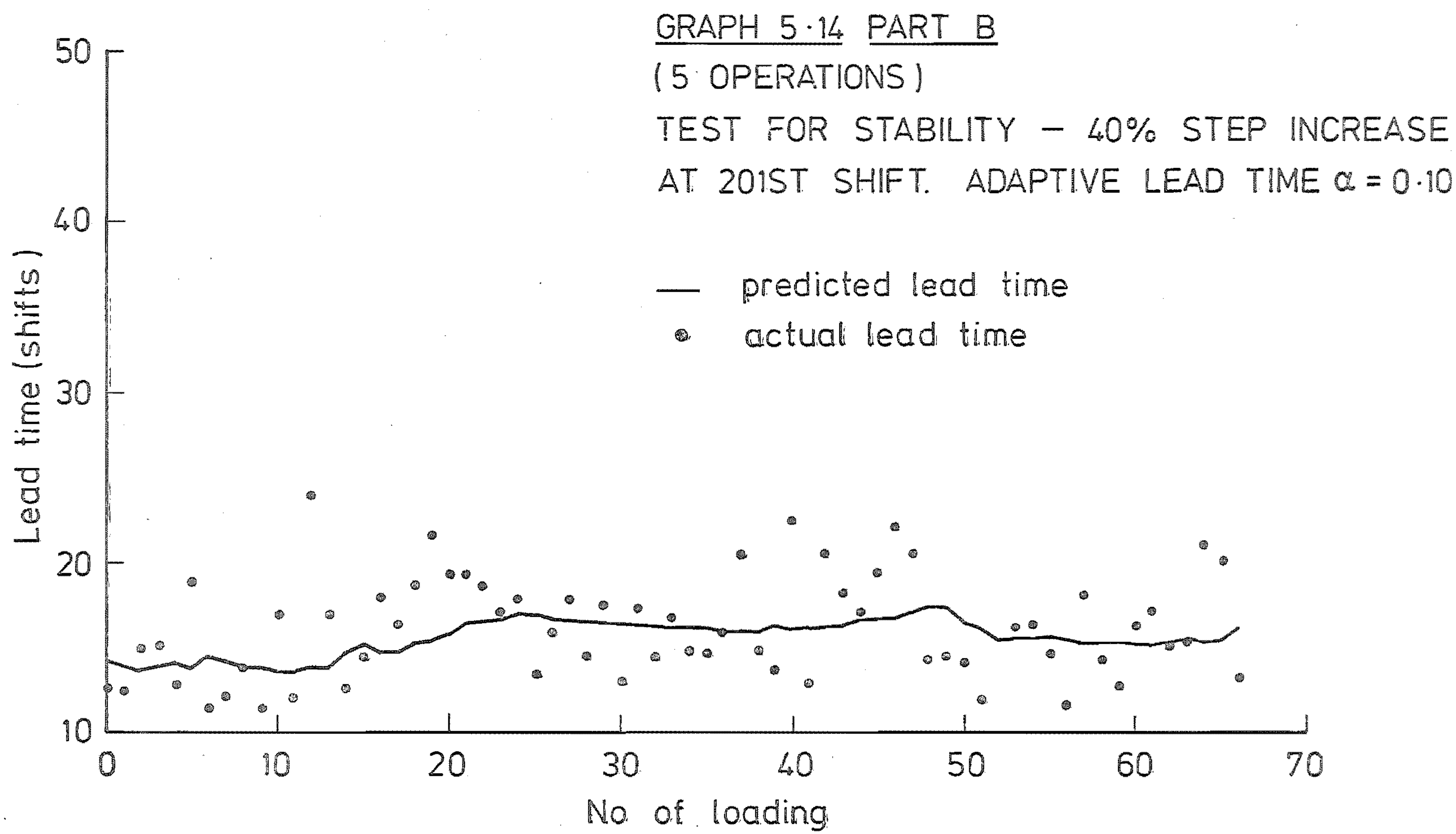
Graphs 5-24, 5-26 and 5-28 showed the same for a part of five operations (Part B).

The shift length used was 8.00 hours. The values of the exponentially smoothing constants used were indicated on the graphs.

In all the graphs, the actual lead time produced by the simulation rises after the step increase, to a new level which appears to be steady-state. The same behaviour is exhibited by the predicted lead time. Hence it may be concluded that the adaptive systems tested are stable.

During the 200 shifts of steady-state operations, part A was loaded approximately 10 times; so was part B. Hence the step increase in demand affected the batches of parts A and B loaded on and after the 11th occasion.

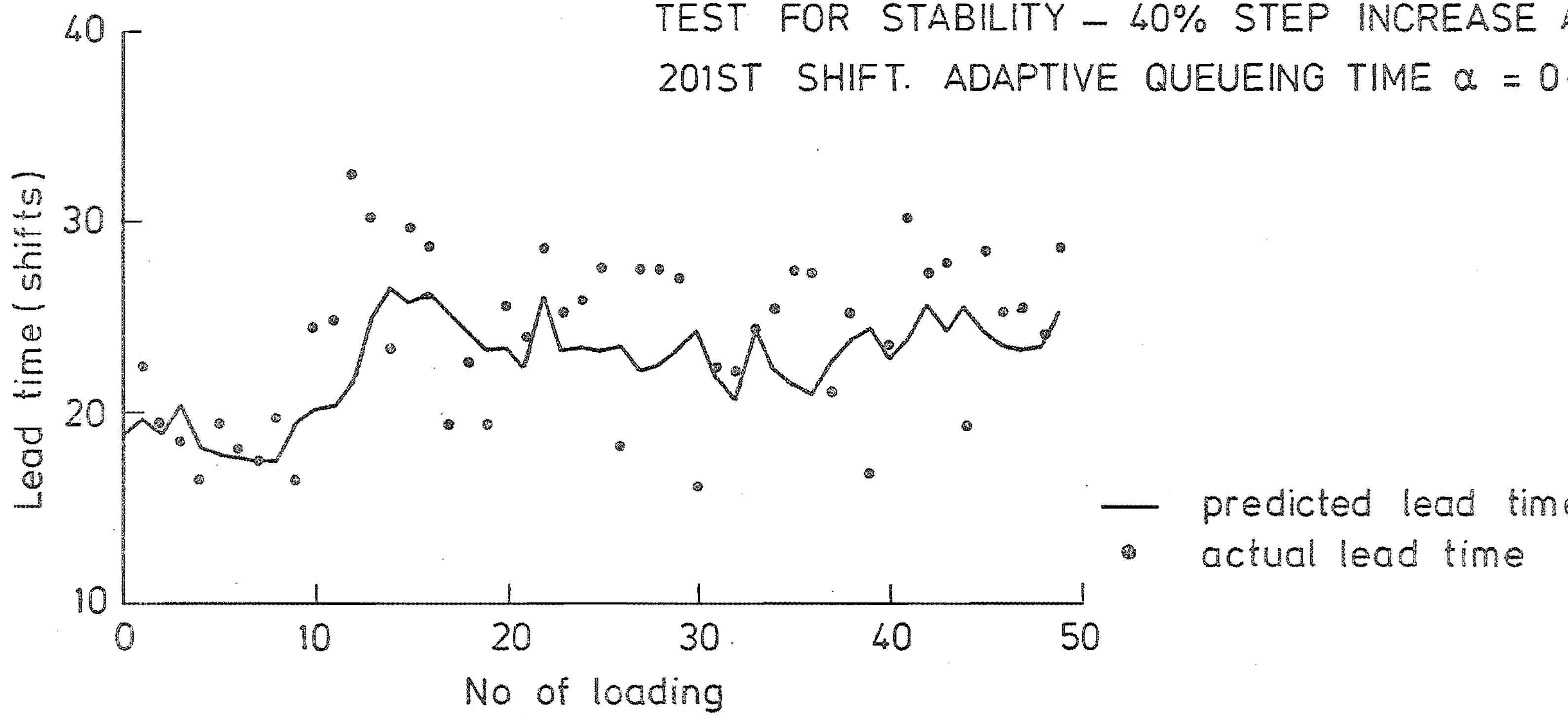

GRAPH 5·13 PART A
(10 OPERATIONS)
TEST FOR STABILITY - 40% STEP INCREASE AT 201ST SHIFT. ADAPTIVE LEAD TIME α = 0·10
predicted lead time
actual lead time
Lead time (shifts)
50
40
30
20
10
0
10
20
30
40
50
60
70
No of loading



GRAPH 5·14 PART B
(5 OPERATIONS)
TEST FOR STABILITY – 40% STEP INCREASE
AT 201ST SHIFT. ADAPTIVE LEAD TIME α = 0·10
predicted lead time
actual lead time
Lead time (shifts)
50
40
30
20
10
0
10
20
30
40
50
60
70
No of loading

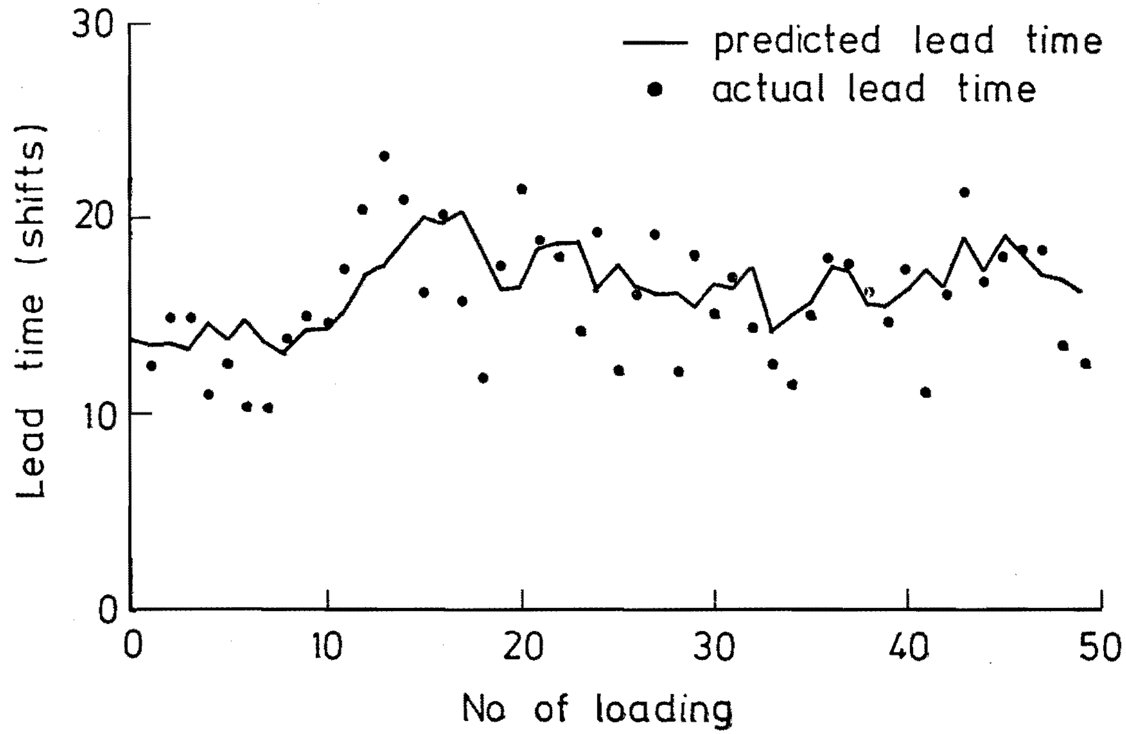
GRAPH 5·15 PART A (10 OPERATIONS)

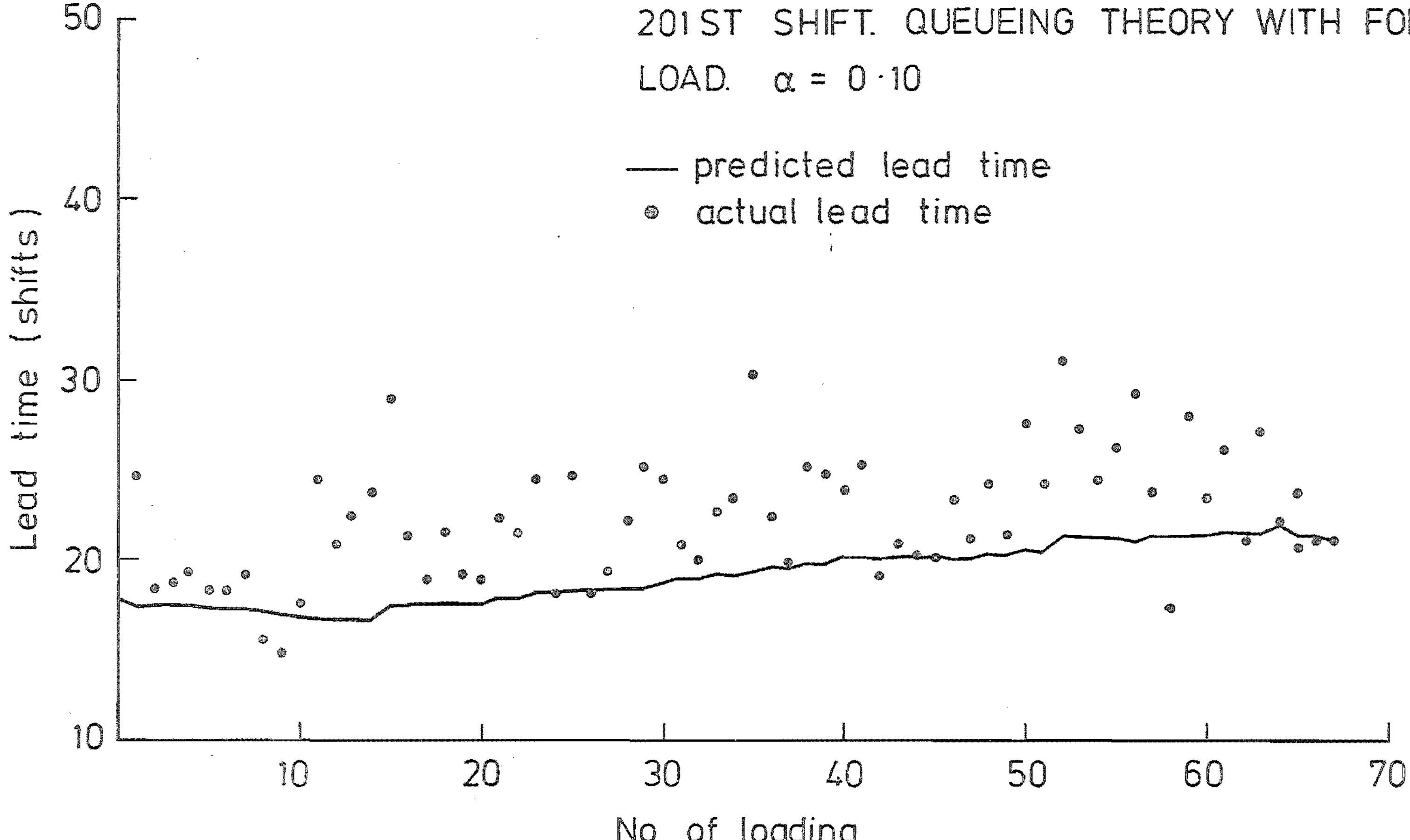
TEST FOR STABILITY – 40% STEP INCREASE AT 201ST SHIFT. ADAPTIVE QUEUEING TIME α = 0·10



GRAPH 5·16 PART B (5 OPERATIONS)

TEST FOR STABILITY – 40% STEP INCREASE AT 201ST SHIFT. ADAPTIVE QUEUEING TIME $\alpha = 0 \cdot 10$

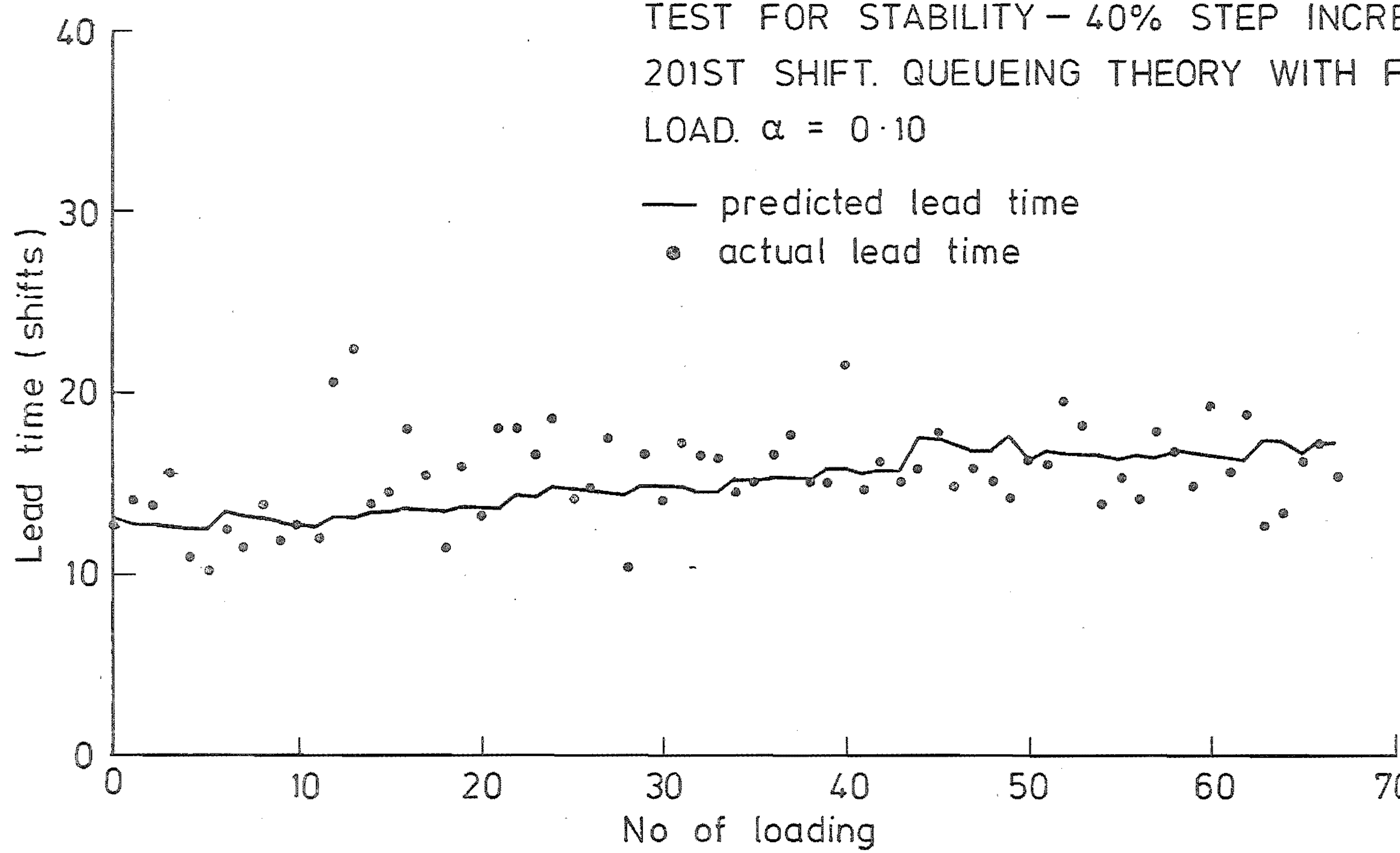




GRAPH 5·17 PART A (10 OPERATIONS)
TEST FOR STABILITY – 40% STEP INCREASE AT 201ST SHIFT. QUEUEING THEORY WITH FORWARD LOAD. $\alpha = 0 \cdot 10$

GRAPH 5·18 PART B (5 OPERATIONS)

TEST FOR STABILITY – 40% STEP INCREASE AT 201ST SHIFT. QUEUEING THEORY WITH FORWARD LOAD. $\alpha = 0 \cdot 10$



CHAPTER 6

EVALUATION OF THE FOUR LEAD TIME PREDICTION METHODS - PART II

6.1 INTRODUCTION

In previous research of methods (a), (b), (c) and (d), the mean of the lead time error histogram was not mentioned.

In this chapter, simulations using different methods will be made, under identical conditions, to produce lead time error histograms for comparison and analysis.

A range of average shop utilisations were simulated.

6.2 TESTS

Data Set 1 was used. This gave rise to an unevenly loaded machine shop (see Chapter 4).

By shortening the shift length from 8.00 hours to 5.00 hours, in steps of 0.50 hour, the average machine shop utilisation increased from 41% to 65%, approximately.

At a shift length less than 5.00 hours, the machine shop's behaviour became unstable.

Each test consisted of 800 shifts of operations. For a machine shop operating five shifts a week, this represents three years' operation.

6.3 RESULTS

The results of the simulation runs are shown in Table 6-1.

The means and the standard deviations of the lead time error histograms produced by the methods are plotted against the average shop utilisation in Graphs 6-1 and 6-2 respectively.

The means and the standard deviations of the lead time histograms are plotted against the same in Graphs 6-3 and 6-4 respectively.

A theoretical curve of the mean of the lead times obtained by using Equation (2), (P. 60), is also plotted in Graph 6-3 for comparison.

To obtain the average queueing time per machine group for each method, at each shift length, the sum of the average queueing times at the machine groups was divided by the number of groups. The results thus obtained are plotted against the average shop utilisation in Graph 6-5.

Note that the range of shop utilisation obtained is much lower than that obtained by Davies. In particular, this machine shop became unstable at a shop utilisation above 65%, whereas Davies' became unstable only at a shop utilisation above 94% or more.

This is due to the fact that the loading on the machine shop used in this simulation is not balanced, whereas Davies' was.

McCallion *et al.*[8] showed that: when a balanced shop was replaced by an unbalanced, the corresponding result occurred at a lower utilisation.

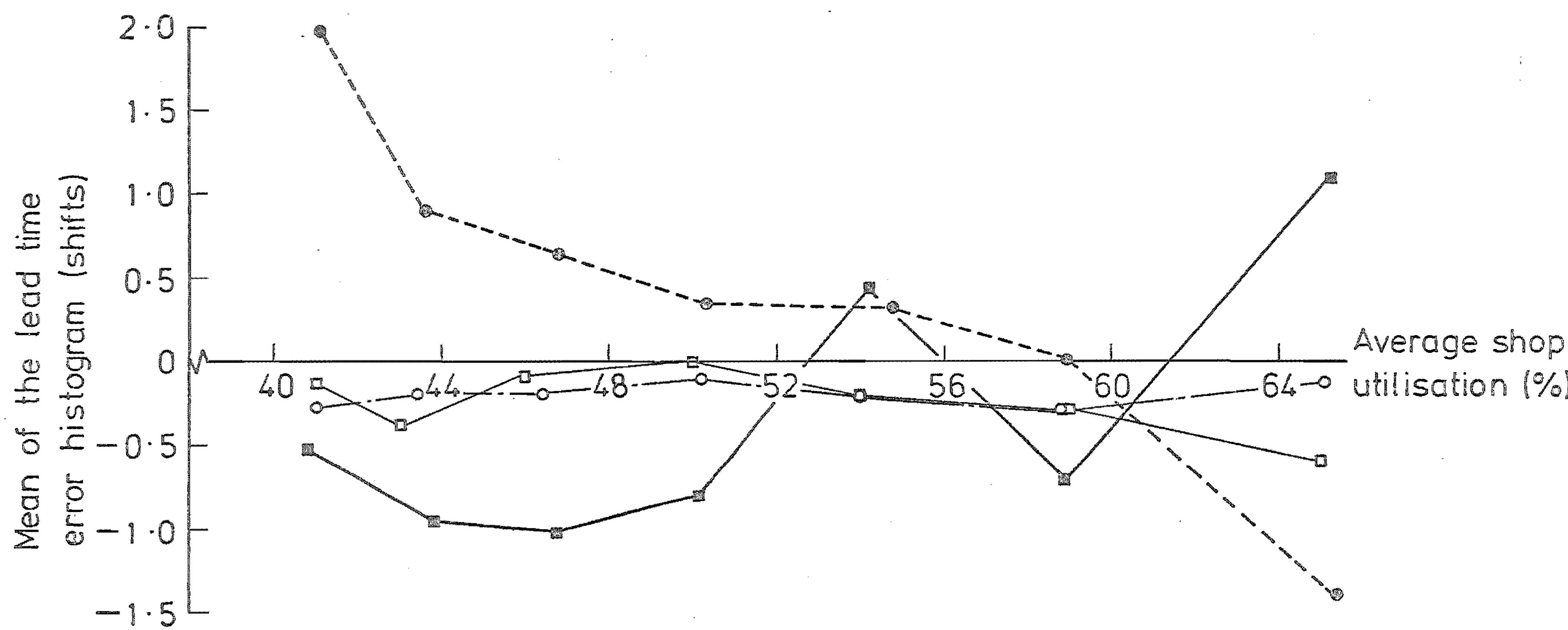
TABLE 6-1.

	Shift length (hrs)		8.00	7.50	7.00	6.50	6.00	5.50	5.00
Constant lead time method	Lead time error histogram	Mean(shifts)	1.94	0.90	0.63	0.34	0.33	0.04	-1.41
		Standard deviation(shifts)	2.50	2.77	3.28	3.69	4.12	4.21	4.56
	Lead time histogram	Mean(shifts)	10.99	11.65	13.00	14.30	17.30	19.60	26.20
		Standard deviation(shifts)	5.10	5.46	6.02	6.79	7.77	8.59	11.20
	Average shop utilisation (%)		41.10	43.68	46.79	50.29	54.74	59.00	65.40
	Average Queueing time (shifts) per machine group		0.65	0.86	1.00	1.11	1.23	1.52	2.05
Adaptive lead time method α =0.10	Lead time error histogram	Mean(shifts)	-0.15	-0.39	-0.11	-0.01	-0.20	-0.30	-0.60
		Standard deviation(shifts)	2.23	2.52	2.80	3.35	3.89	4.46	5.97
	Lead time histogram	Mean(shifts)	10.40	11.60	12.80	14.50	17.10	19.80	28.00
		Standard deviation(shifts)	5.02	5.59	5.96	6.63	8.05	8.74	11.50
	Average shop utilisation (%)		41.00	43.00	46.00	50.00	54.00	59.00	65.00
	Average Queueing time (shifts) per machine group		0.81	0.82	0.87	1.07	1.11	1.58	2.01
Adaptive queueing time method α =0.10	Lead time error histogram	Mean(shifts)	-0.27	-0.21	-0.21	-0.12	-0.20	-0.29	-0.13
		Standard deviation(shifts)	2.23	2.52	3.05	3.72	4.38	4.94	5.59
	Lead time histogram	Mean(shifts)	10.49	11.50	12.68	14.29	16.10	19.90	25.60
		Standard deviations(shifts)	4.88	5.19	5.78	6.79	7.69	8.89	11.20
	Average shop utilisation (%)		40.96	43.38	46.41	50.17	54.07	58.80	65.10
	Average Queueing time (shifts) per machine group		0.75	0.83	0.97	1.00	1.07	1.50	1.97
Queueing theory with forward load method α =0.10	Lead time error histogram	Mean(shifts)	-0.53	-0.94	-1.04	-0.82	0.44	-0.72	1.10
		Standard deviations(shifts)	2.16	2.56	2.89	3.24	3.63	3.97	4.60
	Lead time histogram	Mean(shifts)	9.82	11.70	12.70	15.30	16.60	18.60	27.40
		Standard deviations(shifts)	4.75	5.39	5.78	6.74	7.80	8.42	11.00
	Average shop utilisation (%)		40.79	43.68	46.72	50.10	54.23	58.88	65.21
	Average Queueing time (shifts) per machine group		0.78	0.86	0.93	0.93	1.04	1.36	1.69

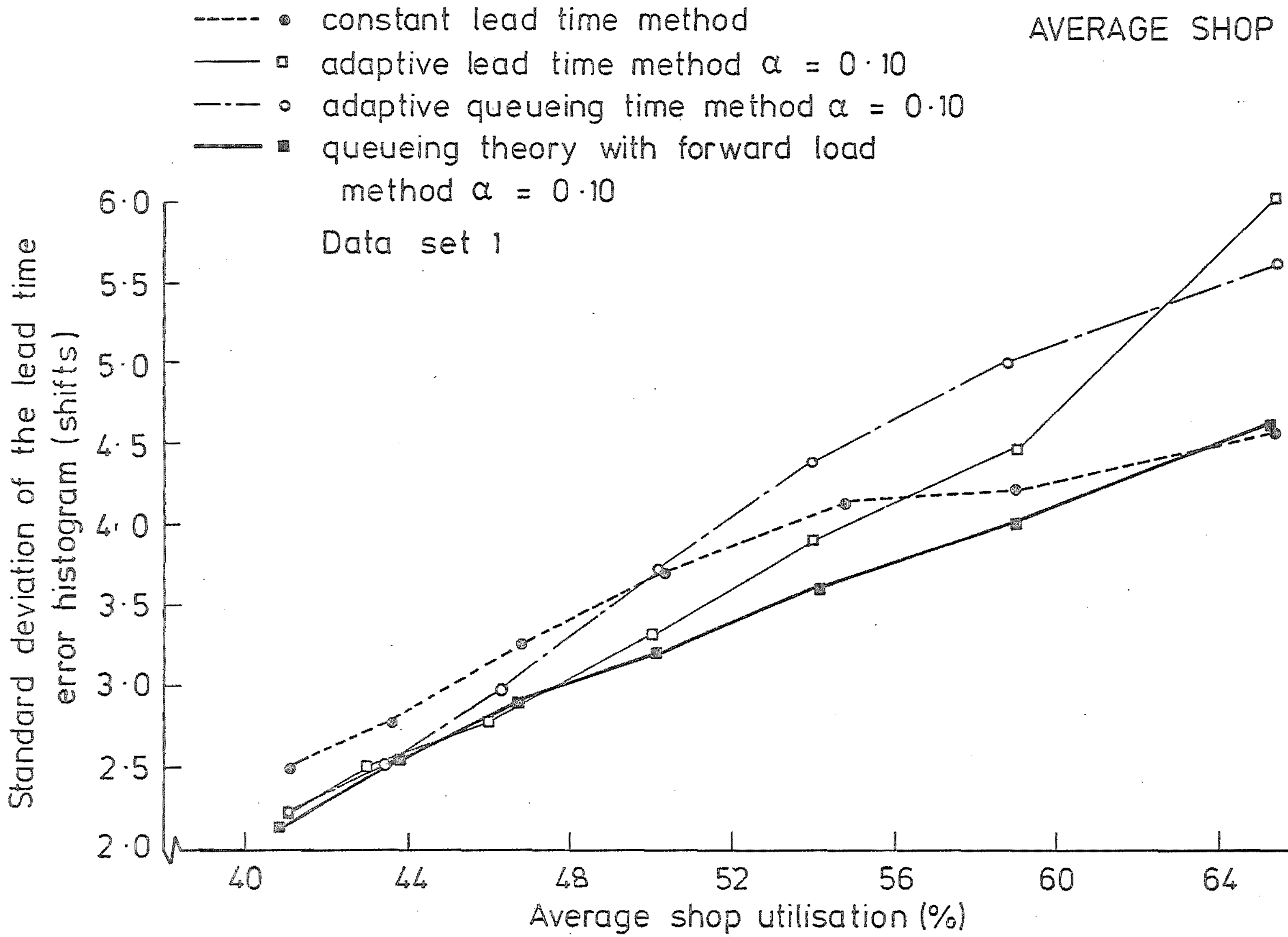
GRAPH 6·1 MEAN OF THE LEAD TIME ERROR HISTOGRAM vs AVERAGE SHOP UTILISATION

- - - - - constant lead time method
——— adaptive lead time method α = 0·10
—·— adaptive queueing time method α = 0·10
——— queueing theory with forward load method α = 0·10

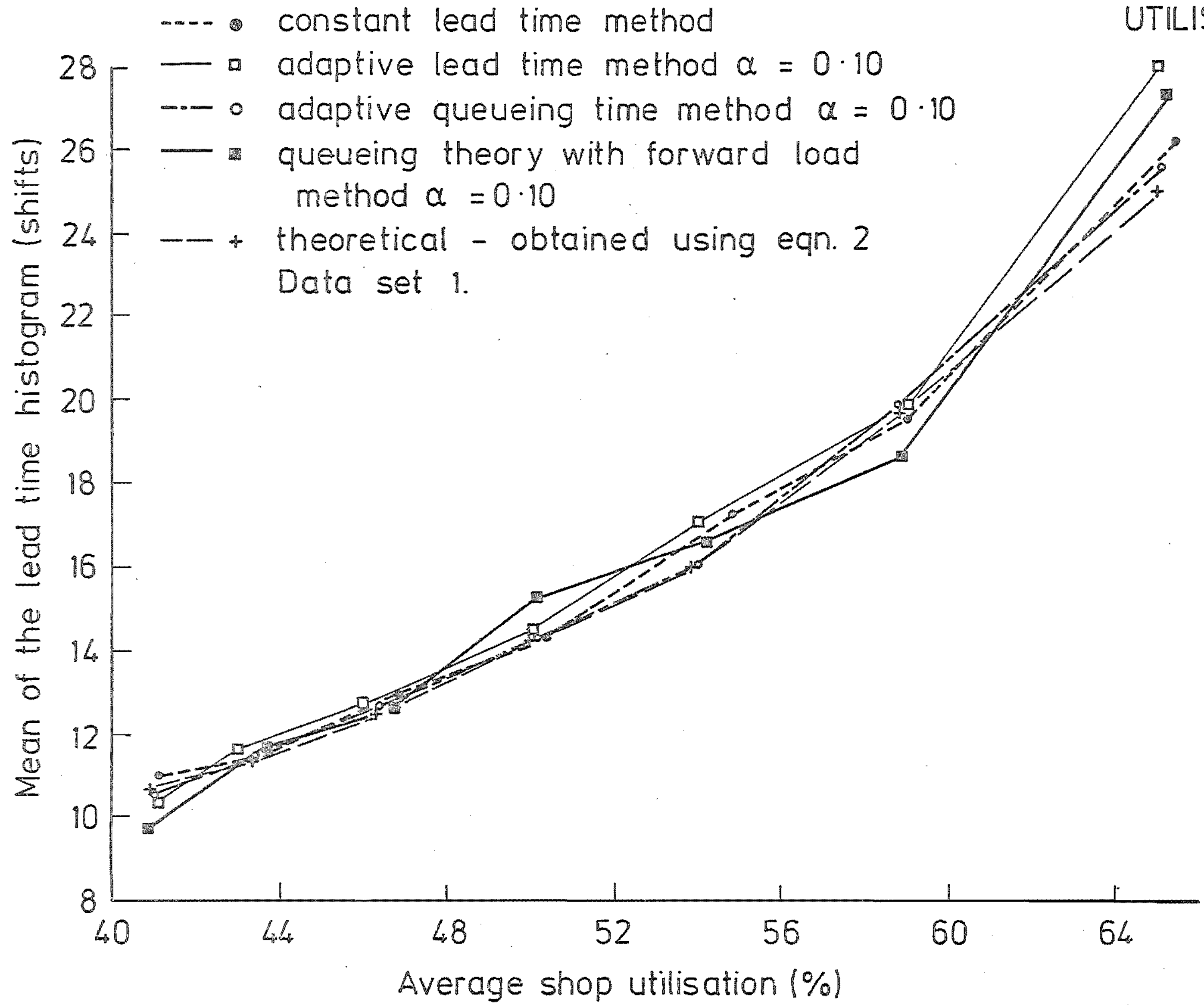
Data set 1.



GRAPH 6·2 STANDARD DEVIATION OF THE LEAD TIME ERROR HISTOGRAM vs AVERAGE SHOP UTILISATION.



GRAPH 6·3 MEAN OF THE LEAD TIME HISTOGRAM vs AVERAGE SHOP UTILISATION.



GRAPH 6·4 STANDARD DEVIATION OF THE LEAD TIME HISTOGRAM vs AVERAGE SHOP UTILISATION.

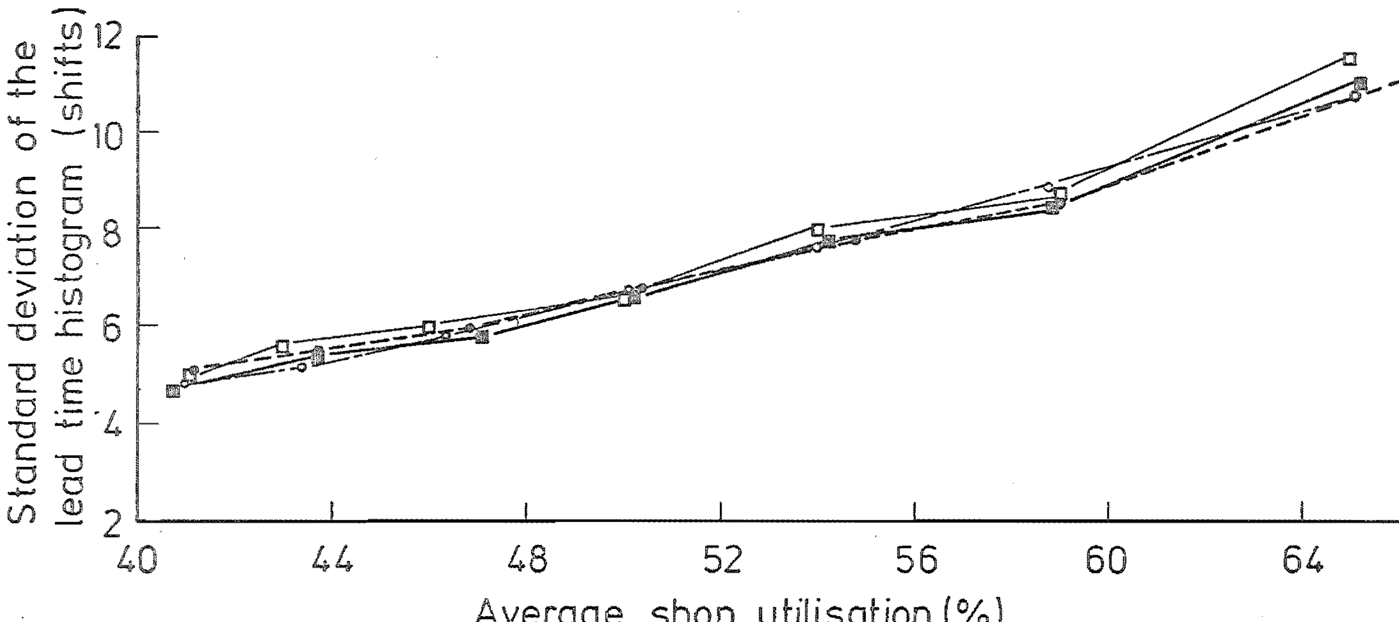
---- ● constant lead time method

—— □ adaptive lead time method α = 0·10

—·— ○ adaptive queueing time method α = 0·10

—— ■ queueing theory with forward load method α = 0·10

data set 1



GRAPH 6·5 AVERAGE QUEUEING TIME vs AVERAGE SHOP UTILISATION

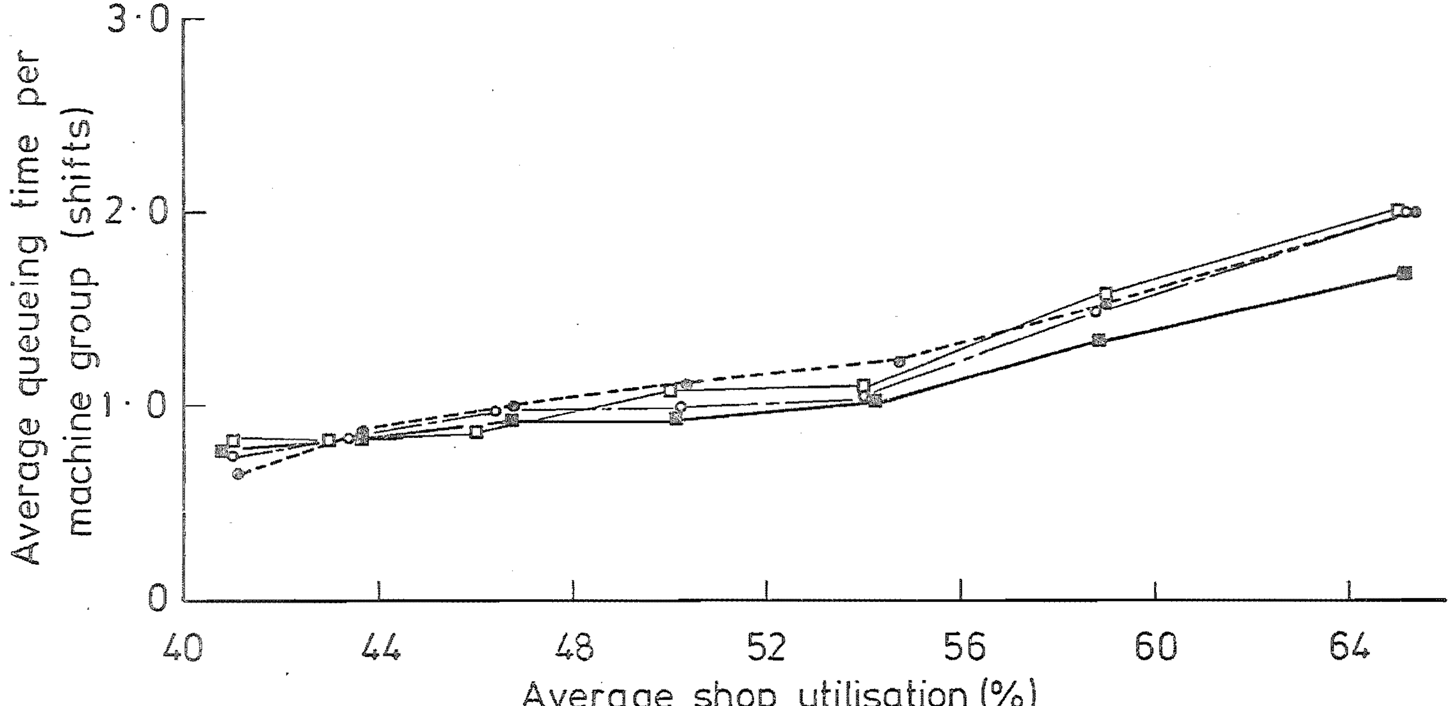
---- ● constant lead time method

—— □ adaptive lead time method α = 0·10

—·— ○ adaptive queueing time method α = 0·10

—— ■ queueing theory with forward load method α = 0·10

Data set 1



6.4 DISCUSSION

With reference to Graph 6-3: —

The four methods produced very similar means of the lead time histograms, at the same shop utilisation.

It can be seen that the mean curves follow closely the one obtained theoretically.

With reference to Graph 6-4: —

The four methods also produced very similar standard deviations of the lead time histograms, at the same shop utilisation.

With reference to Graph 6-2 for the standard deviations of the lead time error histograms: —

The standard deviation curves produced by the four methods are not significantly different.

Recall that in Davies' thesis the standard deviations produced by the constant lead time method differed vastly from those produced by the adaptive lead time method.

The results here showed otherwise. Also contrary to his results, the standard deviations produced by the constant lead time method conform to the general shape according to the classical queueing theory.

So it may be concluded that Davies' standard deviation figures for the constant lead time method are erroneous. Further it seems that he used the procedure described in Chapter 2, to calculate the standard deviations for the constant lead time method, and for the other two methods he investigated.

With reference to Graph 6-1 for the means of the lead time error histograms:—

The mean produced by the constant lead time method becomes zero at (approx.) 61% average shop utilisation.

For average shop utilisations lower than 61%, the means are positive, i.e. on average, batches are finished early. And for average shop utilisation higher than 61%, the reverse is true.

The means produced by the adaptive lead time method are negative throughout the range of shop utilisation tested.

However, except at very high shop utilisations, the means are greater than -0.4 shift.

In other words, on average, the batches are late, but only by 0.4 shift or less, except at very high shop utilisations.

In general, the means produced by this method are closer to zero than the corresponding ones produced by the constant lead time method.

This indicates that the adaptive lead time method predicts lead time more accurately.

The means produced by the adaptive queueing time method are also negative, but all greater than -0.3 shift.

For utilisations up to 54% (approx) the means produced by this method are very similar to the corresponding ones produced by the adaptive lead time method.

Above this utilisation, the adaptive queueing time method predicts more accurately.

The means produced by the queueing theory with forward load method fluctuates between $\pm$ 1 shift.

The large errors are due to the inability of the method to respond to deviations from the predicted conditions, as described in the last chapter.

The system response rates and the accuracy of the lead times predicted by the methods have been investigated and compared.

In the following sections, their effect on the operating cost of a machine shop will be examined.

Assume that the operating cost of a machine shop consists of: the cost of unused machine time, the shortage cost and the work in progress cost.

The unused machine time cost is a function of the machine utilisation, and is not affected by the system response rate or the accuracy of the lead times predicted.

The shortage cost is affected more by the amount of lateness than the accuracy of the lead times predicted, as discussed in Chapter 2.

But is it also affected by the system response rate? And if so, how?

The work in progress cost may be affected by the system response rate, and/or by the accuracy of the lead times predicted. But how?

If any of the relationship in question exists, it would be reflected in Davies' results on costs.

Hence by examining and analysing Davies' results, it may be possible to detect if any of the relationships in question exists, and to establish the exact nature of those relationships that exist.

1. <u>The Shortage Costs</u>

With reference to Tables 2-1, 2-2 and 2-3:–

Davies' results on shortage costs show that:

(a)(i) Up to (approx.) 86% shop utilisation, the adaptive lead time method produced a slightly lower shortage cost than the constant lead time method, at the same shop utilisation.

(ii) Above (approx.) 86% shop utilisation, the amount of reduction in shortage cost increased greatly.

(b)(i) Likewise, up to (approx.) 86% shop utilisation, the adaptive queueing time method produced a lower shortage cost than the adaptive lead time method, at the same shop utilisation.

(ii) Above (approx.) 86% shop utilisation, the amount of reduction in shortage cost increased greatly.

However, if the means and the standard deviations of the lead time error histograms are used to deduce the relative shortage costs, the following are expected:

With reference to Graphs 6-1 and 6-2:-

(a)(i) Up to (approx.) 61% shop utilisation, the adaptive lead time method will give a higher shortage cost than the constant lead time method, at the same shop utilisation.

This is because the standard deviations produced by the two methods are similar, but the means produced by the adaptive lead time method are less than those produced by the constant lead time method, at the same shop utilisation.

Hence a larger amount of lateness is produced by the adaptive lead time method.

(Similar reasonings are used to derive the rest of the deductions).

(ii) Above (approx.) 61% shop utilisation, the reverse will be true.

(b)(i) Up to (approx.) 59% shop utilisation, the adaptive queueing time method will produce a shortage cost similar to that produced by the adaptive lead time method, at the same shop utilisation.

(ii) Above (approx.) 59% shop utilisation, the adaptive queueing time method will give a lower shortage cost than the adaptive lead time method, at the same shop utilisation.

According to McCallion _et_ _al_., when a balanced machine shop is replaced by an unbalanced one, the corresponding result occurs at a lower shop utilisation.

Hence, it is reasonable to assume that the results up to (approx.) 59% or 61% shop utilisation, obtained in this research, correspond to the results up to (approx.) 86% shop utilisation, obtained by Davies; and likewise for results above the shop utilisation figures quoted.

It can be seen that deductions (a)(i) and (b)(i) disagree with Davies' results (a)(i) and (b)(i) respectively, although deductions (a)(ii) and (b)(ii) agree with Davies' results (a)(ii) and (b)(ii) respectively.

In other words, the mean and the standard deviation of the lead time error histogram above cannot account for the relative shortage costs obtained by Davies.

Other factors must be operating.

Obviously, the delay in the loading of a new batch will often result in a shortage, even if the lead time is estimated correctly when the batch is eventually loaded.

Hence the additional factor is thought to be the ability to detect the possibility of a shortage sufficiently early, so that a new batch can be loaded <u>in time</u> to avoid the shortage.

Deduction (b)(i) and Davies' result (b)(i) seem to suggest that for two lead time prediction methods giving the same amount of lateness, the method with a faster system response rate has a better ability to detect possible shortages early.

To explain Davies' result (a)(i):

At any one average shop utilisation, the lead times of parts fluctuate with time.

The fixed lead time estimates used in the constant lead time method may be equal to or larger than most of the actual lead times, for most of the time. But unless the fixed lead time estimates used are very large, there are occasions where the actual lead times exceed the fixed estimates.

Very often, the failure to load new batches immediately in these occasions will result in shortages.

The constant lead time method cannot assess shortage/surplus according to the changing levels of the actual lead times, and therefore cannot detect possible shortages in these occasions.

Although the batches in question will be loaded eventually, the delay of the loading, in itself, will result in shortages.

On the other hand, due to its adaptive nature the adaptive lead time method is able to detect shortages in these occasions, as well as in all the other occasions, and load new batches accordingly.

Hence, although the constant lead time method gives a smaller amount of lateness than the adaptive lead time method, the former's inability to load new batches immediately in the occasions mentioned resulted in an overall shortage cost higher than that of the latter.

The ability to detect shortages early can also be used to account for part of the large reductions in the shortage cost results (a)(ii) and (b)(ii). (The rest of the reductions is due to the differences in the amount of batches finished late)..

Davies used buffer stock in his research.

However, the buffer stock used is expected to give the methods, the same degree of protection against shortages.

Hence the foregoing reasoning is not affected by the use of buffer stock in his simulation.

2. <u>Work in Progress Cost</u>

The work in progress cost provided by Davies is the sum of the cost due to the finished parts in the stores, and the cost due to the work in progress in the shop.

Since neither of the two costs is given, analysis is made difficult.

The demand on parts is fixed, so the same number of parts will be manufactured no matter which method is used to predict lead times, over a sufficiently long period. But if one method gives higher average waiting times than another, it will produce a larger amount of work in progress in the shop.

With reference to Graph 6-5:—

The four methods have very similar average waiting times per machine group, at the same utilisations.

Hence, the amount of work in progress in the shop is expected to be the same for the methods.

Davies' result is not expected to differ.

The amount of finished parts in the stores depends on the amount of batches finished late and the level of buffer stocks.

The buffer stock used by Davies was two hours per operation per part.

Assuming on average, there are four operations per part, then, on overage, the buffer stock is eight hours per part.

At a shift length of three hours, on average, the buffer stock will reduce shortages due to lateness of up to 2.7 shifts, and at a shift length of two hours, up to 4 shifts.

Thus, the margin of protection against lateness is set higher at a shorter shift length, where the utilisation is higher.

As a result, the storage cost will be higher at a higher utilisation.

With reference to Tables 2-1, 2-2 and 2-3:-

Both the adaptive lead time method and the adaptive queueing time method produced "work in progress cost" that increased with increasing utilisation.

This can be explained by the increase in storage cost due to the increase in buffer stock with increasing utilisation.

Note that the average queueing time increases with increasing utilisation.

Thus, the cost due to the work in progress in the shop will also increase with increasing utilisation.

The reversing of this trend found with the constant lead time method, is thought to be caused by: the rapid decrease of the mean of the lead time error histogram with increasing average shop utilisation from a fairly large positive value, to a fairly large negative value.

In other words, with increasing average shop utilisation, the situation changes from one with a large number of batches finished early, to one with a large number of batches late.

As a result the storage cost falls rapidly with increasing utilisation.

CHAPTER 7

THE EFFECT OF CHANGING THE JOB MIX ON THE MEAN AND THE STANDARD DEVIATION OF THE LEAD TIME ERROR HISTOGRAM

7.1 INTRODUCTION

It is possible that the results obtained in Chapter 6 are dependent on the characteristics of the job mix used.

This Chapter will investigate the effect of varying the characteristics of the job mix, on the mean and the standard deviation of the lead time error histogram.

7.2 TESTS

Only the adaptive queueing time method was used for the investigation.

The tests involved were very similar to those described in Chapter 6. The only variant being the data set.

Data Sets 1, 2 and 3 were used to investigate the effect of reducing only the mean, the effect of reducing only the standard deviation, and the effect of reducing both, of the lead time histogram.

7.3 RESULTS AND DISCUSSION

The results are shown in Tables 7-1 and 7-2.

Graph 7-1 shows the mean of the lead time error histogram

TABLE 7-1.

Data Set 2.

Adaptive Queueing Time Method $\alpha = 0.10$

Shift length (hours)		8.00	7.50	7.00	6.50	6.00	5.50
Lead Time error histogram	Mean(shifts)	-0.13	-0.18	-0.17	-0.24	-0.37	-0.21
	Standard deviation (shifts)	2.37	2.82	3.19	3.57	4.41	5.00
Lead time histogram	Mean(shifts)	10.90	12.20	13.90	16.30	19.20	22.70
	Standard deviation (shifts)	3.47	3.84	4.67	5.73	6.97	7.89
Average shop utilisation (%)		45.20	48.12	51.54	55.61	60.11	65.30
Average queueing time (shifts)		0.5	0.8	0.97	1.14	1.48	1.74

TABLE 7-2.

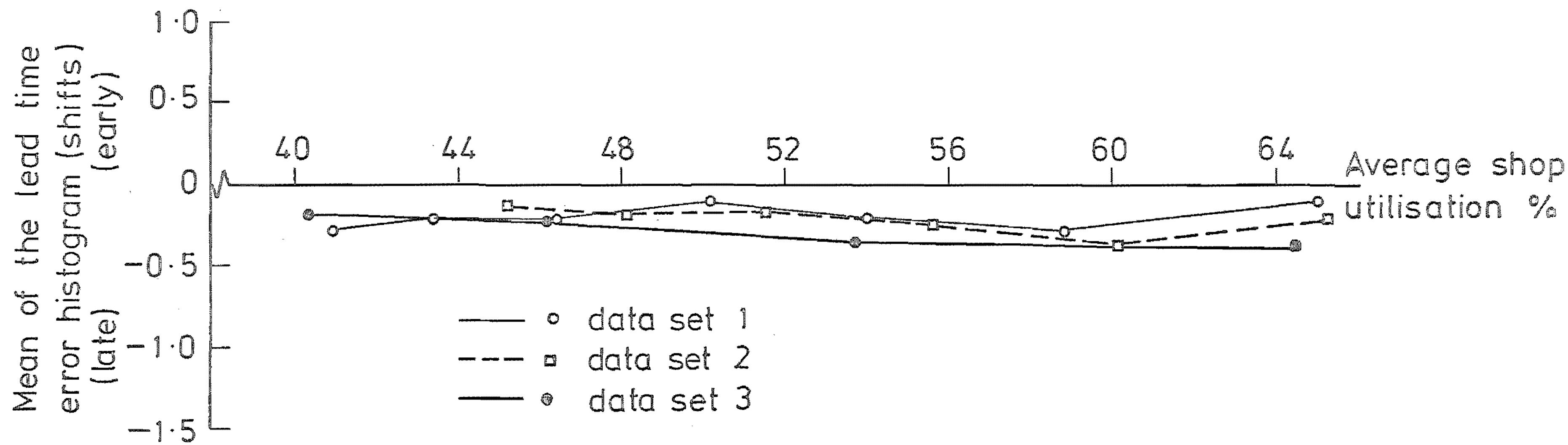
Data Set 3.

Adaptive queueing time method $\alpha = 0.10$.

Shift length (hrs)		8.00	7.00	6.00	5.00
Lead time error histogram	Mean(shifts)	- 0.17	- 0.22	- 0.34	- 0.38
	Standard deviation (shifts)	2.13	2.58	3.33	5.02
Lead time histogram	Mean(shifts)	10.20	12.60	16.20	24.00
	Standard deviation (shifts)	2.97	3.92	5.10	8.70
Average shop utilisation (%)		40.36	46.15	53.73	64.47
Average queueing time (shifts)		0.74	0.71	1.54	2.39

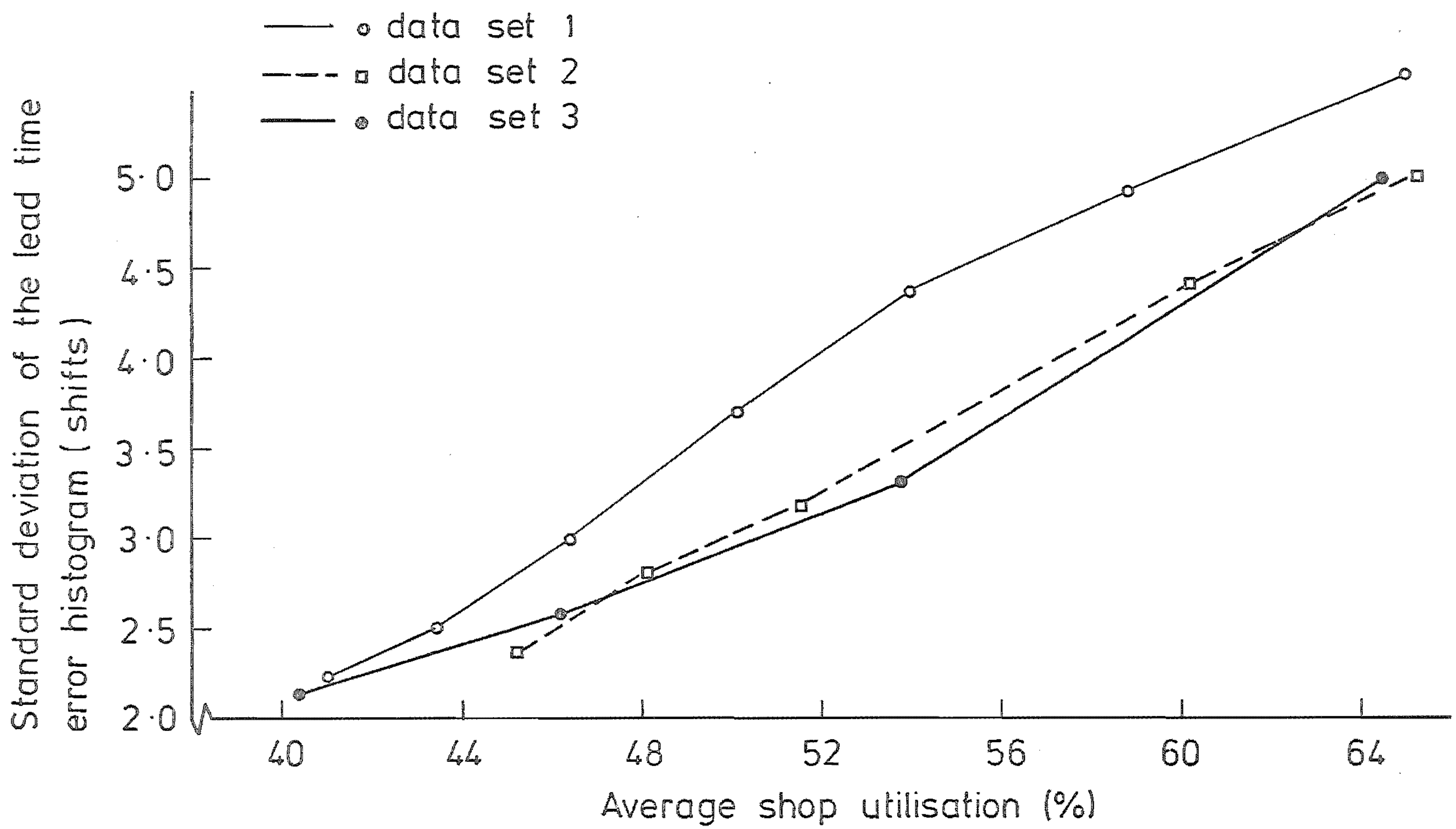
GRAPH 7·1 MEAN OF THE LEAD TIME ERROR HISTOGRAM vs AVERAGE SHOP UTILISATION

ADAPTIVE QUEUEING TIME METHOD $\alpha = 0 \cdot 10$

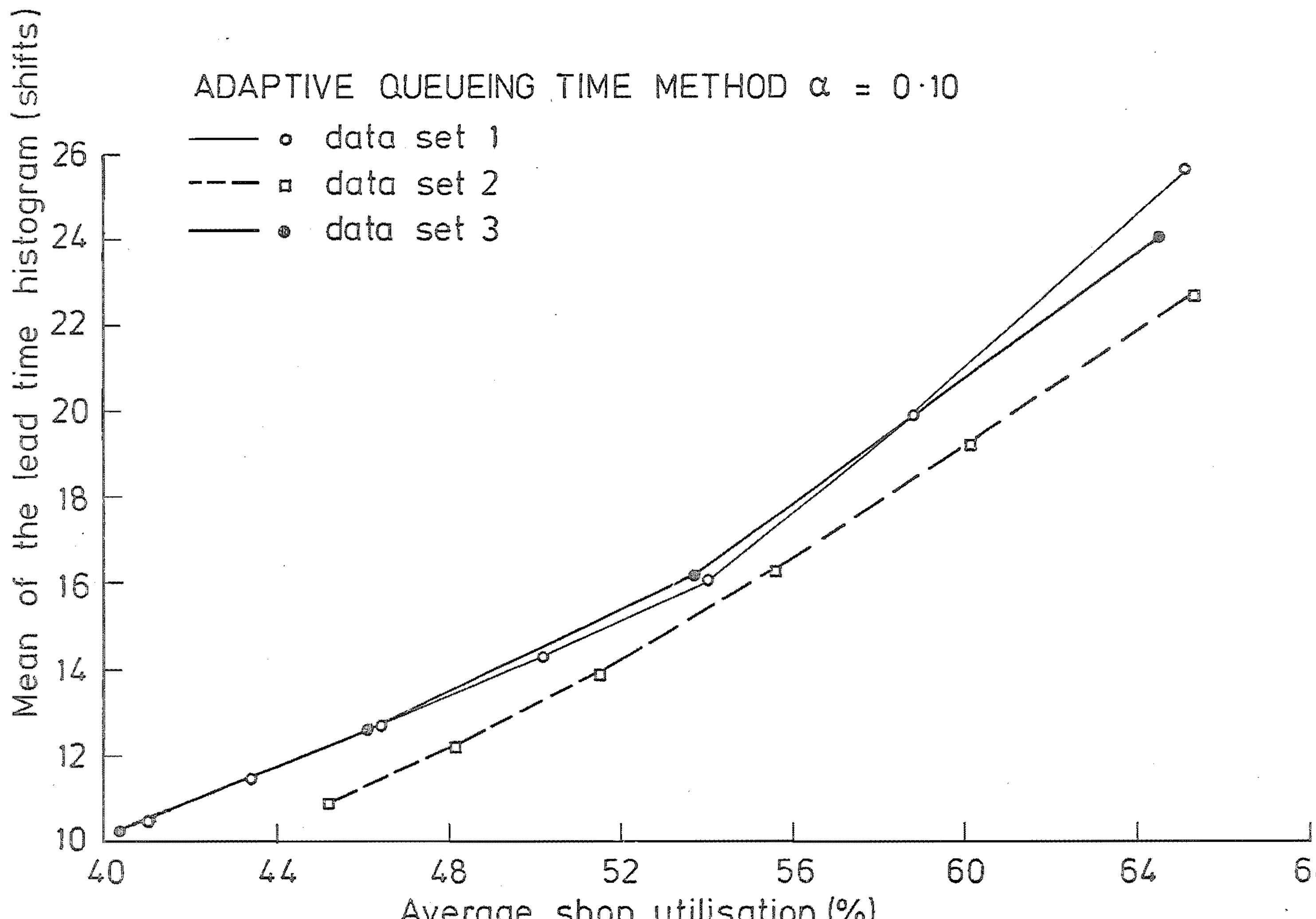


GRAPH 7·2 STANDARD DEVIATION OF THE LEAD TIME ERROR HISTOGRAM vs AVERAGE SHOP UTILISATION.

ADAPTIVE QUEUEING TIME METHOD α = 0·10



GRAPH 7·3 MEAN OF THE LEAD TIME HISTOGRAM vs AVERAGE SHOP UTILISATION.



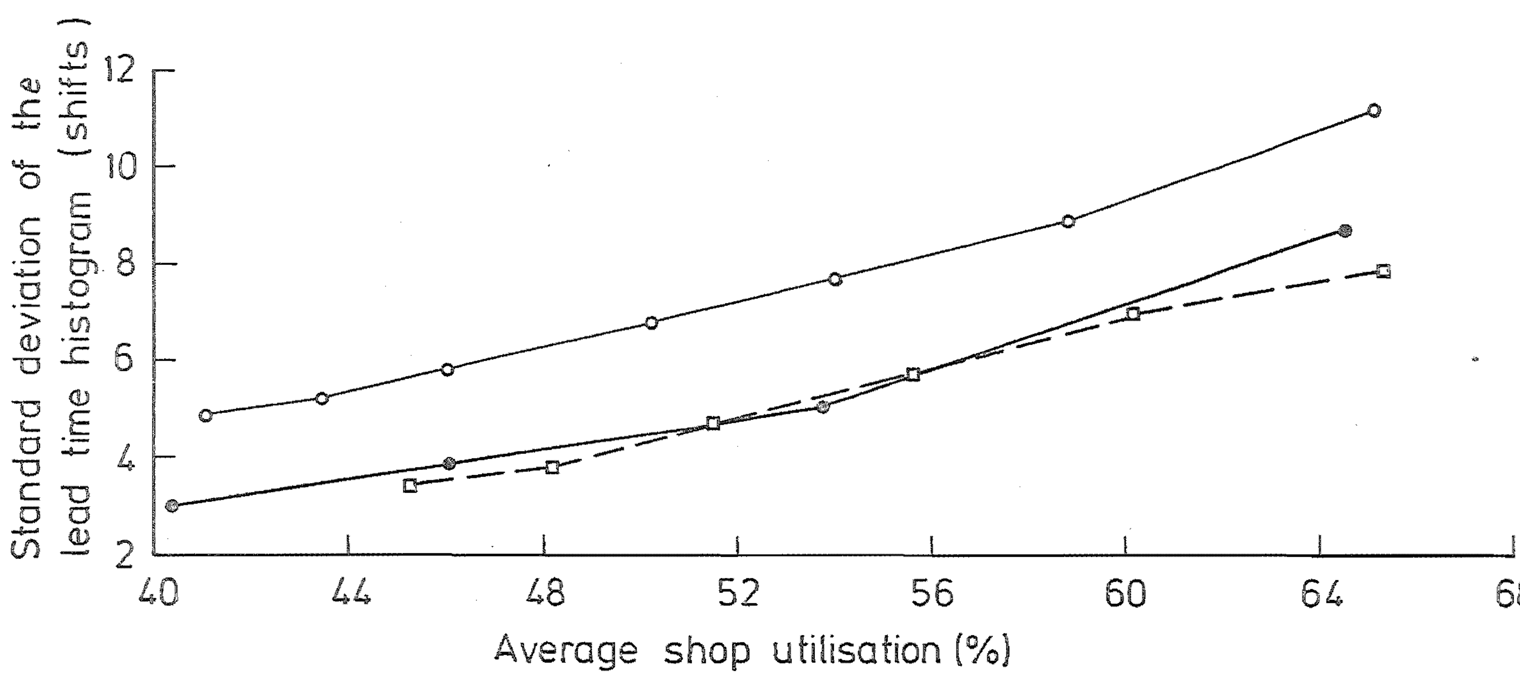
GRAPH 7·4 STANDARD DEVIATION OF THE LEAD TIME HISTOGRAM vs AVERAGE SHOP UTILISATION

ADAPTIVE QUEUEING TIME METHOD α = 0·10

—— o data set 1

– – – ▫ data set 2

—— ● data set 3



plotted against the average shop utilisation, for the three data sets.

Graph 7-2 shows the standard deviation of the lead time error histogram plotted against the same, for the three data sets.

Graph 7-3 shows the mean of the lead time histogram plotted against the same, for the three data sets.

And Graph 7-4 shows the standard deviation of the lead time histogram plotted against the same, for the three data sets.

From Graphs 7-3 and 7-4, it can be seen that Data Set 2 gives a lower mean and a lower standard deviation of the lead time histogram than Data Set 1, at the same average shop utilisation.

And Data Set 3 gives a lower standard deviation of the lead time histogram than Data Set 1, but a similar mean at the same average shop utilisation.

Also, Data Set 3 gives a higher mean of the lead time histogram than Data Set 2, but a similar standard deviation, at the same average shop utilisation.

With reference to Graph 7-1:—

The three different data sets give very similar means of the lead time error histograms, at the same average shop utilisation.

In other words, changing the mean and/or the standard deviation of the lead time histogram has no effect on the mean of the lead time error histogram.

With reference to Graph 7-2:-

Data Set 3 gives a lower standard deviation of the error histogram than Data Set 1, at the same average shop utilisation.

The decrease of the standard deviation is caused by the decrease of the standard deviation in lead time histogram, since the two data sets give similar means of the lead time histograms, at the same average shop utilisation.

Data Sets 2 and 3 give very similar standard deviations of the error histograms, at the same average shop utilisation.

As seen earlier, the two data sets gives very similar standard deviations, but very different means of the lead time histograms, at the same average shop utilisation.

Hence, changing only the mean of the lead time histogram has no effect on the standard deviation (or the mean) of the error histogram.

Like Data Set 3, Data Set 2 also gives a lower standard deviation of the error histogram than Data Set 1, at the same average shop utilisation.

As seen earlier, Data Set 2 gives a lower mean, as well as a lower standard deviation, of the lead time histogram than Data Set 1, at the same average shop utilisation.

However since:

(i) the reduction in the standard deviation of the lead time histogram by Data Set 2 is similar to the reduction by Data Set 3, and

(ii) the reduction in the standard deviation of the error histogram by Data Set 2 is similar to the reduction by Data Set 3,

it is the reduction in the standard deviation, not the mean, of the lead time histogram that has caused the reduction in the error histogram.

It was found that with the reduction in the standard deviation of the lead time histogram, the standard deviations of the service time distributions at the machine groups were reduced.

Hence, it can be further concluded that a reduction in the standard deviation of the lead time histogram caused a reduction in the standard deviation of the error histogram.

CHAPTER 8

THE DYNAMIC BEHAVIOUR OF THE ADAPTIVE QUEUEING TIME METHOD

8.1 INTRODUCTION

The four lead time prediction methods have been tested for their system response rates and accuracy in Chapters 5 and 6.

The test for accuracy was performed using a machine shop in a steady-state condition.

However, in real life, due to fluctuating demands and other disturbances, a steady-state condition is seldom achieved in a machine shop.

Therefore, before any of the methods can be implemented for a real life machine shop, its performance in a dynamic situation must be tested, and found acceptable.

Of the four methods tested, the adaptive queueing time method has shown to be the most promising.

Under a steady-state condition, it produced very accurate lead time estimates.

Even at very high shop utilisations, where all the other methods performed badly, the adaptive queueing time method still produced very accurate lead time estimates.

Further, in terms of the total operating cost of a machine shop, Davies has shown that the adaptive queueing time method performed better than the constant lead time method and the adaptive lead time method.

Hence, the adaptive queueing time method is investigated further, for its performance in the dynamic operation of a machine shop.

In this chapter, a machine shop using the adaptive queueing time method will be subjected to different types of fluctuating demands.

The means and the standard deviations of the lead time error histograms obtained will be analysed.

8.2 THE ADAPTIVE RESPONSE RATE

In forecasting systems using exponential smoothing, it is customary to use a small value (e.g. 0.2 or less) for the exponential smoothing constant, α, to filter out the major part of the noise in the input.

However, when the system encounters a sudden genuine change in the underlying process, with a low value of α, it will take a considerably long time to home in to the new level. Biased forecasts will occur, and will continue for some time.

Trigg[11] proposed a method which will automatically increase the value of α, when forecasts go out of control. This gives more weight to recent data, and hence more rapid homing in to the new situation.

Once the system has homed in, it will also automatically reduce the value of α , in order to filter out the noise.

Trigg's method is known as the "Exponential Smoothing With Adaptive Response Rate".

It computes α as follows:

$$\alpha = \text{modulus} \left(\frac{\text{Smoothed error}}{\text{Smoothed absolute error}} \right)$$

where: new smoothed error

$= \gamma \text{ x error } + (1-\gamma) \text{x(old smoothed error)}$

new smoothed absolute error

$= \gamma \text{ x(absolute error)} + (1-\gamma) \text{x(old smoothed absolute error)}$

and error = (actual value of series at time t)

− (predicted value of series for time t),

= an exponential smoothing constant with a fixed value.

γ is another exponential smoothing constant, $(0 \leq \gamma \leq 1)$

It is felt that for a dynamic situation, it is more beneficial to use the adaptive response rate than the fixed.

Hence, the adaptive response rate will be incorporated into the adaptive queueing time method.

And this version of the adaptive queueing time method will be tested against the original version, under identical dynamic conditions.

A test under identical steady-state conditions at a shift length of 8.00 hours will also be performed.

By using the adaptive response rate instead of the fixed, the system response rate of the adaptive queueing time method will be increased.

To show the increase in the system response rate, a first generation simulation run for the modified version of the adaptive queueing time method was performed, using conditions identical to those used for testing the original version in Chapter 5.

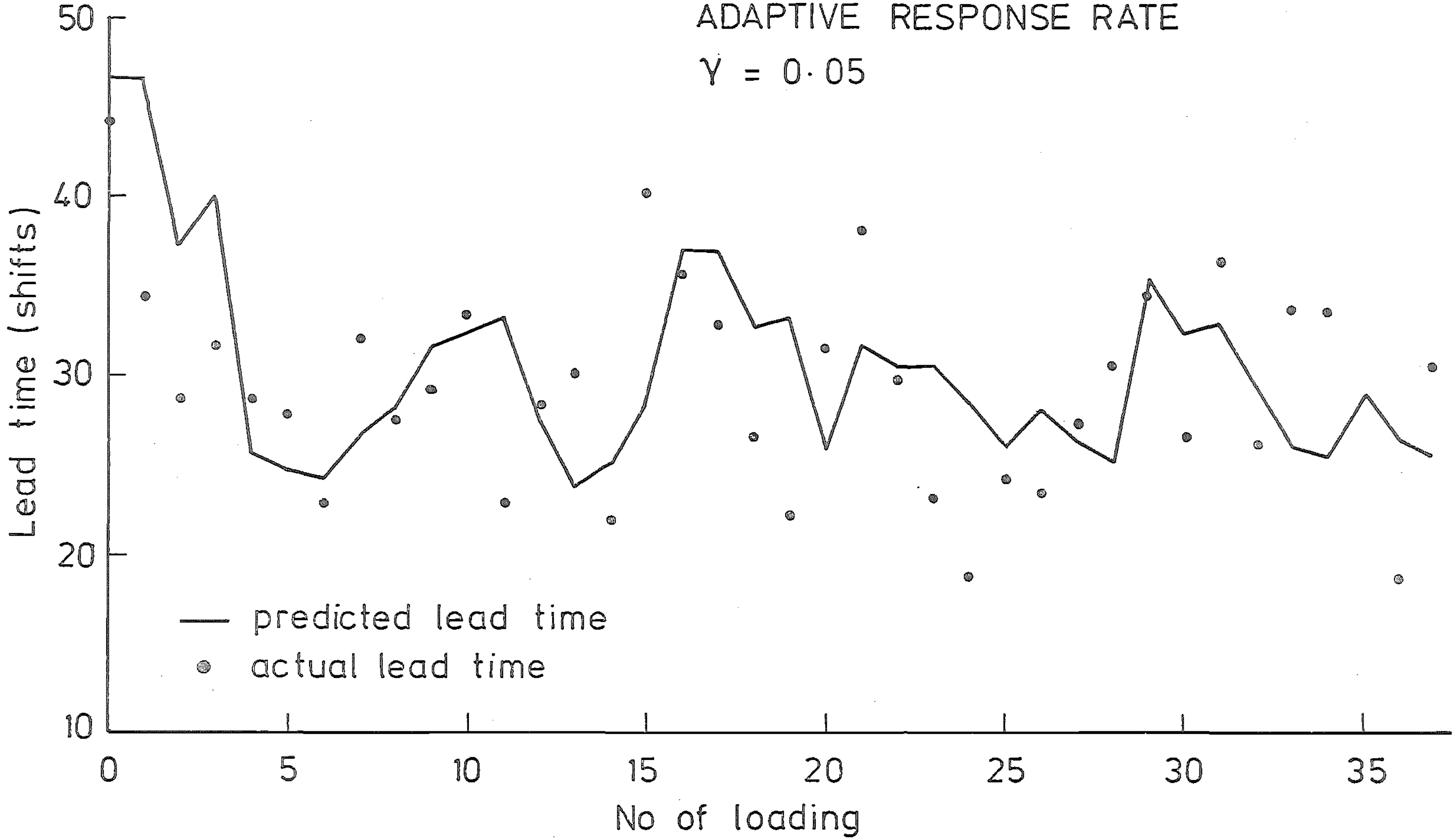
The value used for γ was 0.05.

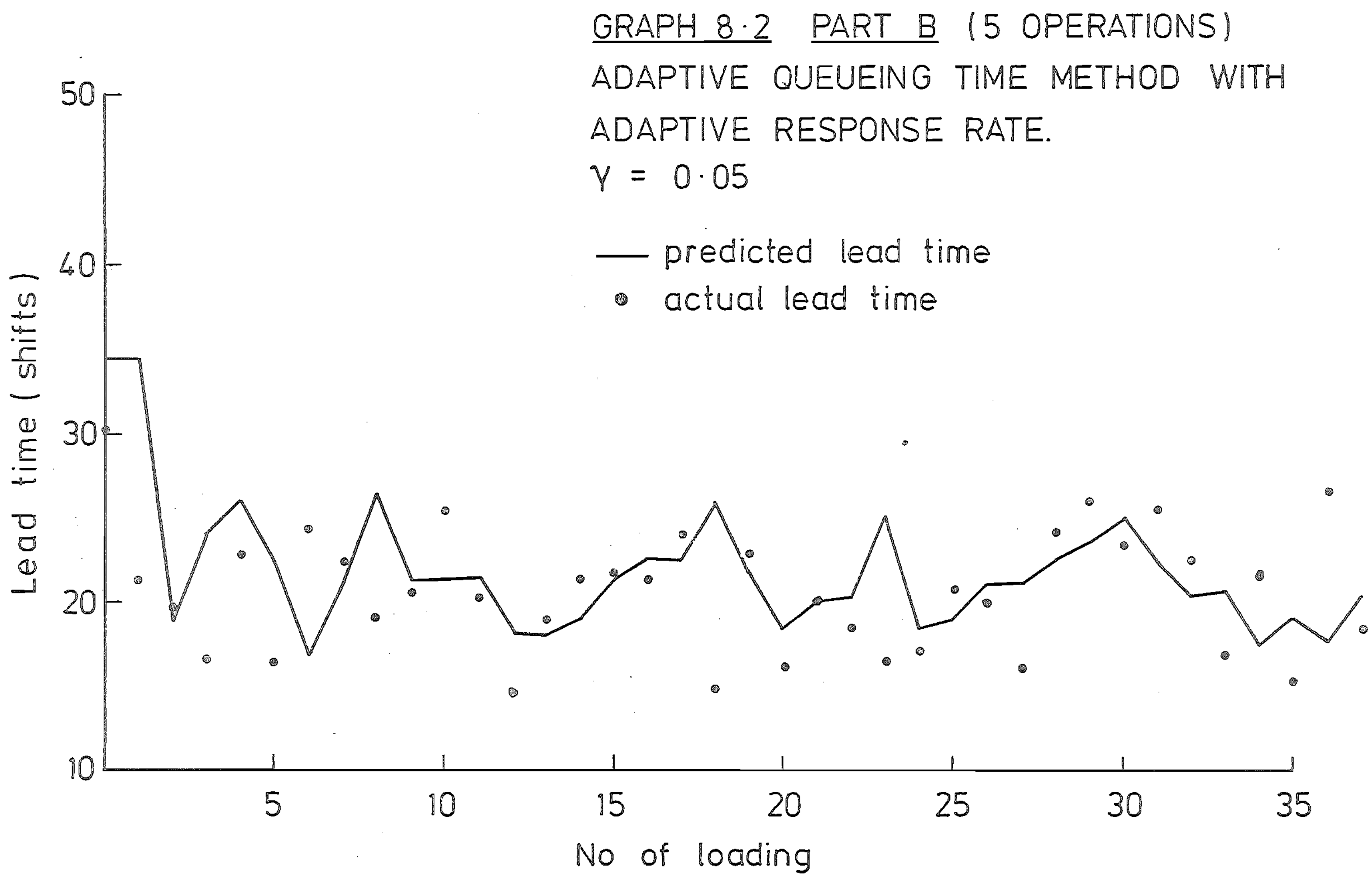
Graphs 8-1 and 8-2 show the actual and the predicted lead times of two typical parts.

It can be seen that the lead time of Part A is estimated correctly at the 8th attempt, which is the same as for the original version.

However, the lead time of Part B is estimated correctly at the 2nd attempt, a reduction of seven attempts over the original version.

GRAPH 8·1 PART A (10 OPERATIONS)
ADAPTIVE QUEUEING TIME METHOD WITH
ADAPTIVE RESPONSE RATE
$\gamma = 0\cdot05$




GRAPH 8·2 PART B (5 OPERATIONS)
ADAPTIVE QUEUEING TIME METHOD WITH
ADAPTIVE RESPONSE RATE.
γ = 0·05
predicted lead time
actual lead time
Lead time (shifts)
No of loading
10
20
30
40
50
5
10
15
20
25
30
35

8.3 TESTS

To subject the machine shop to dynamic changes, the following patterns of demands on parts were used:

(a) STEP DEMAND

With a step increase of: (i) 20%, (ii) 40% and (iii) 60% on the original demand of each part.

The step demand was applied at the start of the simulations.

(b) SINUSOIDAL DEMAND

With a period of: (i) 20 shifts, (ii) 40 shifts, (iii) 120 shifts and (iv) 240 shifts.

On the basis of five shifts a week, these correspond to (approx.) (i) monthly, (ii) 2 monthly, (iii) half-yearly and (iv) yearly cycles respectively.

The (peak-to-trough) amplitude was 120% of the original demand. This was superimposed on the demand, so that the highest and the lowest demands in the simulations were 160% and 40% respectively of the original demand.

The sinusoidal demand was also applied at the start of the simulations.

The initial state for each simulation run was steady-state obtained as described in Chapter 4.

Each test was run for 800 shifts, at a shift length of 8.00 hours.

Data Set 1 was used.

For the adaptive queueing time method with fixed response rate, α = 0.10 was used.

For the version with adaptive response rate, γ = 0.05 and γ = 0.10 were used.

8.4 RESULTS AND DISCUSSION

The results are tabulated, as in Table 8-1.

With reference to the results for the adaptive queueing time method with fixed response rate, in Table 8-1:-

As the step size is increased from 20% to 60% of the original demand, the mean of the lead time error histogram decreases from -0.38 shift to -0.82 shift.

For the 60% step size, the most heavily loaded machine had a utilisation of 96% averaged over the whole simulation period.

The dynamic disturbances caused by such a violent step increase in demand must be considerable.

Yet the adaptive queueing time method was able not only to remain stable, but also to predict lead times quite accurately, as indicated by the mean obtained.

For any of the sinusoidal demand simulated, the mean obtained is very close to zero.

At the peak of the demand, the demanded utilisation of the most heavily utilised machine was about 96%; and at the trough, about 24%.

Thus, despite the sustained large disturbances, the method again remained stable, and produced very accurate lead times.

TABLE 8-1. The adaptive queueing time method with fixed and adaptive response rates.
Shift length = 8.00 hours.

Demand	Step increase (%) or Peak-to-trough amplitude(%)	Period (shifts)	Fixed Response Rate $\alpha = 0.10$		Adaptive Response Rate $\gamma = 0.05$		Adaptive Response Rate $\gamma = 0.10$	
			Lead Time Error Histogram		Lead Time Error Histogram		Lead Time Error Histogram	
			Mean (shifts)	Standard deviation (shifts)	Mean (shifts)	Standard deviation (shifts)	Mean (shifts)	Standard deviation (shifts)
Constant			- 0.27	2.23	0.17	2.23	0.31	2.47
Step	20		- 0.38	2.85	0.10	3.00	0.40	3.12
	40		- 0.51	3.13	0.18	3.49	0.47	3.53
	60		- 0.82	3.59	0.70	3.85	1.26	4.24
Sinusoidal	120	20	- 0.20	2.20	0.11	2.49	0.38	2.90
		40	- 0.23	2,81	0.16	2.75	0.22	3.41
		120	- 0.35	3.71	-0.28	3.93	-0.15	4.07
		240	- 0.17	3.68	0.02	3.68	0.30	3.35

Hence the adaptive queueing time method with fixed response rate can be used for a dynamic real life job shop.

With reference to the results for the adaptive queueing time method with adaptive response rate: $\gamma = 0.05$ in Table 8-1:- The means produced by this version are higher than the corresponding ones produced by the version with fixed response rate.

The means are also closer to zero. Thus, the lead times produced are more accurate.

As the step size is increased from 20% to 60% of the original demand, the mean increases from 0.10 to 0.70 shift.

From a manufacturer' point of view, these increases in the mean with increasing step disturbance are preferable to the reverse, as it is with the version with fixed response rate.

For the sinusoidal disturbances, all the means (except one) are slightly positive.

For the constant demand or steady-state case, the mean is also slightly positive.

The standard deviations produced by this version are, in general, slightly higher than the corresponding ones produced by the version with fixed response rate.

However, this version is preferred to the version with fixed response rate, because of the better accuracy of the lead times predicted, and the preference of a slight earliness to a slight lateness.

With reference to the results for the adaptive queueing time method with adaptive response rate: $\gamma = 0.10$, in Table 8-1:-

The means and the standard deviations are higher than the corresponding ones for $\gamma = 0.05$.

Thus, should higher means be preferred, γ could be increased to achieve it, at the expense of increasing the standard deviations.

CHAPTER 9

CONCLUSIONS AND RECOMMENDATIONS

9.1 CONCLUSIONS

The conclusions reached in this research may be summarised briefly as the following:

1. Testing under similar steady-state conditions shows that: of the four lead time prediction methods reported by Davies and McCallion *et al.*, the adaptive queueing time method gives the best accuracy.

2. Its system response rate is at least as fast as that of the queueing theory with forward load method, and is faster than that of the adaptive lead time method.

3. Testing under dynamic conditions, created by varying the pattern of demand, shows that it is able to remain stable and to give very accurate lead time estimates, even when subjected to very large and sustained disturbances.

4. By changing the response rate of the exponential smoothing, used in the adaptive queueing time method from fixed to adaptive, the system response rate is increased.

The mean of the lead time error histogram is also increased at the expense of increasing the standard deviation slightly. This is true for the steady-state conditions as well as the dynamic conditions.

5. For the adaptive queueing time method using adaptive response rate, the increase of the mean of the lead time error histogram (and the accompanying increase of the standard deviation), depends on the value chosen for γ : the larger the γ , the larger the increase of the mean.

6. By changing from fixed to adaptive response rate, the accuracy of the adaptive queueing time method is improved.

9.2 RECOMMENDATIONS

Based on the findings of this research, the author recommends that the next stage of the research be the implementation of the adaptive queueing time method with adaptive response rate in a real life machine shop, for the accurate prediction of lead times.

APPENDIX 1

Since the following informations for the parts used by Brittain was not available, it was generated randomly as below:

(a)(i) The Random Generation of the Total Number of Operations for a Part.

Graph 4-1 is the distribution of the total number of operations for a part, reproduced from Brittain's thesis.

The cumulative distribution of the distribution was plotted, as in Graph A1-1.

The right hand side of the distribution was assigned a linear scale of between 0 and 1.

A random number generator, which generated random numbers on the scale 0-1, was used to produce random numbers.

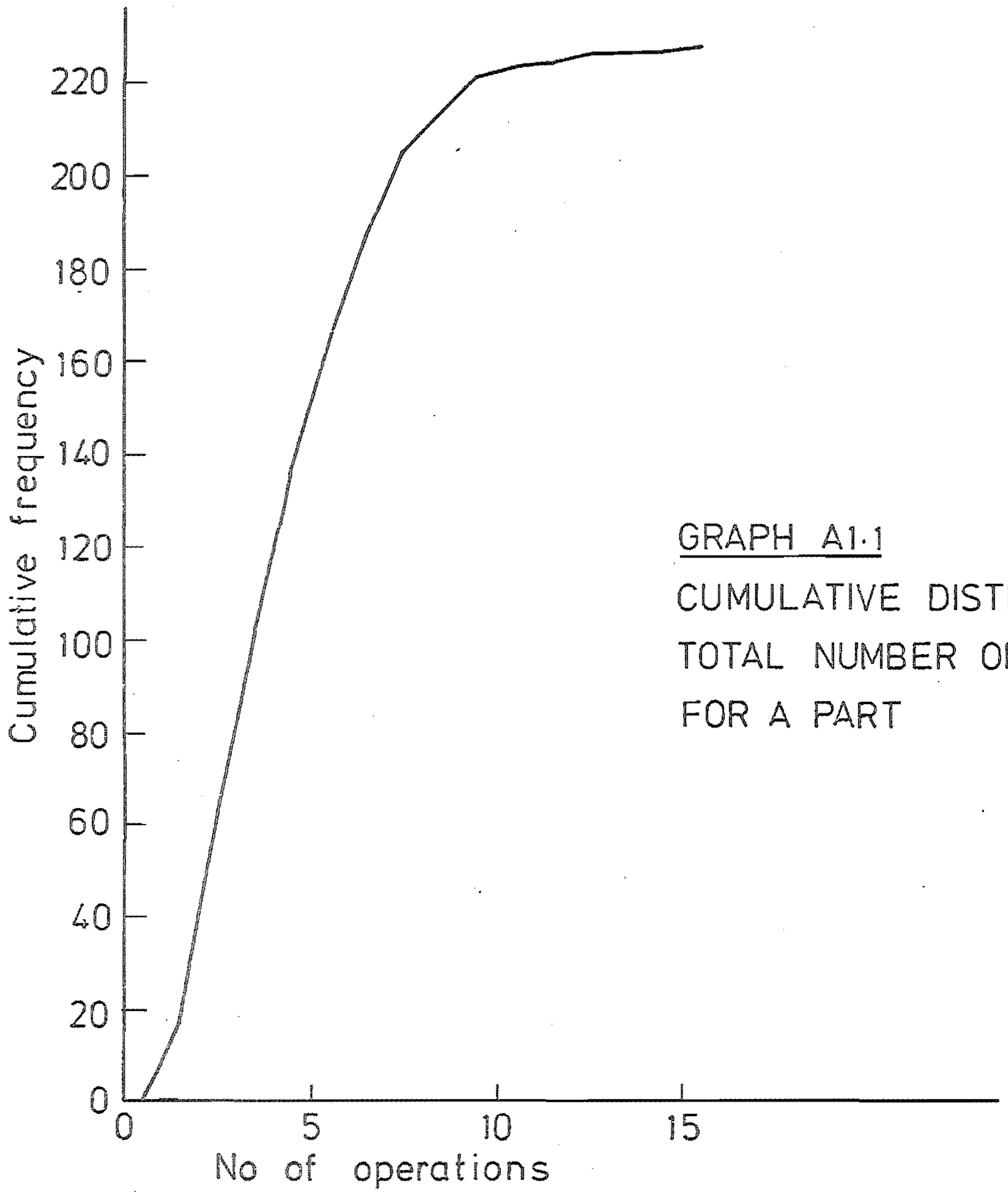
For each random number generated, a line was drawn horizontally to intersect with the distribution, to determine the total number of operations for a part.

(ii) The Random Generation of the Operation Time for Each operation.

Graph 4-2 is the distribution of the service time for an operation, reproduced from Brittain's thesis.

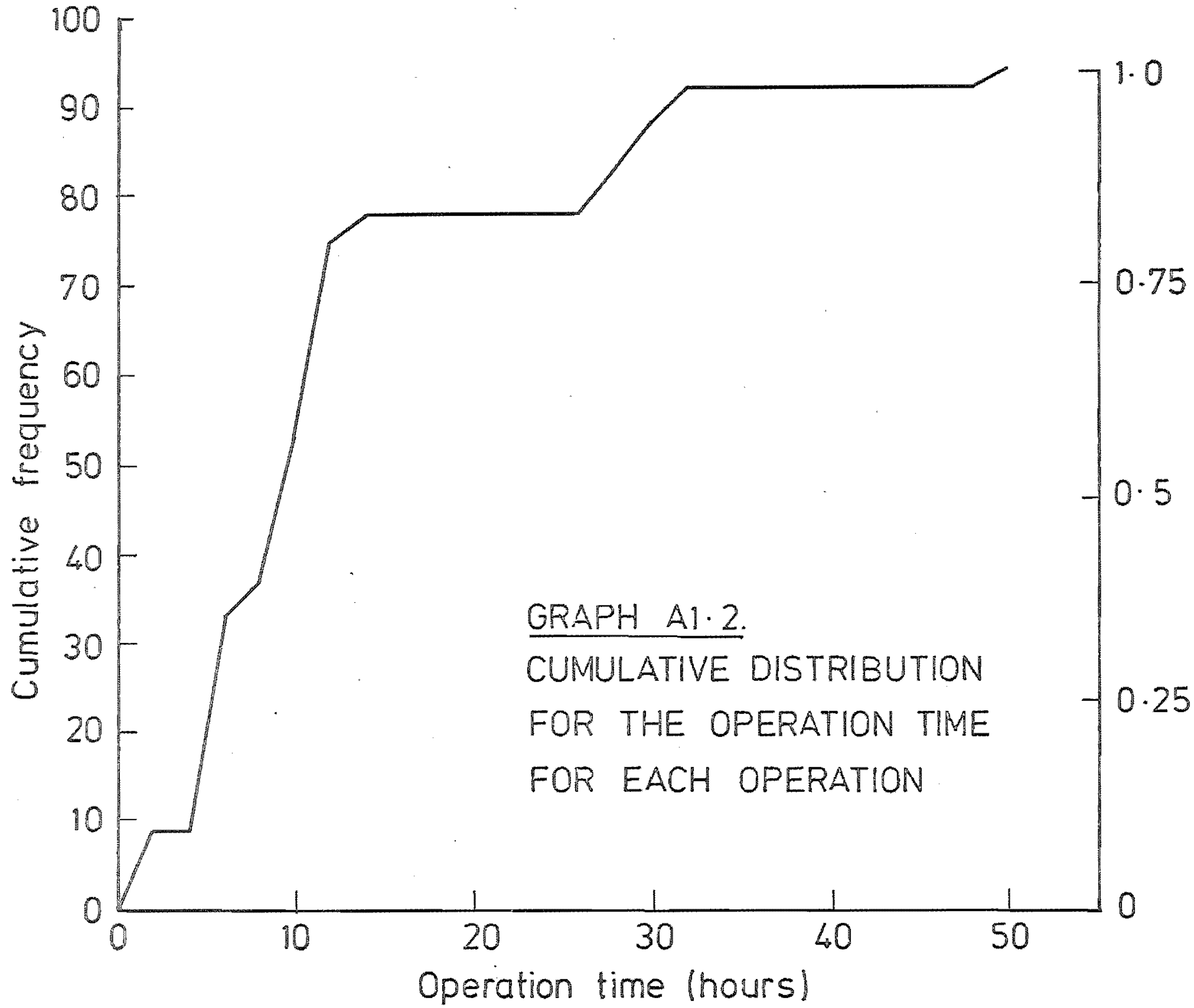
The cumulative distribution of the distribution was plotted as in Graph A1-2.

Then the operation time for each operation was



GRAPH A1.1

CUMULATIVE DISTRIBUTION FOR THE TOTAL NUMBER OF OPERATIONS FOR A PART



GRAPH A1·2.
CUMULATIVE DISTRIBUTION
FOR THE OPERATION TIME
FOR EACH OPERATION

determined from the cumulative histogram, in the same manner as the total number of operation for a part.

(b) The Selection of Parts For Data Sets 2 and 3.

To investigate the effect of job mix on the lead time error histogram, two sets of data, with characteristics different from that of Data Set 1, were generated.

Data Set 2 has a lower mean, as well as a lower standard deviation, of the lead time histogram, than Data Set 1.

Data Set 3 has a lower standard deviation, only, of the lead time histogram, than Data Set 1.

To select parts for Data Sets 2 and 3, the total operation times of the parts in Data Set 1 were averaged, and the value obtained used as a reference.

New parts were generated, and their total operation times were compared with the reference.

The new parts, whose total operation times were smaller than the reference, but were fairly close to each other, were selected for Data Set 2.

When 20 parts were selected, a simulation run was performed to see if the reductions in the mean and the standard deviation of the lead time histogram were achieved. If not, some parts were replaced by more suitable ones, until the simulation run showed that the reductions were achieved.

The new parts, whose total operation times were slightly larger or slightly smaller than the reference, were selected for Data Set 3.

Similarly, when 20 parts were selected, simulation runs were performed and parts replaced by more suitable ones, until the reduction in the standard deviation, only, of the lead time histogram was achieved.

APPENDIX 2

KEY TO THE SIMULATION PROGRAM

ALPHA	Exponential smoothing constant
AVEQT ( )	Real array containing exponentially smoothed queueing times
AVEUTL ( )	Real array containing average utilisations of the machine groups (%)
BDATA ( )	Integer array containing batch details BDATA(1) contains number of works currently used in array. The remainder is divided into eight word blocks: (i) Batch number (ii) Location in PARTT ( ) (iii) (Not used) (iv) Remaining machining time of current operation (v) Initial float (vi) Time elapsed since batch was placed into the machine shop (vii) Queueing time of batch at the current machine group (viii) Lead time to which batch was originally loaded.
BDL	Length of BDATA ( )
BNUM	Number of batches completed so far
BQ	Batch quantities
FIN	Upper limit of the LQE ( )
FLOAT	Float on a batch
FLT	Float on a batch
FPO	Float per operation

FWQ ( )	Integer array containing waiting times at the machine groups as predicted by the forward load program
FWU ( )	Integer array containing utilisations of the machine groups as predicted by the forward load program
HR	Clock counter, in hundred hour steps.
L	Location in BDATA ( )
LCN	Location used in LQE ( )
LF	Load factor = 0 a change of state = 1 no change of state
LQE ( )	Integer array containing histograms of lead time, queueing time and of their errors
LQEINT	Time interval used in LQE ( )
LT	Lead time
LTE	Lead time error
M	Location in PARTT ( )
MAXQ	Maximum number allowed in any queue
MACHS ( )	Integer array containing six items of information for each machine: (i) Machine set number (ii) Total machining time done by the machine (iii) = 0 if no job is on the machine (iv) Operation number of current job (v) Float of current job (vi) Machining time left on current job since the start of the current operation
MCS	Counter
MCSET	Counter
MIN	Time increment in the machine shop

MPRO	Maximum period of forward loading
MTIME	Machining time done on each group
MN	Mean of the lead time histogram, queueing time histogram, or their error histograms
N	Location of machine set in PARTT ( )
NO	Number of parts used in each simulation
OPNO	The number of an operation
OPS	Total number of operations of a part
OPT	Operation time
PARTT ( )	Integer array containing the information about the parts, divided into blocks of 35 words:

	(i)	Part number
	(ii)	Total number of operations
	(iii)	Batch quantities
	(iv)	Raw batch value
	(v)	Finished part value
10 sets of 3	(vi)	Machine group number for 1st operation
	(vii)	Operation time per piece
	(viii)	Cost of machining time on the machine group

PRD	Counter
QSET ( )	Queue location for each of the machine groups. Each location contains three items of informations:

(i)	Location in BDATA ( )
(ii)	The number of current operation
(iii)	Current float

QT	Queueing time
QTE	Queueing time error
QUEUE	The length of the queue at a machine group
RULE	Rule number

SCF ( )	Stock control file - eight items of information for 20 parts (i) Location in PARTT ( ) (ii) Shortage cost (iii) Current stock level (iv) Shortages (v) Usage or demand (vi) Lead time (vii) Finished batch value (viii) Quantity in work-in-progress
SETNO	Total number of machine groups
SFWQ ( )	Integer array containing smoothed waiting times obtained from the forward load predictor
SH	Shortages
SL	Stock level
ST	Lower limit of LQE ( )
STD	Standard deviation of the lead time histogram, queueing time histogram, or their error histograms
STSET ( )	Integer array containing information about each machine group: (i) (not used) (ii) Total number of machines in the group (iii) Load factor = 1 all machines occupied = 0 not all machines occupied (iv) (not used) (v) Queue length
TOTMCS	Total number of machines
USE	Demand or usage of stock
WIP	Quantity in work in progress

APPENDIX 3

LISTING OF THE SIMULATION PROGRAM

```
C       MACHINE SHOP SIMULATION
C       ************************************************************
C       *  APPROACH USING  ADAPTIVE WAITING TIME                   *
C       *  IQT=ALPHA*IQT+(1.-ALPHA)*AVEQT(MCS)                     *
C       ************************************************************


        DIMENSION AVEQT(10),AVEUTL(10),BDATA(4001),FWO(10),FWU(10)
       1            ,HUMACH(10),LOAD(10,100),LQE(51,22),MACHS(10,6),MTIME(10)
       2            ,PARTT(1750),QSET(10,100,3),RANK(100),SCF(50,8),SFWQ(10)
       3            ,STDQ(10),STSET(10,5),UTIL(10)
        COMMON /C1/ALPHA,AVEQT,AVEUTL,BDATA,BETA,CAP,FLOAT,FWQ,FWU,HR,LOAD
       1      ,LQEINT,LT,LTSF,MACHS,MIN,MN,MTIME,OPT,PARTT,PRDINT,QSET
       2      ,RLOAD,SCF,SFWQ,STD,STDQ,STSET,WIPT,WORK
        COMMON /C2/BDL,BQ,FIN,FSHIFT,HUMACH,IP,L,LCN,LF,LQE,LTS,M,MAXQ,MCS
       5      ,MCSET,MPRD,N,NO,NXTBNO,OPNO,OPS,PRD,QUEUE,RANK,RULE,SETNO
       6      ,SL,SH,SHIFT,SHL,ST,TOTMCS,USE,WIP,WK,WKL
        INTEGER     AVEQT,BDATA,CAP,FLOAT,FWQ,FWU,HR,LOAD,LQEINT,LT
       1            ,MACHS,MIN,MTIME,OPT,OPTTIM,PRDINT,PARTT,QSET,RLOAD
       2            ,SCF,SFWQ,STDQ,STSET,WIPT,WORK
       3            ,OPTIME
       4            ,BDL,BQ,FIN,FSHIFT,HUMACH,IP,L,LCN,LF,LQE,LTS,M,MAXQ,MCS
       1            ,MCSET,MPRD,N,NO,NXTBNO,OPNO,OPS,PRD,QUEUE,RANK,RULE
       2            ,SETNO,SL,SH,SHIFT,SHL,ST,TOTMCS,USE,WIP,WK,WKL
       3            ,BCHADD,LOCATE
        REAL LTSF,MN,STD,MNHRS,STDHRS
        READ(5,5) RULE,SETNO,TOTMCS,NO,BDL,MAXQ,FSHIFT,NXTBNO

        READ(5,5) LQEINT,PRDINT,MPRD,ST,FIN,SHL
        READ(5,5) ALPHA,LTSF
        READ(5,5) (AVEQT(I),I=1,SETNO)
        READ(5,5) (SFWQ(I),I=1,SETNO)
        READ(5,5) (AVEUTL(I),I=1,SETNO)
      5 FORMAT(10G8.2)
        WRITE(6,9) (AVEUTL(I),I=1,SETNO)

      9 FORMAT(/5X,'AVEUTL=',10G10.2)
        WRITE(6,20) (AVEQT(I),I=1,SETNO)
     20 FORMAT(/5X,'AVEQT=',10G10)
        WRITE(6,10) RULE,SETNO,TOTMCS,NO,BDL,MAXQ,FSHIFT,NXTBNO
     10 FORMAT(/5X,'RULE=',G4,2X,'SETNO=',G4,2X,'TOTMCS=',G4,2X,'NO=',G4,
       1            2X,'BDL= ',G4,2X,'MAXQ= ',G4,2X,'FSHIFT=',G4,
       2            2X,'NXTBNO=',G4)
        WRITE(6,11) LQEINT,PRDINT,MPRD,FIN,ST,SHL,ALPHA,LTSF
     11 FORMAT(/5X,'LQEINT=',G5,5X,'PRDINT=',G5,5X,'MPRD=',G5,5X,
       1            'FIN=     ',G5,5X,'ST=     ',G5,5X,'SHL= ',G5,
       2         /5X,'ALPHA=',G7.2,5X,'LTSF=',G7.2)
C        TO READ IN INFORMATIONS ABOUT JOBS -- SET UP PARTT ARRAY
        CALL INPUT
C        READ IN THE INITIAL STATES OF THE MACHINES AND INFORMATIONS ABOUT
C        THE BATCHES
        WKL=5*SHL
        BDATA(1)=BDL
        DO 12 I=2,BDL
     12 BDATA(I)=0
        DO 13 I=1,SETNO
        DO 13 J=1,MAXQ
        DO 13 K=1,3
     13 QSET(I,J,K)=0
C        SET UP STSET,MACHS,HUMACH ARRAYS,AND SCF,BDATA,QSET ARRAYS
        CALL RELOAD
        DO 23 IJ=1,TOTMCS
     23 MTIME(IJ)=0
        JS=SETNO*2+2
        DO 30 I=1,51
        DO 30 J=1,JS
     30 LQE(I,J)=0
C
        DO 200 IP=1,FSHIFT
        WRITE(6,47) IP
     47 FORMAT(/1X,'START OF SHIFT',I5,'**********************************')
        HR=0
C        TO LOAD BATCHES INTO THE MACHINE SHOP -- SET UP BDATA ARRAY
     48 CALL BLOAD
C
C        TO LOAD JOBS ONTO THE MACHINES FROM QUEUES,USING A PRIORITY RULE
     50 MIN=SHL-HR
```

```
      IF(MIN.LE.0) GO TO 130
      CALL JOBLD
C
C      TO DET NEXT EVENT
   60 DO 80 MCS=1,TOTMCS
      IF(MACHS(MCS,3).EQ.0) GO TO 80
      IF(MACHS(MCS,6).GT.MIN) GO TO 80
      MIN=MACHS(MCS,6)
   80 CONTINUE
      HR=HR+MIN
C
C      TO ADJUST PARAMETERS OF BATCHES IN QUEUES
      DO 100 I=1,SETNO
      QUEUE=STSET(I,5)
      IF(QUEUE.EQ.0) GO TO 100
      DO 90 J=1,QUEUE
      QSET(I,J,3)=QSET(I,J,3)-MIN
      L=QSET(I,J,1)
      BDATA(L+5)=BDATA(L+5)+MIN
      BDATA(L+6)=BDATA(L+6)+MIN
   90 CONTINUE
  100 CONTINUE
C
C      TO ADJUST PARAMETERS OF BATCHES ON MACHINES
      DO 125 MCS=1,TOTMCS
      L=MACHS(MCS,3)
      OPNO=MACHS(MCS,4)
      IF(L.EQ.0) GO TO 125
  110 MACHS(MCS,6)=MACHS(MCS,6)-MIN
      MACHS(MCS,2)=MACHS(MCS,2)+MIN
      MTIME(MCS)=MTIME(MCS)+MIN
      BDATA(L+3)=BDATA(L+3)-MIN
      BDATA(L+5)=BDATA(L+5)+MIN
C
C      TO ADJUST PARAMETERS IF AN OPERATION IS COMPLETED
      IF(MACHS(MCS,6).NE.0)GO TO 125
      FLOAT=MACHS(MCS,5)
      DO 120 J=3,6
      MACHS(MCS,J)=0
  120 CONTINUE
      MCSET=MACHS(MCS,1)
      STSET(MCSET,3)=0
      OPNO=OPNO+1
      CALL FORMQ
  125 CONTINUE
      GO TO 50
  130 CONTINUE
C
C      TO REDUCE STOCK ACCORDING TO USAGE.
      DO 143 I=1,NO
      SL=SCF(I,3)
      SH=SCF(I,4)
      USE=SCF(I,5)
      IF(SL.LT.USE) GO TO 141
      SL=SL-USE
      GO TO 142
  141 SH=SH+USE-SL
      SL=0
  142 SCF(I,3)=SL
      SCF(I,4)=SH
  143 CONTINUE
C
C      PRINT OUT THE WEEKLY UTILIZATION AND THE AVERAGE WAITING TIME
      IF (IP.NE.5*(IP/5)) GO TO 200
      WRITE(6,210) (AVEQT(I),I=1,SETNO)
  210 FORMAT(/5X,'AVEQT=',10G10)
      SUM=0.
      DO 153 I=1,SETNO
      AVEUTL(I)=100.*MTIME(I)/(5*SHL)
      MTIME(I)=0
      SUM=SUM+AVEUTL(I)
  153 CONTINUE
      WRITE(6,154) (AVEUTL(I),I=1,SETNO)
  154 FORMAT(/5X,'AVEUTL=',10G10.4)
      PCUTIL=SUM/SETNO
      WRITE(6,155) PCUTIL
  155 FORMAT(/5X,'AVERAGE UTILIZATION OF MACHINE SHOP=',G10.4)
```

```
C
C
  200 CONTINUE
C
C        TO PRINT OUT THE STATE OF THE MACHINE SHOP
      SUM=0.
      DO 160 I=1,TOTMCS
      UTIL(I)=100.*MACHS(I,2)/(FSHIFT*SHL)
  160 SUM=SUM+MACHS(I,2)
      WRITE(6,162) (UTIL(I),I=1,TOTMCS)
  162 FORMAT(/5X,'AVE UTIL OF MACHINE OVER THE SIMN PERIOD=',/5X,10G9.2)
      PCUTIL=SUM/(TOTMCS*FSHIFT*SHL)*100.
      WRITE(6,165) PCUTIL
  165 FORMAT(/5X,'THE AVE UTIL OF THE MACHINE SHOP OVER THE SIMULTION
     1        PERIOD = ',G10.4,'****************************************')
      CALL OUTPUT
      DO 150 I=1,SETNO
      LCN=I*2-1
      CALL STOCAL(LQE,LQEINT,O,FIN-ST,LCN,MN,STD,MNHRS,STDHRS,NH)
      CALL OPHIST(LQE,O,FIN-ST,LCN,MN,STD,MNHRS,STDHRS,NH)
      LCN=I*2
      CALL STOCAL(LQE,LQEINT,ST,FIN,LCN,MN,STD,MNHRS,STDHRS,NH)
      CALL OPHIST(LQE,ST,FIN,LCN,MN,STD,MNHRS,STDHRS,NH)
  150 CONTINUE
      LCN=SETNO*2+1
      CALL STOCAL(LQE,LQEINT,O,FIN-ST,LCN,MN,STD,MNHRS,STDHRS,NH)
      CALL OPHIST(LQE,O,FIN-ST,LCN,MN,STD,MNHRS,STDHRS,NH)
      LCN=SETNO*2+2
      CALL STOCAL(LQE,LQEINT,ST,FIN,LCN,MN,STD,MNHRS,STDHRS,NH)
      CALL OPHIST(LQE,ST,FIN,LCN,MN,STD,MNHRS,STDHRS,NH)
      STOP
      END
```

```
      SUBROUTINE BLOAD
      DIMENSION AVEQT(10),AVEUTL(10),BDATA(4001),FWQ(10),FWU(10)
     1          ,HUMACH(10),LOAD(10,100),LQE(51,22),MACHS(10,6),MTIME(10)
     2          ,PARTT(1750),QSET(10,100,3),RANK(100),SCF(50,8),SFWQ(10)
     3          ,STDQ(10),STSET(10,5)
      COMMON /C1/ALPHA,AVEQT,AVEUTL,BDATA,BETA,CAP,FLOAT,FWQ,FWU,HR,LOAD
     1        ,LQEINT,LT,LTSF,MACHS,MIN,MN,MTIME,OPT,PARTT,PRDINT,QSET
     2        ,RLOAD,SCF,SFWQ,STD,STDQ,STSET,WIPT,WORK
      COMMON /C2/BQL,BQ,FIN,FSHIFT,HUMACH,IP,L,LCN,LF,LQE,LTS,M,MAXQ,MCS
     5        ,MCSET,MPRD,N,NQ,NXTBNO,OPNO,OPS,PRD,QUEUE,RANK,RULE,SETNO
     6        ,SL,SH,SHIFT,SHL,ST,TOTMCS,USE,WIP,WK,WKL
      INTEGER   AVEQT,BDATA,CAP,FLOAT,FWQ,FWU,HR,LOAD,LQEINT,LT
     1          ,MACHS,MIN,MTIME,OPT,OPTTIM,PRDINT,PARTT,QSET,RLOAD
     2          ,SCF,SFWQ,STDQ,STSET,WIPT,WORK
     3          ,OPTIME
     4          ,BQL,BQ,FIN,FSHIFT,HUMACH,IP,L,LCN,LF,LQE,LTS,M,MAXQ,MCS
     1          ,MCSET,MPRD,N,NQ,NXTBNO,OPNO,OPS,PRD,QUEUE,RANK,RULE
     2          ,SETNO,SL,SH,SHIFT,SHL,ST,TOTMCS,USE,WIP,WK,WKL
     3          ,BCHADD,LOCATE
      REAL LTSF,MN,STD
C
      DO 40 J=1,NG
      M=SCF(J,1)
      SL=SCF(J,3)
      SH=SCF(J,4)
      USE=SCF(J,5)
      WIP=SCF(J,8)
C
      OPS=PARTT(M+1)
      ISUM=0
      DO 5 I=1,OPS
      N=M+3*I+2
      MCS=PARTT(N)
      ISUM=ISUM+AVEQT(MCS)+OPTIME(M,N,PARTT)
    5 CONTINUE
      LT=ISUM
      LTS=(ISUM+0.999*SHL)/SHL
      SCF(J,6)=LT
      LTU=LTS*USE
    8 IF(SL+WIP-SH.GE.LTU) GO TO 40
      L=BCHADD(BDATA)
      BDATA(L)=NXTBNO
      NXTBNO=NXTBNO+1
      BDATA(L+1)=M
      BDATA(L+2)=PARTT(M+3)
      N=M+5
      BDATA(L+3)=OPTIME(M,N,PARTT)
C
      MCS=PARTT(N)
      FLOAT=0
      IOPT=0
      DO 25 I=1,OPS
      N=M+3*I+2
      MCS=PARTT(N)
   15 OPT=OPTIME(M,N,PARTT)
      IOPT=IOPT+OPT
      FLOAT=FLOAT+AVEQT(MCS)
   25 CONTINUE
      BDATA(L+4)=FLOAT
      BDATA(L+5)=0
      BDATA(L+6)=0
      LT=IOPT+FLOAT
      BDATA(L+7)=LT
C
C      UPDATE WORK IN PROGRESS.
      BQ=PARTT(M+2)
      WIP=WIP+BQ
      SCF(J,8)=WIP
C
C      PLACE THE BATCH INTO THE APPROPRIATE QUEUE.
      OPNO=1
      CALL FORMQ
      GO TO 8
   40 CONTINUE
C
      WRITE(6,45) NXTBNO


   45 FORMAT(/5X,'LOADING COMPLETED,NXTBNO=',G5)
      RETURN
      END
```

```
      SUBROUTINE FORMQ
      DIMENSION AVEQT(10),AVEUTL(10),BDATA(4001),FWQ(10),FWU(10)
     1          ,HUMACH(10),LOAD(10,100),LQE(51,22),MACHS(10,6),MTIME(10)
     2          ,PARTT(1750),QSET(10,100,3),RANK(100),SCF(50,8),SFWQ(10)
     3          ,STOQ(10),STSET(10,5)
      COMMON /C1/ALPHA,AVEQT,AVEUTL,BDATA,BETA,CAP,FLOAT,FWQ,FWU,HR,LOAD
     1       ,LQEINT,LT,LTSF,MACHS,MIN,MN,MTIME,OPT,PARTT,PROINT,QSET
     2       ,RLOAD,SCF,SFWQ,STO,STOQ,STSET,WIPT,WORK
      COMMON /C2/BOL,BQ,FIN,FSHIFT,HUMACH,IP,L,LCN,LF,LQE,LTS,M,MAXQ,MCS
     5       ,MCSET,MPRO,N,NO,NXTBNO,OPNO,OPS,PRO,QUEUE,RANK,RULE,SETNO
     6       ,SL,SH,SHIFT,SHL,ST,TOTMCS,USE,WIP,WK,WKL
      INTEGER   AVEQT,BDATA,CAP,FLOAT,FWQ,FWU,HR,LOAD,LQEINT,LT
     1          ,MACHS,MIN,MTIME,OPT,OPTTIM,PROINT,PARTT,QSET,RLOAD
     2          ,SCF,SFWQ,STOQ,STSET,WIPT,WORK
     3          ,OPTIME
     4          ,BOL,BQ,FIN,FSHIFT,HUMACH,IP,L,LCN,LF,LQE,LTS,M,MAXQ,MCS
     1          ,MCSET,MPRO,N,NO,NXTBNO,OPNO,OPS,PRO,QUEUE,RANK,RULE
     2          ,SETNO,SL,SH,SHIFT,SHL,ST,TOTMCS,USE,WIP,WK,WKL
     3          ,BCHADD,LOCATE
      REAL LTSF,MN,STO
      M =BDATA(L+1)
      OPS=PARTT(M+1)
      IF(OPNO.LE.OPS) GO TO 10
C
C       ENTER THIS BRANCH OF THE PROGRAM ONLY IF THE BATCHES HAVE ALL
C       THEIR OPERATIONS FINISHED.
      RHR=0.01*HR
      LT=BDATA(L+5)
      LTE=BDATA(L+7)-BDATA(L+5)
      BNUM=BDATA(L)
      BQ=PARTT(M+2)
      CALL STORES
      LCN=2*SETNO+1
      CALL STRLQE(LT,0,FIN-ST,LCN,LQEINT,LQE)
      LCN=2*SETNO+2
      CALL STRLQE(LTE,ST,FIN,LCN,LQEINT,LQE)
      DO 150 IJJ=1,8
      IJK=L+IJJ-1
      BDATA(IJK)=0
  150 CONTINUE
      RETURN
C
C       PLACE THE BATCH AT THE MACHINE SET OF ITS NEXT OPERATION.
   10 N=M+3*OPNO+2
      MCSET=PARTT(N)
      QUEUE=STSET(MCSET,5)
      LF=STSET(MCSET,3)
      IF(MACHS(MCSET,3).EQ.0.AND.QUEUE.EQ.0) LF=0
      QUEUE=QUEUE+1
      IF(QUEUE.GT.MAXQ) GO TO 20
      STSET(MCSET,3)=LF
      OPT=OPTIME(M,N,PARTT)
      STSET(MCSET,5)=QUEUE
      QSET(MCSET,QUEUE,1)=L
      QSET(MCSET,QUEUE,2)=OPNO
      QSET(MCSET,QUEUE,3)=FLOAT
      BDATA(L+6)=0
      BDATA(L+3)=OPT
   40 CONTINUE
      RETURN
C
   20 WRITE(6,25) MAXQ,MCSET
   25 FORMAT(//1X,'QSET OVERFLOW,MAXQ=',G5,5X,'MCSET=',G5)
      STOP
      END
```

```
      SUBROUTINE INPUT
      DIMENSION AVEQT(10),AVEUTL(10),BDATA(4001),FWQ(10),FWU(10)
     1          ,HUMACH(10),LOAD(10,100),LQE(51,22),MACHS(10,6),MTIME(10)
     2          ,PARTT(1750),QSET(10,100,3),RANK(100),SCF(50,8),SFWQ(10)
     3          ,STDQ(10),STSET(10,5)
      COMMON /C1/ALPHA,AVEQT,AVEUTL,BDATA,BETA,CAP,FLOAT,FWQ,FWU,HR,LOAD
     1       ,LQEINT,LT,LTSF,MACHS,MIN,MN,MTIME,OPT,PARTT,PROINT,QSET
     2       ,RLOAD,SCF,SFWQ,STD,STDQ,STSET,WIPT,WORK
      COMMON /C2/BOL,BQ,FIN,FSHIFT,HUMACH,IP,L,LCN,LF,LQE,LTS,M,MAXQ,MCS
     5       ,MCSET,MPRD,N,NO,NXTBNO,OPNO,OPS,PRD,QUEUE,RANK,RULE,SETNO
     6       ,SL,SH,SHIFT,SHL,ST,TOTMCS,USE,WIP,WK,WKL
      INTEGER   AVEQT,BDATA,CAP,FLOAT,FWQ,FWU,HR,LOAD,LQEINT,LT
     1          ,MACHS,MIN,MTIME,OPT,OPTTIM,PROINT,PARTT,QSET,RLOAD
     2          ,SCF,SFWQ,STDQ,STSET,WIPT,WORK
     3          ,OPTIME
     4          ,BOL,BQ,FIN,FSHIFT,HUMACH,IP,L,LCN,LF,LQE,LTS,M,MAXQ,MCS
     1          ,MCSET,MPRD,N,NO,NXTBNO,OPNO,OPS,PRO,QUEUE,RANK,RULE
     2          ,SETNO,SL,SH,SHIFT,SHL,ST,TOTMCS,USE,WIP,WK,WKL
     3          ,BCHADD,LOCATE
      REAL LTSF,MN,STD
      WRITE(6,9)
    9 FORMAT(/5X,'PARTT ARRAY IS')
      DO 20 J=1,NO
      IL=(J-1)*35+1
      IH=IL+34
      READ(5,2) (PARTT(I),I=IL,IH)
    2 FORMAT(20I4)
      WRITE(6,10) (PARTT(I),I=IL,IH)
   10 FORMAT(//1X,5I4,/10X,30I4)
   20 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE JOBLD
      DIMENSION AVEQT(10),AVEUTL(10),BDATA(4001),FWQ(10),FWU(10)
     1          ,HUMACH(10),LOAD(10,100),LQE(51,22),MACHS(10,6),MTIME(10)
     2          ,PARTT(1750),QSET(10,100,3),RANK(100),SCF(50,8),SFWQ(10)
     3          ,STDQ(10),STSET(10,5)
      COMMON /C1/ALPHA,AVEQT,AVEUTL,BDATA,BETA,CAP,FLOAT,FWQ,FWU,HR,LOAD
     1      ,LQEINT,LT,LTSF,MACHS,MIN,MN,MTIME,OPT,PARTT,PROINT,QSET
     2      ,RLOAD,SCF,SFWQ,STD,STDQ,STSET,WIPT,WORK
      COMMON /C2/BOL,BQ,FIN,FSHIFT,HUMACH,IP,L,LCN,LF,LQE,LTS,M,MAXQ,MCS
     5      ,MCSET,MPRD,N,NO,NXTBNO,OPNO,OPS,PRD,QUEUE,RANK,RULE,SETNO
     6      ,SL,SH,SHIFT,SHL,ST,TOTMCS,USE,WIP,WK,WKL
      INTEGER   AVEQT,BDATA,CAP,FLOAT,FWQ,FWU,HR,LOAD,LQEINT,LT
     1          ,MACHS,MIN,MTIME,OPT,OPTTIM,PROINT,PARTT,QSET,RLOAD
     2          ,SCF,SFWQ,STDQ,STSET,WIPT,WORK
     3          ,OPTIME
     4          ,BOL,BQ,FIN,FSHIFT,HUMACH,IP,L,LCN,LF,LQE,LTS,M,MAXQ,MCS
     1          ,MCSET,MPRD,N,NO,NXTBNO,OPNO,OPS,PRD,QUEUE,RANK,RULE
     2          ,SETNO,SL,SH,SHIFT,SHL,ST,TOTMCS,USE,WIP,WK,WKL
     3          ,BCHADD,LOCATE
      REAL LTSF,MN,STD
      DO 50 MCS=1,TOTMCS
      MCSET=MACHS(MCS,1)
      QUEUE=STSET(MCSET,5)
      IF(QUEUE.EQ.0) GO TO 50
      LF=STSET(MCSET,3)
      IF(LF.EQ.1) GO TO 50
      L=MACHS(MCS,3)
      IF(L.GT.0.AND.LF.EQ.0) GO TO 60
      LF=1
      CALL PRIOR
      IQT=BDATA(L+6)
      LCN=MCSET*2-1
      CALL STRLQE(IQT,0,FIN-ST,LCN,LQEINT,LQE)
      IQTE=AVEQT(MCS)-BDATA(L+6)
      LCN=MCSET*2
      CALL STRLQE(IQTE,ST,FIN,LCN,LQEINT,LQE)
      AVEQT(MCS)=ALPHA*IQT+(1.-ALPHA)*AVEQT(MCS)
      BDATA(L+6)=0
      MACHS(MCS,3)=L
      MACHS(MCS,4)=OPNO
      MACHS(MCS,5)=FLOAT
      MACHS(MCS,6)=BDATA(L+3)
      QUEUE=QUEUE-1
      STSET(MCSET,3)=LF
      STSET(MCSET,5)=QUEUE
   50 CONTINUE
      RETURN
C
   60 WRITE(6,70)
   70 FORMAT(/5X,'FAULT IN MACHINE SET,LOCATED IN SUBROUTINE JOBLD')
      STOP
      END
```

```
      SUBROUTINE OPHIST(LOE,ST,FIN,LCN,MN,STD,MNHRS,STDHRS,NH)
      DIMENSION HIST(150),LOE(51,22)
      INTEGER   FIN,HIST,LCN,LOE,ST
      REAL MN,STD,MNHRS,STDHRS
C   HIST ARRAY IS USED TO STORE THE FREQUENCY DISTRIBUTION FOR PRINT OUTP UT.
C     K=1 IF THE NUMBER IN THE ARRAY DOES NOT EXCEED 120
C     K=2 OTHERWISE.
      JJ=FIN-ST+1
      DO 45 KK=1,JJ
      K=1
      I=LOE(KK,LCN)
      IF(I.LE.120) GO TO 10
      K=2
      I=120
   10 IK=KK+ST-1
      DO 20 IJ=1,I
      HIST(IJ)=K
   20 CONTINUE
      IF(I.LE.0) WRITE(6,30) IK,I
   30 FORMAT(2I4)
      IF(I.GT.0) WRITE(6,40) IK,I,(HIST(IJ),IJ=1,I)
   40 FORMAT(2I4,2X,120I1)
   45 CONTINUE
      WRITE(6,50) MN,MNHRS,STD,STDHRS,NH
   50 FORMAT(/5X,'MEAN=',G10.3,G10.3,5X,'STD DEV=',G10.3,G10.3,
     1          'NO IN HIST=',G4)
      RETURN
      END
```

```
      SUBROUTINE OUTPUT
      DIMENSION AVEQT(10),AVEUTL(10),BDATA(4001),FWQ(10),FWU(10)
     1          ,HUMACH(10),LOAD(10,100),LQE(51,22),MACHS(10,6),MTIME(10)
     2          ,PARTT(1750),QSET(10,100,3),RANK(100),SCF(50,8),SFWQ(10)
     3          ,STDQ(10),STSET(10,5)
      COMMON /C1/ALPHA,AVEQT,AVEUTL,BDATA,BETA,CAP,FLOAT,FWQ,FWU,HR,LOAD
     1      ,LQEINT,LT,LTSF,MACHS,MIN,MN,MTIME,OPT,PARTT,PROINT,QSET
     2      ,RLOAD,SCF,SFWQ,STD,STDQ,STSET,WIPT,WORK
      COMMON /C2/BDL,BQ,FIN,FSHIFT,HUMACH,IP,L,LCN,LF,LQE,LTS,M,MAXQ,MCS
     5      ,MCSET,MPRD,N,NJ,NXTBNO,OPNO,OPS,PRD,QUEUE,RANK,RULE,SETNO
     6      ,SL,SH,SHIFT,SHL,ST,TOTMCS,USE,WIP,WK,WKL
      INTEGER   AVEQT,BDATA,CAP,FLOAT,FWQ,FWU,HR,LOAD,LQEINT,LT
     1          ,MACHS,MIN,MTIME,OPT,OPTTIM,PROINT,PARTT,QSET,RLOAD
     2          ,SCF,SFWQ,STDQ,STSET,WIPT,WORK
     3          ,OPTIME
     4          ,BDL,BQ,FIN,FSHIFT,HUMACH,IP,L,LCN,LF,LQE,LTS,M,MAXQ,MCS
     1          ,MCSET,MPRD,N,NJ,NXTBNO,OPNO,OPS,PRD,QUEUE,RANK,RULE
     2          ,SETNO,SL,SH,SHIFT,SHL,ST,TOTMCS,USE,WIP,WK,WKL
     3          ,BCHADD,LOCATE
      REAL LTSF,MN,STD
      RHR=HR*0.01
      WRITE(6,5) RHR
    5 FORMAT(1H0,'STATE OF MACHINE SHOP AT TIME ',F8.2,'HR*************')
      K=0
      DO 70 I=1,SETNO
      WRITE(6,10) I
   10 FORMAT(1H0,'MACHINE SET NO ',I4)
      QUEUE=STSET(I,5)
      WRITE(6,20) QUEUE
   20 FORMAT(/5X,'QUEUE LENGTH = ',I4)
      IF(QUEUE.EQ.0) GO TO 55
      DO 50 J=1,QUEUE
      AB=QSET(I,J,3)*0.01
      IY=QSET(I,J,1)
      WRITE(6,30) IY,BDATA(IY),QSET(I,J,2),AB
   30 FORMAT(/5X,'BDATA(',I3,')=',I8,/5X,'OPERATION NO = ',I4,
     1        /5X,'CURRENT FLOAT = ',F8.2)
   50 CONTINUE
C
   55 MCS=STSET(I,2)
      DO 60 J=1,MCS
      IY=K+J
      A=MACHS(IY,5)*0.01
      B=MACHS(IY,6)*0.01
      C=MACHS(IY,2)*0.01
      WRITE(6,25) C
   25 FORMAT(/5X,'MACHINING TIME DONE ON MACHINE = ',F8.2)
      L=MACHS(IY,3)
      IF(L.EQ.0) GO TO 60
      WRITE(6,40) IY,BDATA(L),MACHS(IY,4),A,B
   40 FORMAT(/5X,'CURRENT JOB ON MACHINE NO ',I4,5X,'IS',I4,
     1        /5X,'OPERATION NO = ',I8,
     2        /5X,'FLOAT = ',G8.2,'HOURS',
     3        /5X,'MACHINING TIME LEFT = ',G8.2,'HOURS')
   60 CONTINUE
      K=K+MCS
   70 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE PRIOR
      DIMENSION AVEQT(10),AVEUTL(10),BDATA(4001),FWQ(10),FWU(10)
     1          ,HUMACH(10),LOAD(10,100),LQE(51,22),MACHS(10,6),MTIME(10)
     2          ,PARTT(1750),QSET(10,100,3),RANK(100),SCF(50,8),SFWQ(10)
     3          ,STDQ(10),STSET(10,5)
      COMMON /C1/ALPHA,AVEQT,AVEUTL,BDATA,BETA,CAP,FLOAT,FWQ,FWU,HR,LOAD
     1       ,LQEINT,LT,LTSF,MACHS,MIN,MN,MTIME,OPT,PARTT,PRDINT,QSET
     2       ,RLOAD,SCF,SFWQ,STD,STDQ,STSET,WIPT,WORK
      COMMON /C2/BDL,BQ,FIN,FSHIFT,HUMACH,IP,L,LCN,LF,LQE,LTS,M,MAXQ,MCS
     5       ,MCSET,MPRD,N,NO,NXTBNO,OPNO,OPS,PRD,QUEUE,RANK,RULE,SETNO
     6       ,SL,SH,SHIFT,SHL,ST,TOTMCS,USE,WIP,WK,WKL
      INTEGER    AVEQT,BDATA,CAP,FLOAT,FWQ,FWU,HR,LOAD,LQEINT,LT
     1          ,MACHS,MIN,MTIME,OPT,OPTTIM,PRDINT,PARTT,QSET,RLOAD
     2          ,SCF,SFWQ,STDQ,STSET,WIPT,WORK
     3          ,OPTIME
     4          ,BDL,BQ,FIN,FSHIFT,HUMACH,IP,L,LCN,LF,LQE,LTS,M,MAXQ,MCS
     1          ,MCSET,MPRD,N,NO,NXTBNO,OPNO,OPS,PRD,QUEUE,RANK,RULE
     2          ,SETNO,SL,SH,SHIFT,SHL,ST,TOTMCS,USE,WIP,WK,WKL
     3          ,BCHADD,LOCATE
      REAL LTSF,MN,STD
      K=1
      IF(QUEUE.EQ.1) GO TO 15
C
      GO TO (1,3,5,7,9,11),RULE
C
C      MINIMUM FLOAT RULE  (=1)
    1 IMIN=1000000
      DO 2  I=1,QUEUE
      FLOAT=QSET(MCSET,I,3)
      IF(FLOAT.GE.IMIN) GO TO 2
      K=I
      IMIN=FLOAT
    2 CONTINUE
      GO TO 15
C
C      1 ST BATCH VALUE RULE  (=2)
    3 IMIN=0
      DO 4  I=1,QUEUE
      L=QSET(MCSET,I,1)
      M=BDATA(L+1)
      FPV=PARTT(M+4)
      BVALUE=FPV*BDATA(L+5)/BDATA(L+7)
      IF(BVALUE.LE.IMIN) GO TO 4
      K=I
      IMIN=BVALUE
    4 CONTINUE
      GO TO 15
C
C      2 ND BATCH VALUE RULE  (=3)
    5 IMIN=0
      DO 6  I=1,QUEUE
      L=QSET(MCSET,I,1)
      M=BDATA(L+1)
      FPV=PARTT(M+4)
      BVALUE=FPV*(BDATA(L+5)/BDATA(L+7))**2
      IF(BVALUE.LE.IMIN) GO TO 6
      K=I
      IMIN=BVALUE
    6 CONTINUE
      GO TO 15
C
C      SHORTEST IMMINENT OPERATION RULE  (=4)
    7 IMIN=1000000
      DO 8  I=1,QUEUE
      L=QSET(MCSET,I,1)
      IMMOPN=BDATA(L+3)
      IF(IMMOPN.GE.IMIN) GO TO 8
      K=I
      IMIN=IMMOPN
    8 CONTINUE
      GO TO 15
C
C      SHORTEST REMAINING MACHINING TIME RULE  (=5)
    9 IMIN=1000000
      DO 10 I=1,QUEUE
      L=QSET(MCSET,I,1)
```

```
      M=BDATA(L+1)
      OPNU=QSET(MCSET,I,2)
      OPS=PARTT(M+1)
      REMACH=0
      DO 100 J=OPNO,OPS
      N=M+3*J+2
  100 REMACH=REMACH+OPTIME(M,N,PARTT)
      IF(REMACH.GE.IMIN) GO TO 10
      K=I
      IMIN=REMACH
   10 CONTINUE
      GO TO 15
C
C      FIRST COME FIRST SERVED RULE  (=6)
   11 IMIN=0
      DO 12 I=1,QUEUE
      L=QSET(MCSET,I,1)
      FCOME=BDATA(L+6)
      IF(FCOME.LE.IMIN) GO TO 12
      K=I
      IMIN=FCOME
   12 CONTINUE
C
C      INFORMATIONS REGARDING THE BATCH SELECTED.
   15 L=QSET(MCSET,K,1)
      OPNU=QSET(MCSET,K,2)
      FLOAT=QSET(MCSET,K,3)
      IF(K.EQ.QUEUE) GO TO 25
      IF(QUEUE.EQ.1) GO TO 25
C
C      ADJUST QSET AFTER A BATCH IS SELECTED.
      J=QUEUE-1
      DO 20 I=K,J
      QSET(MCSET,I,1)=QSET(MCSET,I+1,1)
      QSET(MCSET,I,2)=QSET(MCSET,I+1,2)
   20 QSET(MCSET,I,3)=QSET(MCSET,I+1,3)
   25 QSET(MCSET,QUEUE,1)=0
      QSET(MCSET,QUEUE,2)=0
      QSET(MCSET,QUEUE,3)=0
   30 RETURN
      END
```

```
      SUBROUTINE RELOAD
      DIMENSION AVEQT(10),AVEUTL(10),BDATA(4001),FWQ(10),FWU(10)
     1          ,HUMACH(10),LOAD(10,100),LQE(51,22),MACHS(10,6),MTIME(10)
     2          ,PARTT(1750),QSET(10,10,3),RANK(100),SCF(50,8),SFWQ(10)
     3          ,STOQ(10),STSET(10,5)
      COMMON /C1/ALPHA,AVEQT,AVEUTL,BDATA,BETA,CAP,FLOAT,FWQ,FWU,HR,LOAD
     1      ,LQEINT,LT,LTSF,MACHS,MIN,MN,MTIME,OPT,PARTT,PRDINT,QSET
     2      ,RLOAD,SCF,SFWQ,STU,STOQ,STSET,WIPT,WORK
      COMMON /C2/BOL,BQ,FIN,FSHIFT,HUMACH,IP,L,LCN,LF,LQE,LTS,M,MAXQ,MCS
     5      ,MCSET,MPRD,N,NO,NXTBNO,OPNO,OPS,PRD,QUEUE,RANK,RULE,SETNO
     6      ,SL,SH,SHIFT,SHL,ST,TOTMCS,USE,WIP,WK,WKL
      INTEGER   AVEQT,BDATA,CAP,FLOAT,FWQ,FWU,HR,LOAD,LQEINT,LT
     1          ,MACHS,MIN,MTIME,OPT,OPTTIM,PRDINT,PARTT,QSET,RLOAD
     2          ,SCF,SFWQ,STOQ,STSET,WIPT,WORK
     3          ,OPTIME
     4          ,BOL,BQ,FIN,FSHIFT,HUMACH,IP,L,LCN,LF,LQE,LTS,M,MAXQ,MCS
     1          ,MCSET,MPRD,N,NO,NXTBNO,OPNO,OPS,PRD,QUEUE,RANK,RULE
     2          ,SETNO,SL,SH,SHIFT,SHL,ST,TOTMCS,USE,WIP,WK,WKL
     3          ,BCHAUD,LOCATE
      REAL LTSF,MN,STU
      READ(5,5) ((STSET(I,J),J=1,5),I=1,SETNO)
    5 FORMAT(10I8)
      WRITE(6,10) ((STSET(I,J),J=1,5),I=1,SETNO)
   10 FORMAT(/5X,'STSET=',(/5X,10I10))
      DO 15 I=1,TOTMCS
   15 READ(5,5) (MACHS(I,J),J=1,6)
      WRITE(6,25) ((MACHS(I,J),J=1,6),I=1,TOTMCS)
   25 FORMAT(/5X,'MACHS=',(/10X,12I8))
      READ(5,5) (HUMACH(I),I=1,SETNO)
      WRITE(6,30) (I,HUMACH(I),I=1,SETNO)
   30 FORMAT(/5X,'HUMACH ARRAY IS',10(G3,'==',G1))
      DO 35 I=1,NO
   35 READ(5,40) (SCF(I,J),J=1,8)
   40 FORMAT((8I8))
      WRITE(6,45) ((SCF(I,J),J=1,8),I=1,NO)
   45 FORMAT(/5X,'SCF ARRAY IS',(/2X,8G8))
      READ(5,40) (BDATA(I),I=1,BOL)
      WRITE(6,65) (BDATA(I),I=2,BOL)
   65 FORMAT(/5X,'BDATA=',(/1X,16I8))
      DO 75 I=1,SETNO
      READ(5,70) ((QSET(I,J,K),K=1,3),J=1,MAXQ)
   70 FORMAT((9G8))
   75 WRITE(6,80) ((QSET(I,J,K),K=1,3),J=1,MAXQ)
   80 FORMAT(/5X,'QSET=',(/2X,15G8))
      RETURN
      END
```

```
      SUBROUTINE STDCAL(LQE,LQEINT,ST,FIN,LCN,MN,STD,MNHRS,STDHRS,NH)
      DIMENSION LQE(51,22)
      INTEGER   FIN,LCN,LQE,ST
      REAL MN,STD,MNHRS,STDHRS,X,Y,Z
      X=0.
      Y=0.
      Z=0.
      NH=0
      K=FIN-ST+1
      DO 10 J=1,K
      I=LQE(J,LCN)
      X=X+I
      Y=Y+I*J
      NH=NH+I
   10 CONTINUE
      IF(X.NE.0) GO TO 20
      X=1
   20 MN=Y/X
      MN=MN+ST-1
      DO 30 J=1,K
      R=-ST+1-J
      Z=Z+((MN+R)**2)*LQE(J,LCN)
   30 CONTINUE
      P=Z/X
      STD=SORT(P)
      MNHRS=MN*LQEINT
      STDHRS=STD*LQEINT
      RETURN
      END
```

```
      SUBROUTINE STORES
      DIMENSION AVEQT(10),AVEUTL(10),BDATA(4001),FWQ(10),FWU(10)
     1          ,HUMACH(10),LOAD(10,100),LQE(51,22),MACHS(10,6),MTIME(10)
     2          ,PARTT(1750),QSET(10,100,3),RANK(100),SCF(50,8),SFWQ(10)
     3          ,STQQ(10),STSET(10,5)
      COMMON /C1/ALPHA,AVEQT,AVEUTL,BDATA,BETA,CAP,FLOAT,FWQ,FWU,HR,LOAD
     1      ,LQEINT,LT,LTSF,MACHS,MIN,MN,MTIME,OPT,PARTT,PROINT,QSET
     2      ,RLOAD,SCF,SFWQ,STQ,STQQ,STSET,WIPT,WORK
      COMMON /C2/BDL,BQ,FIN,FSHIFT,HUMACH,IP,L,LCN,LF,LQE,LTS,M,MAXQ,MCS
     5      ,MCSET,MPRO,N,NO,NXTBNO,OPNO,OPS,PRO,QUEUE,RANK,RULE,SETNO
     6      ,SL,SH,SHIFT,SHL,ST,TOTMCS,USE,WIP,WK,WKL
      INTEGER   AVEQT,BDATA,CAP,FLOAT,FWQ,FWU,HR,LOAD,LQEINT,LT
     1          ,MACHS,MIN,MTIME,OPT,OPTTIM,PRDINT,PARTT,QSET,RLOAD
     2          ,SCF,SFWQ,STQQ,STSET,WIPT,WORK
     3          ,OPTIME
     4          ,BDL,BQ,FIN,FSHIFT,HUMACH,IP,L,LCN,LF,LQE,LTS,M,MAXQ,MCS
     1          ,MCSET,MPRO,N,NO,NXTBNO,OPNO,OPS,PRO,QUEUE,RANK,RULE
     2          ,SETNO,SL,SH,SHIFT,SHL,ST,TOTMCS,USE,WIP,WK,WKL
     3          ,BCHADD,LOCATE
      REAL LTSF,MN,STQ
      I=LOCATE(SCF,M,NO)
      WIP=SCF(I,8)
      WIP=WIP-BQ
      IF(WIP.LT.0) GO TO 30
      SL=SCF(I,3)
      SH=SCF(I,4)
      IF(SH.LT.BQ) GO TO 10
      SH=SH-BQ
      GO TO 20
   10 SL=SL+BQ-SH
      SH=0
   20 SCF(I,3)=SL
      SCF(I,4)=SH
      SCF(I,8)=WIP
      RETURN
C
   30 WRITE(6,35)I,WIP,BQ
   35 FORMAT(/5X,'NEGATIVE WORK IN PROGRESS',
     1       /5X,'I=',I5,5X,'WIP=',I5,5X,'BQ=',I5)
      WRITE(6,40) (SCF(I,8),I=1,NO)
   40 FORMAT(/5X,10(G8))
      STOP
      END
```

```
      SUBROUTINE STRLQE(T,ST,FIN,LCN,LQEINT,ARRAY)
      INTEGER    ARRAY(51,22),ST,FIN,LCN,I,T
      IT=T/LQEINT
      I=IT-ST+1
      IF(IT.GT.FIN) I=FIN-ST+1
      IF(IT.LT.ST) I=1
      ARRAY(I,LCN)=ARRAY(I,LCN)+1
      RETURN
      END
```

```
      INTEGER FUNCTION BCHADD(BDATA)
      INTEGER    BDATA(4001),BDL,IL
      BDL=BDATA(1)
      DO 10 IL=2,BDL,8
      IF(BDATA(IL).NE.0) GO TO 10
      BCHADD=IL
      RETURN
   10 CONTINUE
C
      WRITE(6,20)
   20 FORMAT(//5X,'BDATA OVERFLOW')
      WRITE(6,30) (BDATA(I),I=2,BDL)
   30 FORMAT(/5X,'BDATA=',(/1X,16I8))
      STOP
      END
```

```
      INTEGER FUNCTION LOCATE(SCF,M,NO)
      INTEGER    SCF(50,8),M,NO
      DO 10 I=1,NO
      IF(M.NE.SCF(I,1)) GO TO 10
      LOCATE=I
      RETURN
   10 CONTINUE
C
      WRITE(6,20)
   20 FORMAT(/5X,'PART NOT LOCATED')
      WRITE(6,30) M,NO,(SCF(I,1),I=1,NO)
   30 FORMAT(/10X,'M=',G4,5X,'NO=',G4,/10X,'SCF(I,1)=',(/5X,25G5))
      STOP
      END
```

```
      INTEGER FUNCTION OPTIME(M,N,PARTT)
      INTEGER    PARTT(1750),M,N
    1 OPTIME=PARTT(M+2)*PARTT(N+1)
      RETURN
      END
```

APPENDIX 4

REFERENCES

1. LOCK, DENNIS 'Industrial Scheduling Techniques'. Gower Press, 1971, p.2.

2. IBID, p.2.

3. ROWAN, T.G. and CHATTERTON, A.N. 'Production System Audit'. The Production Engineer, Feb. 1974. p. 71-77.

4. GUPTA, J.N.D. 'M-Stage Scheduling Problem - A Critical Aparaisal'. Int. J. Prod. Res., Vol. 9, No. 2, 1971. p. 267-281.

5. BULKIN, M.H., COLLEY, J.L. and STEINHOFF, H.W. 'Load Forecasting, Priority Sequencing and Simulation In a Job Shop Control System'. Mgmt. Sci., Vol. 13, No. 2, Oct. 1966. p. B-29-B-51.

6. EILON, S. and HODGSON, R.M. 'Job Shop Scheduling with Due Dates'. Int. J. Prod. Res. Vol. 6, No. 1, 1967. p. 1-13.

7. DAVIES, T.R. 'The Integration of Material Control and Machine Shop Scheduling'. Unpublished Ph.D. thesis, Nottingham University, 1969.

8. McCALLION, H., HORSNELL, R., DAVIES, T.R. and BRITTAIN, J.H.C. 'An Adaptive System for Materials Control and Machine Shop Scheduling'. Unpublished research report, Nottingham University, 1970.

9. BRITTAIN, J.H.C. 'A Simulation Model for Machine Shop Scheduling'. Unpublished Ph.D. thesis, Nottingham University, 1969.

10. ASWED, M.M. 'A Smoothed Simulation Model for a Batched Production Shop'. Unpublished M.Sc. thesis, Nottingham University, 1971.

11. TRIGG, D.W. and LEACH, A.G. 'Exponential Smoothing with an Adapative Response Rate'. Operational Research Quarterly, Vol. 18, No. 1, 1967. p.53-59.

12. MELLOR, P. 'A Review of Job Shop Scheduling'. Operational Research Quarterly, Vol. 17, No. 2, 1966, p.161-171.

13. MELLOR, P. 'Job Shop Scheduling'. The Production Engineer, Vol. 46, No. 2, 1967. p.82-87.

14. GERE, W.S. 'Heuristics in Job Shop Scheduling'. Mgmt. Sci., Vol. 13, No. 3, Nov. 1966, p.167-190.