PHD

The buffering of transfer lines

Owen, Geraint Wyn

Award date:
1994

Awarding institution:
University of Bath

The Buffering of Transfer Lines

submitted by Geraint Wyn Owen

for the degree of PhD

of the University of Bath

1994

Summary

Transfer lines are typically used to machine single prismatic type components with high demand. Typically, they comprise of many automatically linked machines, the number of which reflects the component complexity and required cycle time. Stoppages of individual machines due to breakdowns and for tool changes greatly reduce the output of the lines.

Previous research has concentrated on the analytical study of short lines. This has failed to present a clear understanding of buffering requirements for longer lines, or any detailed guidance of how best to buffer a line.

This thesis describes how simulation has been used to study the effects of machine stoppages on the output of the line, and how the output can be improved by inserting buffers of components between machines. The resulting significant increases in output are presented. These show that the positioning of buffers relative to bottlenecks on the line is as equally important as the amount of buffering added to the line. A methodology has been developed which allows a 'near optimum' buffering pattern to be generated for complex lines. The resulting buffering patterns compare favourably with other published buffering strategies.

A case study using data from a real line shows that the buffer pattern generated by the methodology gives an improvement in output (or alternatively a reduction in the buffering needed for the same output) when compared against the existing buffer pattern derived by expertise and best current practices. Area of further development are also presented.

Acknowledgements

Firstly I would like to express my sincere thanks to my supervisor, Dr. Tony Mileham, for his keen and enthusiastic support and guidance throughout the course of this research. I am also grateful to all my departmental colleagues for their assistance and contributions to my work.

Acknowledgements are also due to the Science and Engineering Research Council for providing financial support.

Finally I would like to thank my family and friends for help, support and encouragement throughout the project.

Table of Contents

List of Figures

List of Tables

List of Symbols

Symbol	Name	Units
O	Efficiency of the line	%
N	Number of machines	
C	Cycle time	Mins.
B	Mean number of components between failures (can be refered to as breakdown frequency , even though it is measured in terms of components)	
R	Repair time	Mins.
D	Downtime	% of machine uptime
BC	Buffer capacity	No. of components
S	Average inter-machine buffer capacity	No. of components
T	Time	Mins.

Chapter 1 - Introduction

There are two elements to any production process. The actual operations by which raw materials are converted into the finished product, and the way in which those processes are organised. Significantly more is known and more research has been carried out into the processes themselves. Less, however, it would seem, is known about the way these processes should be arranged and controlled. Consider the number of control ideas and philosophies that have been applied over the last 15 years (MRP, MRP II, JIT, FMS, GT, TOC), all of which have been based around new ideas of material control and machine organisation. Yet none of them have been adopted to such an extent that a majority of companies are now operating using successful versions of these ideologies.

Transfer lines are typically used to machine single prismatic type components with a high demand (some rotational parts (e.g. camshafts) are machined on linear lines, but it is more common to find rotational parts produced on rotary machines). They comprise of many linked machines, the number of which reflects the component complexity and the required cycle time. Division of labour into separate operations on the same job has occurred down the ages and transfer lines are a continuation of this idea, where each machine does one or more operation to each component before passing it to the next machine. True transfer lines were first developed in the 1920s for the automobile industry where the need for dedicated equipment to continuously produce one part, rather than many different parts in batches, was first required. Today vastly more complex lines are built to produce a wide variety of parts in many different industries. Typical uses are still found in the motor industry where examples of lines $^1/_4$ mile long costing up to £150million can be found producing engine blocks at a rate of around one every 20 seconds.

The complexity of designing such a large line is enormous. Pressures to get a newly designed engine into the latest model of car mean the time scales involved put further pressure on the designers. Coupled to this is the added concern of getting the line design right first time when such large sums are being invested. Yet there are few, if any, new philosophies, methodologies or rules to assist the line design team.

The elements involved in designing such transfer lines are shown in Fig 1.1. For a given component, the process plan is drawn up in conjunction with a required cycle time to correspond with expected demand. Operations are then divided between machines (or stations) in an attempt to produce a 'balanced' line (all machines having the same cycle time). In order to achieve a balanced line, the process times can be altered by varying the tooling. A great deal of effort is put into this step, and often new improved tooling is created. Changes at this stage can result in the reallocation of processes as shown by the loop #1. Having determined the processes and machines, the physical layout is determined subject to any physical constraints.

Breakdowns on the line have a significant effect on output. For example, if each machine on a 30 machine line is broken down for 10% of the time, the overall output of the line will typically be less than 40% (see Chapter 6). The effects of these stoppages can be compensated for by placing storage space for components (buffers) between machines. At several companies (e.g. Ford), the buffering level is initially derived from the layout (i.e. using the length of the queuing conveyors between machines). Extra buffers are then placed adjacent to machines which, from breakdown performance data of similar machines, are considered to be likely bottlenecks. Large volumes (up to three days worth of components) can then be added to either end of the line, creating the traditional *Raw Materials Store* and *Finished Parts Store.*

The final element of the design process is to determine the operational strategy for such elements as tool changes and required manning levels.

Normally when designing a line, the steps described above would be linear (though there are reiterations on the loop #1). There is, however, a tendency not to reiterate the whole design process once the buffers have been added (as in #2). One of the reasons for this is the lack of information on how buffering should be best achieved, together with a lack of understanding on the part of many designers as to how and

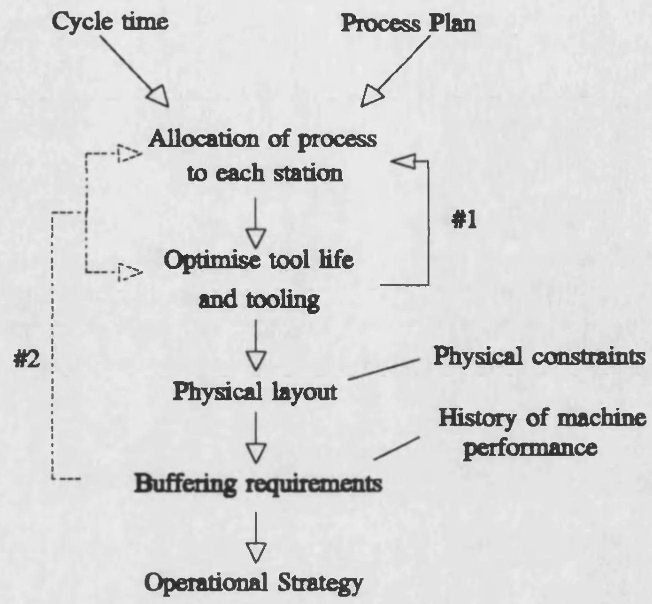


Fig. 1.1 The design process of a transfer line.

why buffers do so significantly improve output. If buffering could be easily, quickly and accurately assigned without the need for the high levels of experience currently used and without the need for the many simulation runs to check and recheck the design, the optimization of the tooling and physical layout could be repeated without increasing the design time. This feedback (as shown by #2) will, therefore, lead to better designs produced in a shorter amount of time.

Thus the aim of this research is to assist the line designer at the buffering stage of the design process by the provision of a greater understanding of the way buffers improve line output together with a methodology for optimumly buffering transfer lines.

1.1 Aims

Producing a global method by which the whole transfer line design process could be achieved is a mammoth task. In order to achieve this, for each element of the design process, a method must be developed to produce the local optimum configuration. Only once this has been done can these local optima be linked and then globally optimised to produce the ideal line. It must, however, be realised that any global optimum will rely on a compromise between the local optima.

Although far more is known about the processes themselves, techniques do exist, such as line balancing, for solving certain organisational problems. There still, however, remains several parts of the process which as yet have no clearly identifiable rules to allow any local optimums to be achieved.

Chapter 3 describes some of the work that has been done, but as is discussed there, little, if any, of this work provides much help to a line designer. Based on these shortcomings, the objectives of this research are:-

- To investigate the factors that affect transfer line performance with particular emphasis on longer lines.
- To develop an understanding of how and why buffers improve line output
- To derive optimum buffering strategies for transfer lines

To avoid the pitfalls of analytical study described in Chapter 3, simulation has been used to study the lines. By using this method, results for different input parameters can be found relatively quickly and the effects of stoppages on the line can be studied visually. The benefits of this approach should be that results obtained will be both realistic and practical.

If the end results of the work are to be realistic with practical application, then the lines being studied must also be realistic. To accomplish this, the following constraints have thus been applied to the lines studied in this research:-

- All lines are assumed to have a non-synchronous control system.
- All stations are automatic.
- All lines are long (over 20 machines). These are considered to be far more relevant than the 3,4 or 5 machine long lines used in most previous analytical studies.

Initial experimentation concentrates on simple lines so an understanding of the simplest parameters such as line length, cycle time and breakdown frequency can be gained. The simulations are then made more complex to include bottlenecks and buffering. This provides an understanding of how buffers work, and consequently the development of the optimum way to buffer both balanced and bottlenecked lines.

In order to reduce the complexity, the following constraints have also been imposed:-

- Only machining lines are considered.
- Lines are fully balanced (machines have equal cycle times).
- The objective of any buffering is to maximise output.
- That cost considerations are not included. This can be easily brought into a buffering strategy at a later date, provided the cost of adding buffers at a point is linear.
- There are no physical constraints on buffer placement.
- There are no labour constraints.

Chapter 2 - Transfer Lines

Flow production can be divided into two sections. Firstly, *flow processes* are used for products which themselves continually flow such as petroleum refining or bulk food production. Secondly, when parts are being produced in discrete operations rather than in a continuous flow, the term *Flow Line* is often used. In a typical machining environment the flow line operations being carried out on a part typically take place on a series of separate machines in which the part must be transferred from machine to machine. When this is the case, the term *Transfer Line* is used to describe the process. The terms transfer line and flow line are, however, often used synonymously in the engineering community.

Although Wild [1972] claims that the basic principles of flow production date back at least 500 years and possibly as early as the 4th century BC, the first examples of what could be called transfer lines were developed in the early 1920s by Henry Ford for the assembly of the Model T. Although strictly *transfer assembly lines*, they were used as a base to pioneer the work done on mass production assembly using flow lines. This then resulted in the development of transfer lines for actual metal removal, the first of which was built by Morris in 1923 [Production Engineer, Feb 1955]. The origins of the first lines together with the input of the automobile industry into their later development has resulted in this of this kind of production being referred to as *Detroit Automation* [Groover, 1980].

2.1 Definition of a Transfer Line

Any collection of machines where the workpiece is transferred from one machine to the next can be said to be a *transfer line*. The term transfer line is, however, normally only applied where the components are transferred between operations automatically. The complexity and number of machines is dependant on the components being worked on and the required cycle time. This ranges from a small number of operations on a rotary type machine no bigger than a pillar drill, working on a simple prismatic part, through to a fully automated line up to 0.5 miles long [Crosby and Murton, 1990], with numerous machines, working on large complex parts (e.g. Machining of an engine block or cylinder head casting). Transfer lines can be used for both machining components and assembly type tasks. Figures 2.1 and 2.3 show schematic arrangements of example transfer lines.

2.1.1 Machining operations using transfer lines

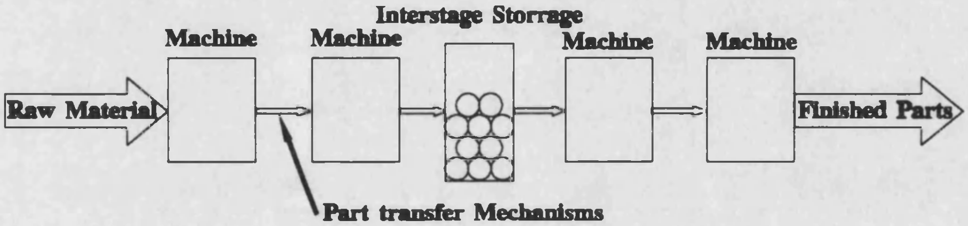


Fig. 2.1 Schematic view of machining type Transfer Line

A typical machining line will consist of many dedicated *machines* (or *stations*) all inter-linked by a transfer mechanism, which carries the workpieces between the stations. At each station an operation will be carried out on the workpiece. Each station can consist of more that one *head* (spindle of a machine tool) with 3 heads being a typical maximum. Thus if 15 operations are to be performed, they may be carried out at 5 stations each with 3 heads. If one head breaks down, all other heads at that particular station are forced to stop. Depending on the transportation method (see Section 2.2), other stations may, however, carry on provided they have a supply of components and somewhere to output their finished work. It is important to note that different people use different terms, but the definitions above will be used throughout this thesis. For example, at Ford [Ford GB], a line is called a machine and this consists of several stations, each with several spindles.

Large transfer lines can and do have a wide variety of machining operations, including the more obvious such as Drilling, Milling, Turning, Boring and Tapping, as well as some less so such as Grinding, Honing, Washing, Broaching, Polishing and Inspection.

It is unusual to find different operations on the same head. A multi-headed station would for example, usually be drilling 8 holes, rather than 6 holes and a turning operation. An example of a transfer line is shown in Figure 2.2.

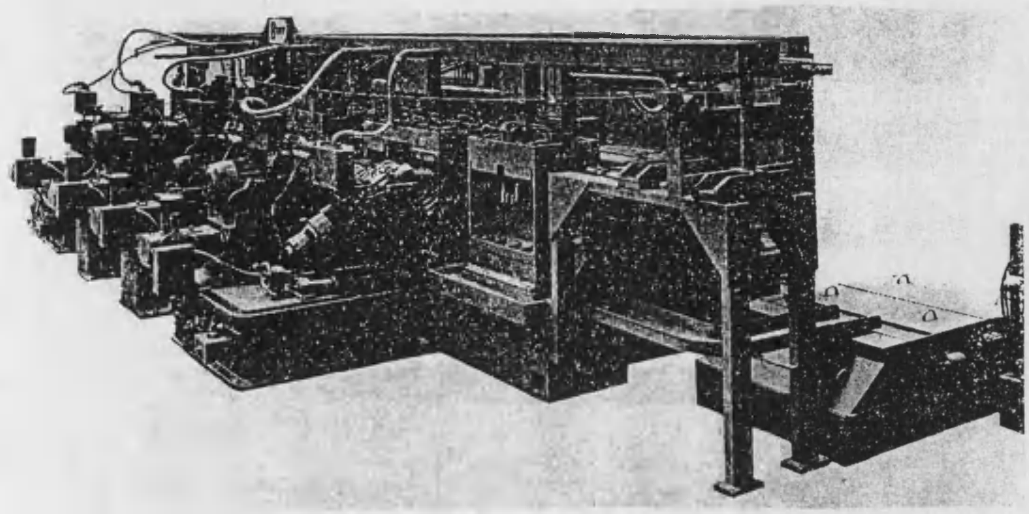


Fig. 2.2 An In-Line type transfer line

The ideal line design has all operations having the same cycle time. This is known as a *Balanced* line. When designing lines, there are various techniques of balancing lines by separating the operations (e.g. Ranked Positional Weight and Kilbridge and Wester methods - see Wild [1980]). Where operations do have different cycle times, cutting speeds and feed rates can be altered on the slower machines in order to achieve a balanced state or new tools and/or tool materials can be developed to increase the speed of the operation.

2.1.2 Assembly transfer lines

An automated assembly line has similar connections between stations as a machining transfer line. At the stations, however, instead of metal removal operations, there are assembly operations. A typical station will have a parts store for the components being fitted to the assembly, a means by which those parts are orientated (e.g. Vibratory Rotary Bowl Feeder) and a mechanism by which the parts are fed or placed on the assembly.

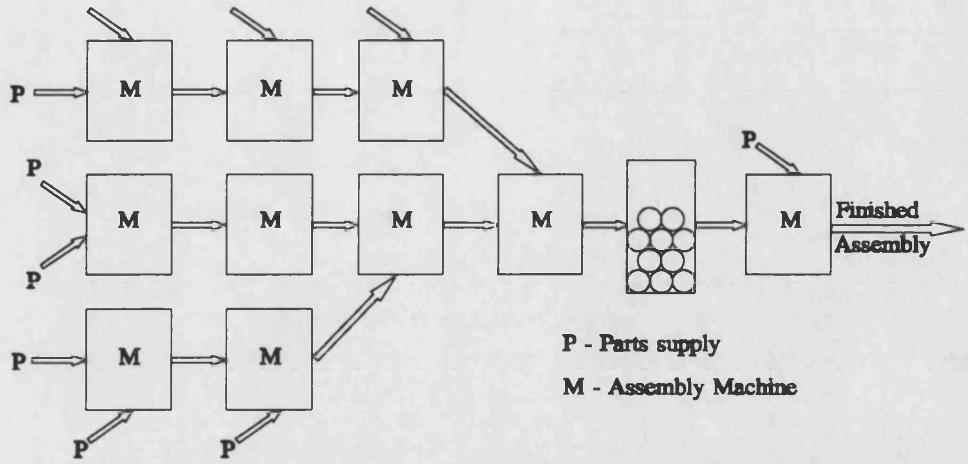


Fig. 2.3 Schematic view of an assembly type transfer line

If sub-assemblies are being assembled in isolation and then fitted to the main assembly, the line will appear tree shaped, with many branches leading to the final line. This is particularly true in larger assemblies such as a car. Section 2.6 goes further into this idea.

Transfer lines and assembly lines differ in the nature of their stoppages. Where transfer lines are primarily stopped by tool changes, failures and machine breakdowns, assembly line stoppages are primarily caused by components jamming in feed mechanisms and defective components not fitting or allowing other parts to fit.

Where there is a need for *'feel'* or *'sight'* to assemble parts it is common to find manually operated stations in the line. This is also true of inspection stations where humans can inspect a wider variety of attributes than some forms of mechanisation might (see Section 2.1.4).

In terms of line design, manual operations in any balanced line with component transfer can cause problems due to the natural variation in cycle times of human operators, as well as possibly a higher frequency of stoppages (toilet, tea breaks, etc).

2.1.3 Food industry

There are many applications of transfer type machines in the food, bottling and packaging industries. There are, however, some underlying differences. Cycle times are typically very small (canning lines can run at 600/min). In order to allow for stoppages, buffer stocks between machines are very large (swirl tables in bottling plants with over 1000 components), but because the value of the product is low compared with machined components, large volumes of WIP are financially viable. This type of production may even be considered to be a continuous process rather than discrete events of production, since the cycle time is so small. Therefore, although some of the work in this report may overlap into these areas, it is not intended to refer to it to any great extent.

2.1.4 Automated inspection

In the same way as it is desirable to automate the machining and assembly of components, inspection on transfer lines can also benefit from automation. Detection can be made as soon after a process as possible (or even during in some cases), and the self correcting of that process can be achieved through 'closed loop' feedback (e.g tool offsets being changed to allow for component size variations).

To check component size on machining lines Automatic Gauging Equipment (AGE) such as LVDT's (linear variable digital transducers) is frequently used. Machine vision is also available, but is more commonly used for assembly operations to detect whether components are present or not.

2.2 Variations in line types and configurations

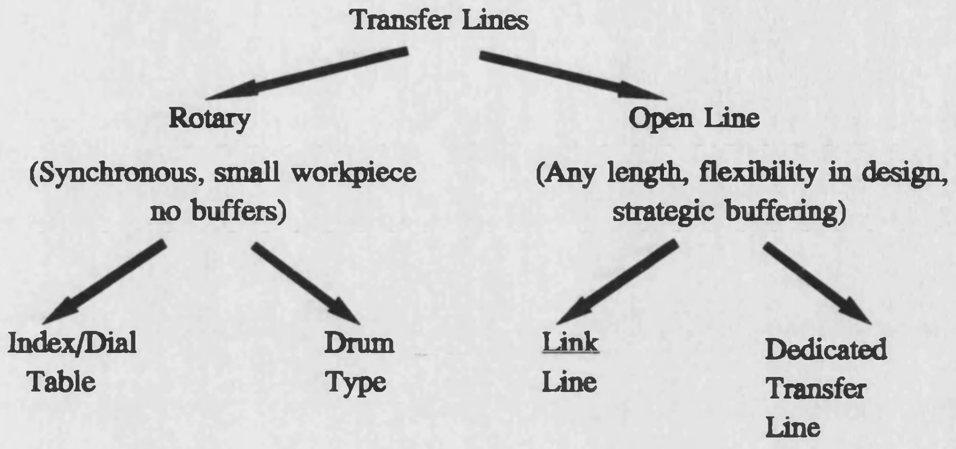
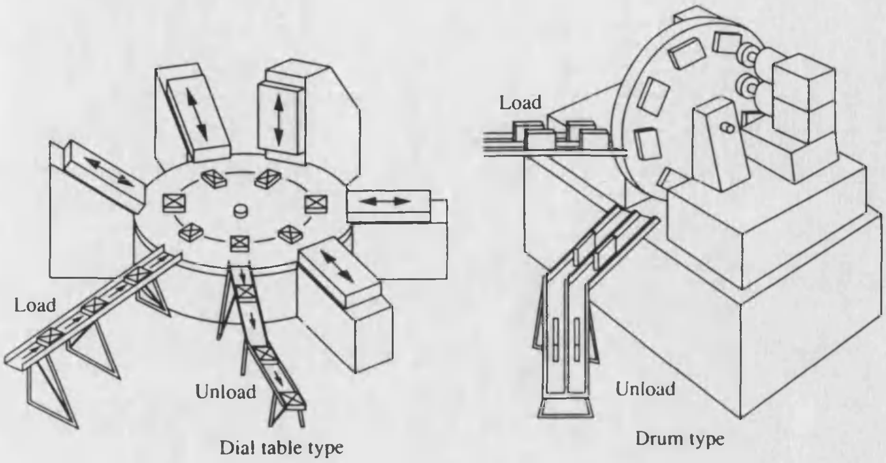


Fig. 2.4 Transfer line hierarchical classification

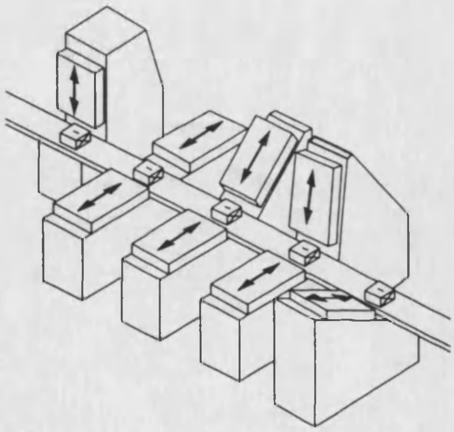
The transfer system can be configured in two main forms. For small components with few operations, a rotary type machine can be used. Due to space constraints around the circle it is normal to have a maximum of 6 stations on such a system (see Fig. 2.5a). These machines tend to be very compact and require comparatively little floor space, a maximum being around 10 feet in diameter [Ryder Machine Tools].

Larger and more complicated parts are processed on open lines (see Fig 2.5b). With an open line there is no constraint on length and hence the number of operations. The line need not be straight, U and W shapes are common as they allow the supply and removal of parts from the same end [Yeoh, 1982] or rectangular as these allow the same worker to load and unload the workpieces [Groover, 1980].

As well as fully mechanised transfer lines using dedicated machines, there are also *linked lines*. A linked line consists of a series of standard machines all interlinked with a specialised transfer system. By using standard machines with a special transfer mechanism, the cost of the line is significantly less. The machines can also be 'recycled' into new lines when the current line is dismantled, with a further cost saving.



a) Rotary type



b) In-line type

Fig. 2.5. Transfer line configurations [from Black, 1991]

There are three different methods by which the workpiece is transported :-

Continuous The workpiece is moved at a constant speed all the way down the line. The heads of the machines must move along the line in order to remain in the same relative position to the workpiece. Although this is not practical for machining operations, this configuration finds several uses in assembly operations and in the food industry.

Intermittent The workpiece moves in a stop-start manner from machine to machine. At each machine the workpiece is then located before any operations commence. All workpieces move together at fixed intervals, and because of this, this method is also termed a *Synchronous Transport System* [Groover, 1980].

Non-synchronous The workpiece moves from machine to machine independently of other parts. Some workpieces can be transferred while others are being processed. There is a greater flexibility with this kind of transfer system. Buffers of components can be built between different operations allowing a degree of in-balance on the line. This is particularly useful if manual operations are involved as it evens out the work time variances. It is because of this greater flexibility that the majority of large transfer lines use this method of workpiece transfer.

In actual line design it is commonplace to see a combination of intermittent and non-synchronous transportations on the same line. Several short intermittent lines can be coupled in a non-synchronous manner as shown in Figure 2.6.

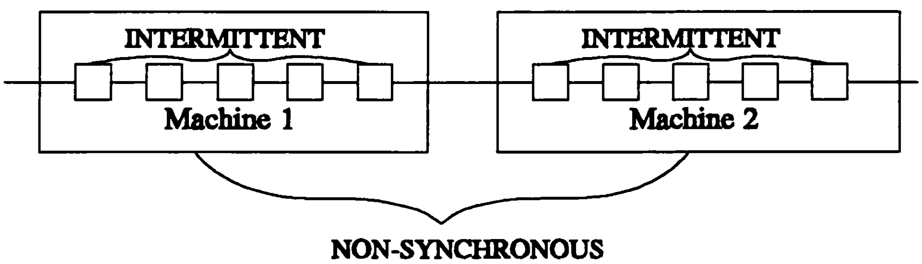


Fig. 2.6 Combination of Intermittent and Non-synchronous Transportation

Where there is insufficient capacity, parallel machines can be connected. These can either be in operation full time as in Figure 2.7a, or be standing by Figure 2.7b.

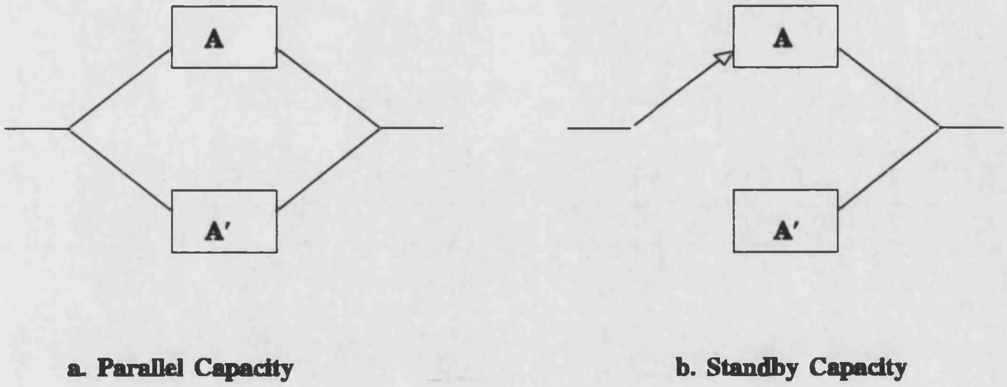


Fig. 2.7 Configurations of parallel machines for extra capacity

2.3 Typical uses and output levels

Due to the high initial capital investment required to build fully mechanised transfer lines, their use is typically confined to very high volume discrete part production, with projected production being constant over a length of time (typically 3 years minimum). The initial capital investment is dependent on the length of the line (each machine consisting of several stations costs approx. £1m) [Crosby and Murton, 1990] which is in turn dependent on the component complexity. Lines for large complex parts would consist of many machines and may cost many millions of pounds.

As an example of the scale of costs and production levels, consider Ford's recent development of a transfer line to machine cylinder blocks for their new Zeta engine. The line was approximately ½ mile long with an initial capital investment of £70million. The result was a line with an expected output of one unit every 17 seconds. In contrast, Rover Group's new K-Series Engine [Rover] production line at Longbridge had a 74 second cycle time (approx 1000 units/day). The line cost between £150-£200million. The reason for the high cost, but lower output is that several of the machines on the Rover line were flexible NC machines. This was done to allow quick change overs between different engine configurations (both in engine capacity and 8 or 16 valve options).

The output efficiency of transfer lines varies according to the type and length of line. There are also several performance measures that can be used including output rate, %uptime, and cost/unit of output (See Appendix A). Overall line uptime is rarely as high as those who purchase transfer lines would expect, but values vary from as low as 40% up to 90% with 70% being typical (see Section 6.1).

2.4 Transfer line stoppages and distributions

As with any complex equipment, individual elements of transfer lines each have varying reliability. Each individual head must stop for a tool change (both planned and unplanned) or the machine may break down for any number of reasons, both electrical and mechanical.

On non-synchronous type lines, a machine will cycle provided that there is a supply of workpieces to the machine and there is somewhere to output the finished piece. When a machine on the line stops two things happen; all machines downstream eventually run of work as the line runs 'dry'; machines upstream are forced to stop as they have nowhere to output finished parts. Buffers can be used to help offset the effects of stoppages (Section 2.6).

This gives 4 machine states :-

Busy	The machine operating normally.
Down	The machine stopped for tool change or due to breakdown.
Blocked	Unable to cycle due to output blocked.
Starved	Unable to cycle through shortage in supply.

The frequency of stoppages is typically represented by an exponential distribution [Crosby and Murton, 1990][Witness Modelling Notes], where the likelihood of failure is independent of time. There are, however, other distributions that can be used.

Repair times have been shown to conform reasonably well to an Erlang distribution with a K value of 2. This is equivalent to sampling twice from an exponential distribution; once to detect the failure, the other to repair it. Again other distributions can be used if the Erlang distribution is considered to be unsuitable (see Section 4.5.3).

2.5 Interstage storage

Interstage storage or buffers can be placed between machines on transfer lines. There are two reasons for the inclusion of buffers in a line. Firstly, buffers can be used to compensate for variations in cycle time between consecutive machines. This is particularly useful when there are human operators, typical in assembly applications.

The second reason for including buffers is to compensate for the effects of machine breakdowns and tool changes. If there is no buffer store between machines, when a machine stops all machines upstream will become blocked. Buffers upstream of any stoppage can be used to store finished components and allow machines upstream of the buffer to continue cycling until the buffer becomes full. Where machines downstream are becoming starved, buffers can feed components downstream of the failed machine and keep the downstream machines busy until the storage becomes empty and machines become starved. A more detailed explanation of the mechanism of blocking and starving is given in Section 5.2.

There are several different types of buffer, which vary in capacity and price. The main ones are:-

Automatic Racking - These are typically, but not exclusively, used at the beginning and end of lines. They are fully automated with no manual intervention and consist of one or more racks where the parts are stored (see Figure 2.8). Ford proposed the installation of automatic racking at their Bridgend plant at a cost of £1million for a maximum capacity of 600 engine blocks.

Queuing Conveyors - These are by far the most common buffers used in transfer lines. Typically they consist of a series of powered rollers, which when the component part is stopped, (by reaching the end stop or a queue of other parts) disengage from the drive (see Fig 2.9). Since conveyors are normally used to transfer parts from machine to machine, using a queuing conveyor rather than a fixed conveyor (where when a part reaches the end the whole belt stops), allows buffering to be built into a line easily and cheaply. The current price of queuing conveyors is approx. £600 per meter fully installed [MCM Conveyor Systems, 1994 prices], fixed conveyers are typically cheaper, but not significantly so. The buffering capacity of a queuing conveyor is governed by its length in relation to the linear dimension of the parts along the conveyor. Although this type of buffering is cheaper than automatic racking, it does however take up a greater floor area and it is not suitable for storing large volumes.

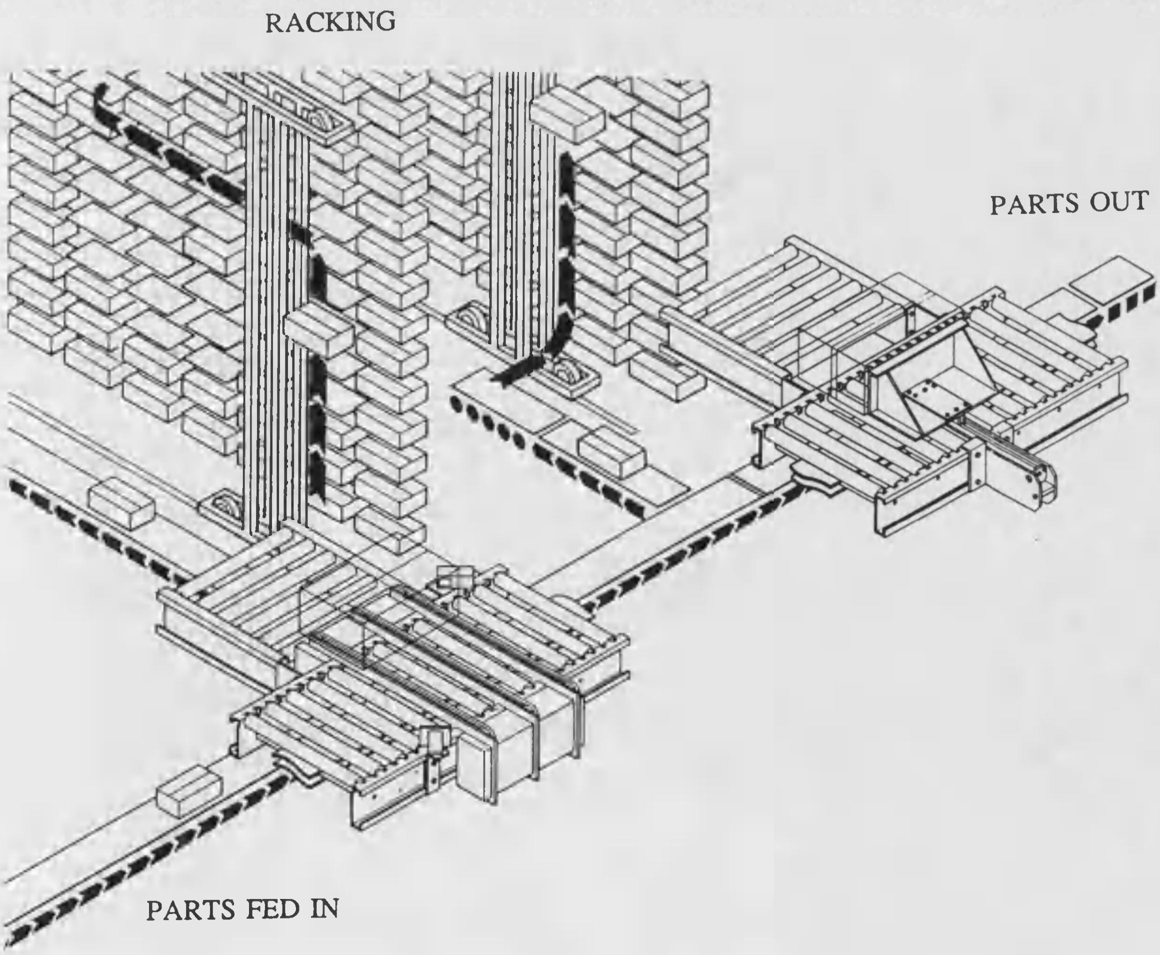


Fig. 2.8 A Schematic of an automatic racking system [from Dexion Conveyors]

Pallets - In the event of a stoppage, parts can be manually loaded or unloaded from conveyors between machines and placed on pallets or stillages for storage away from the line. Provided the parts can be easily lifted from the line and do not need to be held in any special way, using pallets can be a cheap and easy method of buffering a line. A further advantage is that pallets can be placed anywhere on the line, so the buffer capacity can vary from place to place on a day to day basis. The use of pallets does rely on the cycle time being long enough to allow manual loading and unloading, and manual stacking could result in component damage.

Swirl Tables - These are frequently used in bottling and canning plants where large buffer capacities of small components are required. They consist of a circular conveyor which rotates to keep the parts moving. Parts are fed to the middle and then collected from the outside and can be used to accept batches and give a paced output of components.

The designed use of buffers in industry is limited. Typically buffer sizes are only determined when decoupling different production sections. The Rover K-Series production described in Section 2.3 is a good example. Between the casting facility and the machining line, there is a manually stacked buffer with a maximum capacity of approximately 1000 parts. This is equivalent to just over one day's work for the line. A similar size of buffer is placed between the machining line and the assembly line, but this is fully automated. It would seem that there was no finesse or calculation used, merely a desire to decouple the three 'elements'. On the machining line itself, there are 100 heads arranged in 25 machines. The 25 machines are in 8 groups of 2,3 or 4. In between the groups are queuing conveyors, but there is no buffering between machines in a group, thus the line can be said to be 8 intermittent groups coupled non-synchronously. It seemed the grouping was based on process planning requirements rather than buffering needs, and the length of the queuing conveyors (and hence their capacity), was determined by plant layout requirements again at the expense of any buffering needs. At this point it is worth noting that to increase buffering capacity in between the machines, the operators of the line would manually unload the middle of the line onto pallets, which could then be 'fed back in' at a later date.

2.6 The transfer line within the total production system

If buffers can be used to isolate and decouple different parts of the line from each other, then it can be said that the whole production, of say a car, can also be considered as one transfer line. The line is then a combination of machining and assembly operations, and runs under several control systems and as such is a pseudo line. The total production system does, however, exhibit the same characteristics as a transfer line, with different machining and assembly lines being isolated from each other by buffers. The system is also unbalanced and thus the problem is synchronisation. Figure 2.1. shows the path of a typical machined component, for example the engine cylinder head. If this transfer line is then fed into the engine assembly line (represented in Figure 2.3) then the production of the engine can be considered to be one long line. This one line would be Tree shaped, typical of any Sub-assembly/assembly line, however, the ends (or 'branches') would themselves be long machining type lines. This is the case in practice, but the machining lines are buffered from the assembly lines to a greater extent than any parts of the machining line are from each other. It is still obvious that if the production of the component parts stops, then eventually the assembly work will have to stop also.

2.7 Why automate with transfer lines?

The main justification for any capital expenditure in a business environment is to reduce costs. If the total capital expenditure needed for automation by transfer line technology results in a lower overall product cost during the projected life of the line when compared with the alternative and/or current manufacturing techniques, then it can be considered a worthwhile investment. With the capital cost of the equipment varying greatly, depending on the type of line (which in turn depends on the component complexity and the required output levels) and the required volumes for various components, the answer to the question as to whether to invest or not will vary from component to component and from company to company.

Groover [1980] gives 6 reasons to automate using transfer line technology:-

1. Reduce labour costs.
2. Increase production rates.
3. Reduce WIP.
4. Minimise distance between operations.
5. Enable specialisation of operation.
6. Integration of operation.

There is also the added advantage of faster throughput time.

It must be questioned whether the above are all advantages. Certainly reducing labour costs is an advantage in most cases. An increase in production rate is only an advantage provided products produced are saleable. Minimising the distance between operations reduces the floor space required for production, and can therefore be considered as a cost saving. Specialisation in operations can, however, reduce flexibility, and although flexibility may not be an issue with many dedicated transfer lines, in cases where the product life is short it can be an important consideration. Finally, integration of operations is advantageous, however, the use of transfer lines is not a unique way of achieving this.

Acherkan [1969] adds 5 more advantages

1. Reduce time due to less materials handling.
2. Higher machine utilisation.
3. Higher floor space utilisation.
4. Less workpiece damage due to handling.
5. Higher quality.

Many of these advantages overlap, to a greater or lesser extent, with those claimed by Groover. The quality aspect is, however, worth considering. With a total integrated dedicated transfer line lower scrap rates can be expected. A great question mark, however, surrounds the higher machine utilisation. Time spent up and running on dedicated transfer lines can be as low as 40% although 70% is more typical. Typical figures for conventional machines in a batch environment, which transfer lines are in competition with, are also around 70%.

Acherkan also lists some disadvantages:-

1. Closer tolerances needed on incoming work locations.
2. Breakdowns on some machines may cause others to stop.
3. High product change over times.
4. High initial capital expenditure.
5. Need for highly skilled service personnel.
6. Operational development can take a long time (shallow 'learning curve').

The above disadvantages are self explanatory. The question of higher product change over times is twofold. Firstly there is the lack of flexibility with any dedicated machines. This results in high set-up times when changing from one product to another and back again unless the line has been designed for a given range of products (e.g. Rover's Longbridge line described in Section 2.3), particularly when compared with an FMS type manufacturing system. The second problem with change overs is when a major permanent change takes place. There is a lead time in getting the production system in place once the new product has been developed. Consider a car manufacturer producing a new engine to revitalise its old model. It may take as long as two years to design and build the new line. Although much of this work is done in parallel with product development, the time span can not completely overlap and thus product development time is increased. Also, due to labour shortages it may not be possible to run the old line in parallel with the new one, resulting in a sharp fall in production when the new line comes on stream and the need for stockpiling.

Groover [1980] also lists disadvantages in terms of society. The use of automation will cause unemployment, which in turn reduces demand and hence more unemployment, all in a vicious circle. Also, the work that is to be done will be menial and require little skill. This kind of argument is unsound and can be extended back to such a degree that all industry should still be carried out in a blacksmith's shop and that the industrial revolution should have been avoided. There are many other merits both in favour and against this argument, but it is outside the scope of the work being presented in this thesis.

Chapter 3 - The Extent of Current Knowledge

This chapter aims to present a critique of the main ideas published in the field of transfer line design and identify any areas where further work is required.

Possible elements of line design for study include the number of machines and cycle time, line balancing, provision of buffers, tool change strategies and manning levels. The work described in this thesis is, however, concerned with breakdowns and how buffers can be used to improve line output when machines breakdown. Previous research has been done on lines both with and without buffers, some has used a simulation approach, some an analytical approach, some has presented the bowl theory and some has used the Theory Of Constraints (TOC). In this chapter it is hoped to describe the work in all of these areas and examine any published rules or heuristics for line design and in particular buffering techniques.

Several pages of Wild's book [1972] are dedicated to the historical development of flow methods of production. From its origins, in the division of labour as early as the 4th century B.C., the development of flow methods can be traced through to the present day. Some of the major milestones identified include the need for interchangeability of parts during assembly, particularly in the manufacture of arms during the American civil war, and the role of the automobile industry in this century.

The first research published into what can be truly described as the design of transfer lines was in the late 1950s (e.g. Hunt's [1956] Markov Chain analysis on two and three stage lines which has been the basis of most analytical study since). Other analytical studies were carried out during the 60s and early 70s, but all of this work was obstructed by the complication in the mathematics when lines of over 4 or 5 machines were studied. The development of the digital computer during the 70s allowed the study of lines to be carried out through simulation. As the use of computers has grown, and dedicated simulation software has become available for PCs where little specialist programming skill is needed, simulation has become more widely used as a research tool. The analytical studies are, however, still continuing today, but they are being, I suspect, as most analytical studies in the past seem to have been, carried out by mathematicians and statisticians, who are more interested in the mathematical development by trying to apply existing or newly developed statistical ideas to practical problems rather than improve industrial efficiency. A point worth noting is that analytical work tries to give exact results in a complex environment, whereas computer simulation demonstrates a particular feature rather than proving it.

Although the work described in this thesis is concerned with machining type transfer lines, there are parallel areas in assembly type lines (eg Hopp [1993]) and in continuous flow processes. The nature of the stoppages in assembly lines differ from those in a machining lines. The effects of a stoppage go back up the supply tree and the study of these lines is much more complex. In continuous flow processes, the line can be modelled as fluid flowing in a system, where tanks can represent buffers and machine stoppages can be represented by taps. The cycle time of the operation is then governed by the length and diameter of pipe between tanks. There is also a possibility of this work overlapping into areas of computer design where buffers are placed between different devices communicating at different rates. The work in this area is very simplistic as there is no need to deal with breakdowns, merely differences in rates and package size, and no published work has been found in this area.

3.1 Lines without internal storage

The majority of previous research work has involved lines with buffers. Any work that has been carried out on unbuffered lines has nearly always been used as a benchmark by the author for work on buffered lines. The reason for this is obvious, by studying lines without buffering there is one less parameter to be concerned with and the problem is correspondingly less complicated.

The most simple analysis is on intermittent or continuous lines without buffers. Here, when one machine stops, all machines stop. The resulting analysis involves taking the time each machine spends stopped in a given period. The sum of the total stoppage time for all machines is then compared with the total available time to give the percentage up time of the line. This analysis has been published by several authors, a good example of which is Groover [1980]. Groover also develops this idea further by presenting upper and lower bound solutions depending on whether the part is scrapped or not.

Buzzacott [1967,1968,1971] has been one of the main contributors in the area of transfer line research over the last 20 years, most articles cite to one or more of his papers. One of his earlier papers [1967] is based on intermittently controlled lines and considers the connecting of parallel, splitting and standby machines, and compares the results against a single station. The results are not presented in the context of a whole line, but they conclude that parallel machines should be used if a single station is unable to meet the required system production rate. If the individual station is capable of meeting the production rate, but machine breakdowns are causing a bottleneck, then the choice of whether to add redundant capacity and whether it should be arranged in a standby or splitting format is greatly dependant on the individual lines characteristics and required percentage utilisation. He also presents similar ideas and results as Groover for upper and lower bound solutions.

For a non-synchronous line the problem is more complex, since when one machine stops, others may continue to cycle provided they have a supply of parts and somewhere to place the finished part. The main parameters that can be studied on such a line are variations in cycle time, breakdown patterns and line length.

The majority of the work carried out by the author [1991,1992,1993 - copies included in Appendix IV], which is described in this thesis, is based around the assumption that any transfer line studied is balanced in terms of cycle time (i.e. all machines have an equal and constant cycle time). Although in reality this may not be true, it is an ideal towards which the designer may strive when designing a line. This is because, if output is maximised the slowest machine on the line must be operating 100% of the time. Any other with a faster cycle time has excess capacity that is not being used. At

Fords, they will in fact slow a machine down to match cycle times and at the same time attempt to increase tool life. This view is in contrast to some research described in section 3.2 and will be further discussed there.

Throughout this chapter many of the investigations described do, however, utilise variations in cycle time as well as in stoppage patterns. The effect of variations in processing time on a line with no internal storage and no machine breakdowns has been investigated by Conway et al [1988]. No specific relationships were developed, but it was noted that blocking and starving on a two machine line could drop capacity by 15% of that of a single station. The more machines that are added, the lower the output, but the significant loss in capacity occurs in the first 5 machines and additional machines cause little effect. Their work is simulation based using the graphical simulation package XCELL, which they wrote themselves. Of all the work described in this chapter, it is the author's opinion that the work by Conway et al is the most valid.

It has been proved by Yamazaki & Sakasegawa [1975] that the output of a given transfer line is matched by an identical line with the material flow in the opposite direction. The implication of this result is that it may not be necessary to try every combination when assessing different line configurations. The work was, however, carried out analytically on relatively short lines (less than 10 machines). Whether this rule is still true when the line length is increased is not shown. In a non-synchronously controlled environment, blockage effects travel upstream instantly, while the starvation window produced travels downstream at a rate equivalent to a component moving along the line. This difference in the speed that effects travel along the line is not noticeable on short lines, but the question as to whether it causes a difference on long lines, resulting in un-symmetrical lines, is unanswered.

On the whole the work on unbuffered lines is of little use to the line designer. Although there are further comments on these lines in the next section, the fact remains that there is little in the way of rules, heuristics or methodologies to ensure that lines are built in a way so that any goal, be it minimum throughput, minimum

WIP, highest output or lowest cost per unit produced, is achieved. With the exception of the work by Conway, the work is generally an analytical study of the line which itself has several inherent problems.

- Analytical results tend to give detailed exact results for individual cases, rather than a general close approximation for all cases.

- Due to the increasing complexity of the mathematics when dealing with longer lines, studies have been typically made of 3, 4 and 5 machine lines. Indeed, the 5 machine line studies have had to be simplified to allow the mathematics to be computed. Transfer lines are generally far longer than 5 machines, and some features seen on longer lines are not seen on shorter ones.

- There is a lack of reality in some analytical studies. Actual practices and effects on the line can be overlooked or avoided in order to simplify matters. Line length being a typical example. This lack of reality can also be explained by the fact that the research is carried out by academic statisticians, rather than production engineers who have more experience of industry.

These points, however, do not justify a total dismissal of the practical significance of all the work that has been done analytically. It is, however, important to realize the limitations of such work when examining it.

3.2 The bowl effect

The 'Bowl Phenomenon' has been referred to in the previous section. It is a prime example of the problems that can occur when lines are studied analytically. The basis of the bowl phenomenon is best summarized as "assigning lower average production times to the intermediate stations than to the stations on the two ends" [Ding and Greenberg, 1991].

Hillier and Bolring [1966] were the pioneers of this work. Their work was based on a three stage line with no breakdowns, but with unequal cycle times. It was found the line output of a balanced line could be increased if a faster machine was placed in the middle.

This work has been confirmed by Yamazaki & Sakashwara [1975] who added that the phenomenon only worked while there was no buffer capacity. The effect on throughput and the output levels for a 'bowl' line has been estimated by Muth and Alkaff [1967].

In a similar way to creating a bowl by concentrating the work load to the outer machines, Rao [1976] suggests you should place more work on stages with a smaller variance in cycle time. This should be done in parallel to getting a bowl effect, the proportion of work movement being dependant on the coefficient of variance. The ideal of this work would seem to be to have the machines in the middle of the line with an inconsistent cycle time, irrespective of its mean duration, compared with those at the ends of the line.

A further example of the study of process time variations is by Carnall and Wild [1976] who, using a Weibull distribution for variation in cycle time on a balanced line, showed that the capacity of the line is maximised when the most variable processes are put at the end, which conflicts with the findings of Rao described above.

Although the majority of the work in this area has been analytically based, the work has also been confirmed using simulations by Yamashina and Okamura [1983]. Hira and Pandey [1982,1983] used simulation to claim that a balanced line is best, but if there is to be imbalance then the bowl phenomenon is best.

Douglas Smith and Brambaugh [1977] present a further additional strategy to the bowl phenomenon. Having confirmed the existence of the bowl theory for unbuffered lines, both for work content and variance, they add that "reallocation of a given Work-In Progress inventory (buffers) capacity to the vicinity of stations with higher variances in process time..." also improves line performance. Thus they would buffer the middle machines on the ideal line of Rao, described above. In order to design the line with both the bowl theory in mind and using buffers, Ding and Greenberg [1991] claim 'optimal design may be obtained for several servers (machines) in series iteratively by first selecting an optimum order for the servers, then optimizing the buffer allocation, and repeating these steps as necessary'. They also claim that for a line of more than three machines, that the bowl shape is not necessarily the optimum way to allocate work to machines.

Here lies the great problem with the whole 'bowl phenomenon'. Some authors claim that the theory works for all lines, yet others claim that for over 3 machines the theory does not necessarily work. Some authors claim that the theory only works with unbuffered lines, others claim that buffering can be used to improve output on a line with a bowl shaped work distribution. Rao claims large improvements in output by arranging the line this way, but the most recent work by Hillier (one of the originators of the idea) and So [1993] claims that "The improvement in throughput provided by the bowl phenomenon is quite small, only of the order of magnitude of 1%".

Over 25 papers [Muth and Alkaff, 1967] have been published since the bowl phenomenon was first identified by Hillier and Bolring in 1966, and this research is one of the most prominent areas of transfer line study. Yet none have yet produced a definitive set of rules on how the line should be configured, more importantly, the whole idea seems to have lost touch with reality. Why should you be able to put the least reliable machines at the beginning of a line? How can you move the allocation of work from one machine to another? When a line is designed, the number of machines is determined from the required cycle time. This in turn allows the production process plan to allocate the operations to machines. Individual operations are done at individual stations in a strict order and, in the vast majority of cases, an operation cannot be moved to the beginning of the line for the sake of increasing output. It does not take a great deal of imagination to picture the scenario of trying to tap a hole before it is drilled!

Hiller and So's recent work [1993] probably best sums things up. "The optimum allocation of work under the bowl phenomenon is a target that usually cannot be achieved in practice, because of the necessity of assigning discrete micro elements of work to stations on the line". Having questioned the use of the phenomenon at all in light of the small improvements gains they add "It is better to aim at the correct target so that the inevitable deviations from a perfectly balanced line will be in advantageous directions that increase rather than decrease the throughput".

This seems to be the only worthy conclusion, if when building a line, the process plan is such that there has to be imbalance in the cycle time then aim to put the extra work load at the ends. This, however, fails to allow for any imbalance in breakdown frequency and repair time.

3.3 The provision of internal storage

As stated earlier, the provision of internal storage has been one of, if not the, main area for transfer line research. When reading published works, there is no doubt that the provision of internal storage results in an increase in output. The amount of improvement, the amount of buffering to use and of course where to put the buffering on the line still, however, remain the subject of much debate, with no definitive rules existing.

3.3.1 The effect of inventory

All the authors in this section refer to storage as a method of increasing output. As such there is little point in referencing this to them all. Indeed it would be of more interest to find an author who disagrees with this idea! Unfortunately this is not the case.

As mentioned earlier, Buzzacott was a pioneer in transfer line research, and is by far the most frequently referenced author. There are two main papers by Buzzacott which deal with the provision of internal storage on lines. The more recent [1971] discusses the role of buffers under a variety of line conditions without using any values or formulae. In this work he describes how buffers that never change their level are of no use. Although this may seem an obvious statement, the implication is that the effectiveness of a buffer can be measured in terms of variations in its contents. He also states that "a particular inventory bank has no value if the supply from it is less than the supply to it as it will always be full". This statement is surprising since it implies that buffers are only of use on a line which is not only balanced in terms of cycle time, but also in terms of the time spent broken down and that the breakdowns occur simultaneously. As such it should therefore not be possible to buffer a bottleneck. This contradicts with nearly all other work done on how to buffer lines (see Section 3.3.3).

Buzzacott's earlier paper [1968] is concerned with calculating the efficiency of 2 and 3 stage lines with interstage storage, where efficiency is defined as the proportion of time a machine spends operating compared with total time. A lower bound solution is generated for a line with no buffers, where the output is denoted as E_o. By placing infinite buffers on the line, the stoppage of one machine affects no others, and as such the maximum output is achieved. This is the upper bound solution with an efficiency E_z. It is claimed that the efficiency of any line with buffer storage must lie in between the two boundaries. Thus the actual efficiency E lies in the range

$$E_O < E < E_Z$$

By defining a measure of the effectiveness of the buffers g, as the ratio of the gain in production achieved through using buffers compared with the line without buffers.

$$g = \frac{(E - E_o)}{(1 - E_o)}$$

It is then stated:

> "In general *g* is determined solely by the way the line is divided into stages by the buffer, and by the buffer capacities. Thus, if with a certain division and buffer capacities, *g* is 50 per cent, then, if E_0 were 70 per cent, E would be 85 per cent, while, if E_0 were 90 per cent, E would be 95 per cent."

This is an example of the law of diminishing returns. The more efficient the transfer line, the less effect adding buffers will make. A further implication of this is that having placed buffers on a line, adding a further buffer of the same size will not have as great an improvement on the line's efficiency.

3.3.2 Output and its variation with different parameters

The effect of line length on buffered lines with stoppages was studied analytically by Hatcher [1969]. He states that "the greater the number of stages in a line, the less the reduction in output by adding another stage. Apparently disturbances occurring within a line tend to 'damp out' as they travel through the line. Consequently, the net disturbance created by adding a new stage varies inversely with the number of stages already on the line." Yamashina and Okamura [1983] add "for a multi stage line the number of stages and buffer storage capacity between the stages are critical design factors strongly influenced by the production rate of the line. As the number of stages increases, it is of vital importance to install buffer stocks in order to compensate for the decrease in production rate due to linking new stages to the line."

A study of the effects of line length and other parameters was conducted by Magazine and Silver [1978]. They developed an analytical set of heuristics to determine output for a given set of parameters on a line of up to 6 machines. As with other analytical work, further study was limited by the complexity of the mathematics involved.

Estimations of the output of longer lines has been carried out by Murphy [1978]. The output of an N machine line is derived by reducing the line into a series of machine-buffer-machine problems. Questions must, however, be asked about some of the underlying assumptions. The work assumes that if there are no buffers between machines the line is synchronous. Thus if on a 20 machine line the second machine stops, all downstream machines must stop. This is different to the typical non-synchronous control (see Section 3.4) usually associated with longer lines, where the machines downstream would continue to function even if this resulted in emptying the line.

An interesting point concerning breakdowns with buffers is described by Hillier and So [1991]. Although the work was analytically based on 4, 5 and very simplified 6 stage lines, they found that for a given %downtime it was advantageous in terms of output to have many short stoppages rather than fewer longer stoppages.

3.3.3 Rules for allocating the storage

Douglas Smith and Brambaugh [1977] have shown "that adaptive procedures such as the reallocation of inventory capacity to achieve equal utilization of storage banks or equal utilization at the work stations can actually cause a deterioration in line performance". It is therefore important to analyze the whole line and maximise total output rather than concentrate on any one area.

Buzzacott [1967] claims that the line can be separated at any point provided that the bowl effect of machine reliability described in Section 3.2 is maintained. This contrasts with the results of Conway et al [1988] who have found that output decreases with length in such a way that it is far better, for example, to break a 6 station line into two 3 station lines than into a 1 station line and a 5 station line.

Conway's work provides the most complete set of 'rules' on the buffering of lines. They are as follows:

"1. In a line of identical stations, the best buffer allocation is symmetrical if possible.

2. The best buffer pattern has a slightly greater capacity in the centre.

3. The correct allocation can be as important as total buffer capacity.

4. The same production capacity is achieved with a mirror image(i.e. 4-4-3 and 3-4-4) even though the WIP distribution is changed."

these rules are followed by two others

"A. Buffers provide less of an increase in an unbalanced line, and the preferable position for the buffers is displaced towards the bottleneck workstations".

B. Buffer capacity should be in multiples of the number of components produced by another station while one station is being repaired. The size of the multiple depends on the degree of variation of the repair time".

The implications of the two latter rules do, however, raise some questions. Firstly, do buffers provide less increase in output on a balanced line? This suggests the bigger the bottleneck, the less effective the buffering is. This seems to go against ideas based on the TOC (see Section 3.4), and is probably dependent on the way buffer effectiveness is measured. The implication of the second point is that the minimum buffering level between machines is equal to the number of components produced during the time taken to repair a machine. Surely any buffering is better than none? If the buffers are smaller than the equivalent repair time then the effect of a stoppage may have to be spread across several buffers as would surely happen in the case of bigger buffers if they are already full. The idea of the effect of a breakdown spreading across several buffers is further explained in Chapter 7.

Buzzacott has also stated that the capacity or size of the buffer should be in multiples of the mean time to repair the machines on the line. He also claims that buffer capacity at any one point should never be more than 5 times the mean repair time, since it would not be possible to justify the cost involved against the diminishing improvement in output. "Provided the capacity at each point is greater than the mean repair time, the benefit of additional storage decreases. It is unlikely that a capacity of greater than 5 times the mean repair time can be justified unless the additional cost is negligible". These ideas together imply that there are only 6 possible buffer sizes between two machines. No buffers, 1, 2, 3, 4 or 5 times the mean repair time.

Hatcher [1969] disagrees with Conway's first two rules. He claims that for a balanced line of 3 machines with cycle times varying exponentially around a mean, that buffering should be even, and any extra capacity should be placed at the ends, rather than in the middle. Following their simulation study, Yamashina and Okamura [1983] state that for lines of more than 4 machines which have sufficient buffer capacity, central buffers are given the highest allocation and nearly equal but diminishing allocations are successively made to the outer buffers. This is referred to as an inverse bowl or triangular allocation pattern. It is claimed that this pattern is the most effective on balanced lines.

As if there was not enough confusion already concentrating extra capacity at the ends is also advocated by Freeman [1964]. His rules of buffer allocation also include the following:

1. Avoid extreme allocations- lots between some machines, none between others. Even with a large difference between good and bad stations this allocation is poor.
2. The bigger the difference the more buffering should be allocated between them.
3. More should be allocated between a bad and a bad than a bad and a good. The worse the two bads are, the larger the total allocation needed, then the poorer are the results of mis-allocation.
4. The optimum pattern is invariant to total capacity.
5. The end of the line is more critical than the beginning.

For a bottlenecked line these rules seem to contradict each other. Rule 2 says you should add more around the bottleneck, the bigger the bottleneck the greater the concentration around it (most authors seem to agree on this point). Rule 1, however, states that extreme buffer allocations (i.e around a big bottleneck) should be avoided.

Rule 5 directly contradicts the generally accepted work of Yamashina and Sakashwara [1975] who proved for short lines that a balanced line is reversible. If buffering is to be concentrated at one end rather than the other, then is it more beneficial to have it at the beginning? The reason why this may be true is that the speed of stoppage effects passing along the line is not uniform. Those passing upstream, where there are no buffers, travel instantly (all machines become instantly blocked). Those travelling downstream move at the same rate as components. Thus to reduce the effects reaching the end of the line buffers should be placed at the beginning rather than at the end. This idea is further discussed in Chapter 7.

Further rules have also been presented by Yamashina and Okamura [1983]. They claim that buffers should be allocated so that :

1. The difference between production rates on either side of the buffer is minimised.
2. The production rate of the stage before is greater than the stage after the buffer.
3. Uniform buffering is not optimum, even for a balanced line (but for a balanced line it is very close).

The second of these rules seems to be half way towards the bowl effect. It goes against the reversibility rule of Yamasaki and Sakashwara. Yet elsewhere in their work Yamashina and Okamura claim that buffers should be placed in an inverse bowl allocation pattern.

Jafari and Shanthikumar [1989] have used a mathematical approach to establish the optimal buffer storage capacity on a line. Although this work seems to be well directed towards the problem of buffer design, it suffers, like most analytical studies, by only being able to deal with lines of up to 4 machines.

Other research has been carried out into buffering patterns on lines, but it has not resulted in the kind of implicit rules described above. Ho et al [1979] has analyzed the variation of stock levels in very large buffers to determine the optimum level. The idea behind this is that any buffer stock that is not used is a waste. For example, if you have a buffer with a maximum capacity of 100 components, and its capacity is found to vary between 50 and 75 components, then there is no need to have a buffer with a capacity of more than 25 components.

A different approach was used by Yang, Chen, Chang, and Wang [1983]. They used analytical, simulation and control theory to study a line. The line was reduced into a series of machine-buffer-machine problems in a similar way to Murphy. The limitations of this work was that it was carried out on synchronous lines and there was no allowance for blocking and starving by other machines outside the machine-buffer-machine link. The result was a very simple solution as stoppage effects could not travel from one buffer to another.

Complex analytical composition and decomposition techniques were used by Sheskin [1976] to produce an algorithm that is said to work for longer lines. The mathematics is highly complex and is also beyond the average production engineer designing a line. The paper claims that the algorithm provides guidelines as to an approximation of how to buffer a line. Examination of the paper yields little in the way of a useable technique or methodology.

3.3.4 The effect of Cost

Since the maximum output can be achieved by the use of infinite buffers, Ho [1979] claims that the production of an optimum buffer strategy is a function of cost. If buffers had no cost associated with them then the buffering of lines would be easy. The real problem, it is claimed, is to achieve the required level of output using the minimum cost.

A formula for the optimum buffering level has been developed by Anderson and Moodie [1969]. It is based on the even buffering of a balanced line of up to 5 stations with no breakdowns.

$$X = p\sqrt{y - b}$$

where

$$p = \sqrt{\frac{a}{1.45N - 1.32}} \qquad y = \frac{C_3}{C}$$

X = Total Economic buffer capacity for the line

N = Number of machines

C_3 = Delay cost of sales per unit time

C = Effective cost of inventory per unit time

and $a = 0.22 - 0.22N^{-0.76}$ $b = 0.81 - 0.28N^{-0.51}$

Their model nevertheless fails to include any initial capital cost of installing the buffering, nor does it include any payback period.

The cost of buffering is not linear. The cost of putting given buffer capacity at one point on the line is less than putting half that capacity at two points. Buzzacott claims this is the reason why lines should never be separated into more than 5 stages (4 buffers). The reality may well be, however, that the mathematics behind his 5 stage line was too complex to develop for 6 stages as other researchers have also found since. The lack of linearity of buffering cost has also been studied by Hopp, Pati and Jones [1989]. Their work is, however, based around the costing of continuous flow lines (such as steel mills etc).

3.4 The Theory of Constraints

The application of the Theory of Constraints (TOC) to transfer lines is based not on the time machines spend operating or broken down but on the amount of time each machine spends not operating when it could be. The application of the TOC to a batch environment was explained in 'The Goal' by Goldratt & Cox [1984 & Supplement 1991] which was presented in the form of a novel. This book together with its follow up 'The Race' [Goldratt & Fox 1986] which is a more classic reference book outline a series of ideas on how production processes and other systems can be organised. TOC is not a rigid algorithm or method, but is a philosophy in a similar way to JIT.

The core idea is surprisingly simple. No system can operate at a rate quicker than its slowest element. This slowest link is said to be the bottleneck. In order to improve the output, the output of the bottleneck must be improved. Making improvements at the bottleneck can, however, result in secondary bottlenecks. The solution to the problem then lies in identifying the main and subsequent secondary bottlenecks (which are collectively known as Capacity Constraint Resources - CCRs) and making improvements at each one. By eliminating waste at the CCRs the manufacturing process is said to become more synchronized. In the case of transfer lines this improvement can be made by the addition of buffers around a machine. Thus to improve output of the line the buffering needs to be concentrated around bottlenecks.

Although Ho [1979] did describe how "Inefficiency in production is caused by force downs (stoppages) of otherwise healthy machines". The only paper found to really use the TOC to help in improving transfer line output was that of Raban and Nagel [1991]. Their work was concerned with the control of flexible flow lines rather than with the lines design. The line they modelled does not, however, operate non-synchronously and the resulting algorithm to determine the size of the buffer stock to protect the limiting resource is oversimplified. Nevertheless they suggest the pursuit of a method to allocate buffers to a line using the TOC would be a valid and useful development.

3.5 Decreasing throughput time

The work described above has all been concerned with either maximising line output at any cost or with ensuring the production costs are minimised by comparing the extra benefits of adding buffers with the extra cost of doing so. Some work has, however been done on minimising the throughput time.

Tcha, Lee and Yamazaki [1992] concluded that if you increase production by adding buffer stocks on the line, then you must also increase the average throughput time. "Improving performance on one (output) measure does not ensure an improvement on others" [Douglas Smith and Brambraugh, 1977]. Tcha, Lee and Yamazaki also suggest an inverse form of buffering in order to minimise throughput time where "the fastest servers are placed around the smallest buffered stages". This is again a case of ordering the machines by the buffers rather than the process plan. Their work also concentrated on verifying the 'Bowl' phenomenon and as such the same reservations described in Section 3.2 must be applied. Whitt [1985] has also conducted research into minimising throughput time (also described as Sojourn time or Flow time) and has determined a set of heuristics for the placement of workstations.

3.6 The shortcomings of the previous research

Throughout this chapter, the work of many authors using several different approaches to the problems of transfer line design has been discussed. Yet the main problem of how a line should be designed has not been presented. There are, of course, many elements to the problem. One of the most dramatic ways to improve output is to add buffering to the line, but even after all the work that has been done there are few indications as to how this can be effectively achieved. Most authors have drawn up the same or similar guide lines, but there are still some contradicting ideas and methods.

It is the author's opinion that if, for example, a transfer line costing up to £150m is being commissioned by a large motor manufacturer, ideas such as "put more buffering near the least reliable machine" are inadequate. Current line design techniques involve simulations of the proposed line. Many different configurations and layouts are tested against each other before the "optimum" is found and built. The line designer still, however, has to produce these configurations using experience of what they believe is best. This can narrow thinking and perpetuate bad practice that often results in the "we've always had a day's worth of components between the machining and assembly lines" attitude. The other problem is that if you must learn by your mistakes then it is necessary to have experience of building some bad lines before you can produce good ones.

The likely consequence of such a line design strategy is that senior management will see the line on which they have just spent millions of pounds "under producing". Instead of the line producing components at the expected rate for 80% of the time, the line will typically operate for less than 50% of the time. Thus the real financial payback period becomes much longer, factory output is restricted because of the shortage of parts, extra shifts are run and the operating costs increase significantly. The solution, of course, is to get the design right first time.

Chapter 4 - Simulation of Transfer Lines

4.1 General

Simulation can be described as "the technique of imitating the behaviour of a situation by using a model in order to gain information more conveniently. Such a model can be defined as a simplified or idealised description of the system and is devised to facilitate predictions and calculations" [Carrie, 1988]. It allows the study of any system without the need for that system to be built. In term of manufacturing plant, it allows the design of equipment to be optimised before it is installed, and thus can save costs associated with trial and error on the shop floor, so "minimising capital expenditure and risk whilst maximising the effectiveness and economic return of the system" [Chapman, 1993]. It should however, be noted that the use of simulation does not in itself give better designs, it is merely used to compare the merits of existing ideas; i.e. "the dynamic representation of a manufacturing facility by a computer model, so that the impacts of changes can be evaluated to support the decision making process" [Simulation Study Group, 1992].

The most simple model is in the form of an equation or series of equations where the input parameters of the model are used to define the result. Indeed, simple manufacturing environments have been modelled analytically. The majority of this work has been on transfer lines (*flow line* technology) as this is an easier form of manufacturing plant to model than that of a typical *batch* environment. Even so, as the lines become more complex, with more parameters, to analyze them mathematically becomes virtually impossible. The maximum number of machines that can be realistically modelled mathematically is 4 or 5 (see Chapter 3). These more complex manufacturing environments can, however, be modelled using computer simulation packages as an alternative.

4.2 Computer Simulation

Computer simulation of manufacturing plant has been around for quite some time, one of the earliest forms being IBM's General Purpose Simulation System (GPSS) [IBM] which was designed for use on mini-computers and required a high level of programming skill [Schriber, 1974]. A number of other simulation packages have since been released and in recent times there has been a great increase in the use of simulation as a tool within industry in the UK [compare DTI Survey 1992, Christy 1983]. It may be stated that this increase in use is a result of the need of industry to be globally competitive and this demand has created the supply of simulation packages. It, however, may be conversely argued that the increase in use has been due to the marketing skills of the simulation package producers who have produced better and more easily used simulation packages as computer power has increased. The truth probably lies in a combination of them both, but it has resulted in two types of computer simulation. Firstly there are simulation languages which are flexible and require a higher level of programming skill, and secondly there are manufacturing simulators which tend to be more user friendly, with graphic interfaces and are only suited to the manufacturing environment [Law, 1986].

Which ever is the reason for their existence, there are now a number of packages to run on PCs commercially available. These include XCELL, Siman, ProModel, PCModel, WITNESS and Hocus [Packages referenced under name]. These are all manufacturing simulators, with the exception of Hocus and PCModel, but although these are both langauge based, they do have interfaces with a manufacturing bias. At the present time, WITNESS and ProModel are the market leaders, with the cost of both packages in the region of £20,000.

All these packages are described as *Discrete Event Based* packages, that is they work by creating a list of future events, carrying out those events at a given time, and using the results of the event to produce more events in the future. For example, a component on a machine will have a finish time in the future event list. When the time for the component finish is reached, in one element of time the component will be sent on the next part of its route, various counts will be incremented and the machine becomes available to process a new component (resulting in a new finish time being placed somewhere in the future event list) if one is available. It must, however, be noted that some of the packages (e.g. WITNESS Version 7) have the capacity to deal with continuous events such as fluid flows in pipes resulting in changing volumes in tanks etc.

4.3 The WITNESS Simulation package

Compared with batch, cellular or assembly environments, the computer simulation of transfer lines is comparatively easy, since there is normally only one component type and each one must go through the line in the same order. This logical machine order for the model allows quick and easy modelling of lines in most packages. It is, however, important when conducting a project which will involve a considerable number of different 'runs' to select a package that allows input parameters to be changed easily and quickly, that provides good report facilities, and has an acceptably quick simulation speed.

When this research work commenced there were some links with Ford [Ford GB]. They had, as part of their support of some undergraduate research into simulation techniques, funded the purchase of the WITNESS simulation package. Since WITNESS met all the requirements of a package to use while conducting the research envisaged and there were possibilities of continuing links with Ford, the WITNESS package was selected for this study.

Although there have been several updates and enhancements, this research work has all been carried out using Istel WITNESS version 5.0, which was first released in 1989. WITNESS itself is not a simulation package. SEEWHY is a discrete event based simulation language written by Istel, and WITNESS is a user interface for SEEWHY. Data on the system to be modelled is entered through WITNESS which then generates the program code for SEEWHY and automatically runs it without the need for user intervention. The WITNESS/SEEWHY interface is not a closed system. When complex logic and file handling is required, which is beyond the capabilities of WITNESS, additional subroutines may be written in FORTRAN 77 [Fortran]. It was not, however, necessary to use this feature during the course of this study.

The manual [Witness, 1989] describes WITNESS as "a graphic interactive simulation tool with artificial intelligence features which enable the non-simulation specialist to quickly build models of complex operations. WITNESS combines the power of moving colour graphics with user interactions to permit a decision maker or planning team to view a complex factory operation", and then further describes itself "as having :-

- User friendly terminology
- Animated and integrated moving colour graphics
- Totally interactive and interpretive
- PC compatible
- Full simulation capabilities

During the course of this research it has been found that:

- The models are easily constructed
- Machine and buffer parameters are easy to change
- The screen output can be easily switched off to allow far quicker running
- Simulation configurations normally entered in the package can be prepared in a word processor to allow several simulations to be left running overnight/over weekends. This is particularly useful when several long simulations need to be run in parallel to other work and you have access to more than one PC.

This must not, however, be allowed to gloss over some of the packages faults. In the author's opinion the reporting facility is just adequate, with little useful data regarding buffer levels available. The output format of the report facility means that in order to generate hard copy graphic output, it is necessary to use 3 other software packages.

The manual is very poor indeed, even with simulation experience it would not be possible to conduct a complex simulation using just the manual as guidance. Finally the file handling is very basic. For instance, it does not allow directory listings of files once inside the package. It must, however, be noted that many of these 'faults' are not present in more recent releases.

Although at the start of the project WITNESS was the second most used package in education [DTI survey, 1992] (the most commonly used was an outdated version of SIMAN), during the course of the work there has been an increase in the use of ProModel, both nationally and within the Mechanical Engineering Department at the University of Bath. At a midway stage in the project, simulation run-tine was considered to be holding up the progress of the work. A comparison was made between the run-times of the two packages to evaluate them. Although ProModel was more user friendly (current versions of WITNESS are similarly so), the run times were almost exactly the same. Since the modelling techniques and ability to process the results for WITNESS were firmly in place, there was deemed to be no advantage through changing packages at the midpoint. Had ProModel, however, been available from the start then it would have been preferred.

4.5 Model description

In order to evaluate different configurations and strategies many different 'runs' are needed. With this in mind a flexible model was constructed so that the line parameters could be varied using the same model.

4.5.1 Modelling techniques and assumptions

The base model was built with a maximum length of 35 machines. In order to use shorter lines it is possible to define the Nth machine to be the final machine in the line. Figure 4.1 shows a schematic of how this is done for a 30 machine line.

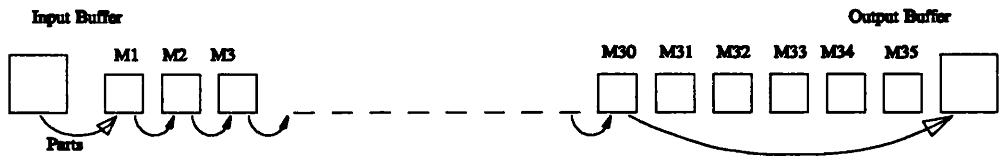


Fig. 4.1 Schematic of transfer line model used. An example of a line 30 machines long

The line is modelled as a *Non-Synchronous* line. On a non-synchronous line, when one machine stops downstream machines can continue cycling. It is also arranged for convenience that the repair time of the machine is a multiple of the cycle time. This ensures that once a machine is repaired it will cycle at the same time as other machines on the line.

Parts are fed to and from the line from an imaginary world. The imaginary world ensures that the input buffer is always full and the output buffer is always empty. This is so the line being examined is isolated from any effects outside the line. Buffers on the line itself can be placed as input buffers for individual machines. Thus, if a buffer between machines 15 and 16 is required, an input buffer is placed on machine 16, not an output buffer on machine 15.

It is assumed that there are always operators available and that there is always sufficient repair labour to repair machines in parallel should more than one machine be broken down at the same time. More than one machine can be broken down at the same time because machines downstream of a breakdown can still be operating even without buffering. It is assumed machines can only breakdown whilst producing.

4.5.2 Input and output parameters

Appendix I lists some of the parameters and measures of a line that can influence the performance of a transfer line. Not all these can be input directly into the model as they are a combination of other inputs (e.g. level of balance is governed by each parameter affecting all the machines). Definitions of the main parameters that can be varied in the model are as follows-

Number of machines	N	The length of line used.
Cycle Time	C	The time taken for a machine to complete its cycle.
Breakdown Frequency	B	Although termed a "frequency" the actual unit of measure is the number of components processed by a machine between breakdowns
Repair Time	R	The time a machine spends idle due to a breakdown.
Buffer Capacity	BC	The maximum number of components that can be held in a buffer.

The output of the line (its efficiency) is measured by considering the percentage of the total time that the last machine on the line spends busy.

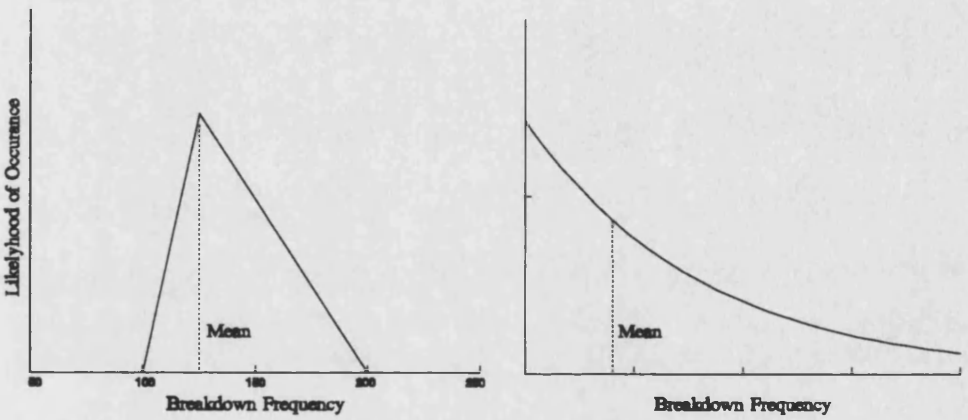
$$\textit{Output Efficiency}, O = \frac{\textit{Actual time spent busy}}{\textit{Total time}} = \frac{\textit{Cycle time x Units produced}}{\textit{Total time}}$$

It is also possible, using the report facility, to determine some of the other output measures listed in Appendix I. The output efficiency has, however, been used as the main measure throughout this project.

4.5.3 Breakdown and repair distributions

Breakdown distributions can be sampled in three ways. Breakdown occurrences can be assessed in terms of *actual time, machine busy time* or *number of components* processed. Sampling distributions in actual time can give false results as a machine that is broken down can breakdown again even though it is not working. Using machine busy time or the number of components avoids this problem. The stoppages described in this report have been assessed in terms of the number of components processed.

At the early stages of this work, the Strian distribution (see Figure 4.2) was used for the breakdown frequency as it gives a more regular breakdown pattern than an exponential distribution, without the breakdowns being of a fixed frequency. Although this is not the most accurate representation of an actual line, it does allow easier understanding of the events occurring on the line.



a) Strian distribution b) Negative Exponential Distribution

Fig. 4.2. Machine breakdown distributions used

The Strian distribution is triangular in shape. An example is shown in Figure 4.2a where 'STRIAN (100,125,200)' has a minimum number of components between failure of 100, a maximum of 200, with 125 being the most likely (the peak of the triangle). In a further effort to keep the initial model simple, the repair time was given a fixed value and not sampled from a distribution. Again this was unrealistic, but it allowed the effects of starvation windows and blocking patterns to be studied.

Having gained an understanding of what is occurring within the line, more realistic distributions were brought in. The breakdown frequency was sampled from a Negative Exponential distribution (see Figure 4.2b) and the repair time was sampled from an Erlang distribution with a K value of 2. The breakdown frequency is sampled from a negative exponential distribution because this assumes that the likelihood of failure is constant in time. Using an Erlang K2 distribution for repair time is the same as sampling from an exponential distribution and adding the results together representing the time to detect a failure, followed by the time to repair it [Witness Modelling Notes, 1989]. The use of these distributions is supported by the work of Crosby and Murton [1990] that was carried out on one years breakdown information for Ford's Fiesta engine plant in Valencia.

4.6 Accuracy of results

The simulation of transfer lines is non-terminating (assuming there are an infinite number of parts to be processed). As such it is said that a measure of performance of such a system can be said to be a steady state parameter. Nevertheless, determining the point at which this steady state is achieved and then choosing a suitable period over which to monitor the steady state in order to achieve consistent results has been the subject of much conjecture and study.

4.6.1 'Run-in' time

At the start of a simulation of a transfer line, the whole line is empty. It is therefore necessary to allow the line to 'run' without taking results, during what is termed the *transient period* until *steady state* conditions are reached. There are no definitive rules for how long a model should run before the steady state is reached [Taka, 1988]. Heuristics available are based on the idea that if the output ceases to exhibit excessive variations after a given period then the steady state has been reached. The variation in output will never actually cease, it is simply up to the user to determine their own confidence levels. The number of components required to reach steady state increases as more buffers are placed on the line, but for a 30 machine line, an output of 3000 units was used as the start point for taking results. The output per period after 3000 units is within 0.5% of the figures achieved at the end of a run of 100,000 units.

4.6.2 Run lengths and random number streams

The results of simulation runs of different lengths and/or using different random number streams, even with the same input parameter, are rarely, if ever, identical. The two main ways to ensure accuracy in results are either to take the results of several runs using different random number streams and calculate a mean or to ensure that the simulation run lengths are such that the difference between runs is small.

Figure 4.3 shows the difference in the accuracy of results between a run of 3000 cycle times and 100,000 cycle times, using the same input parameters. At 3000 units the machines all have the same output, but there is great variation in the time spent blocked and starved. At 100,000 components the resulting graph is smooth and repeatable. The repeatability of results for 100,000 components with different random number streams is shown in Figure 4.4. There are still local variations (less than 0.5%) in the blocking and starvation curves but these are deemed to be within an acceptable tolerance.

In order to reduce the run times of the simulations, some initial study work was done with short runs of 5000 cycles following the period for the line to stabilise. Results of this work are shown as '*Short Run Length*' throughout this report. All other simulation runs were normally for 100,000 cycles.

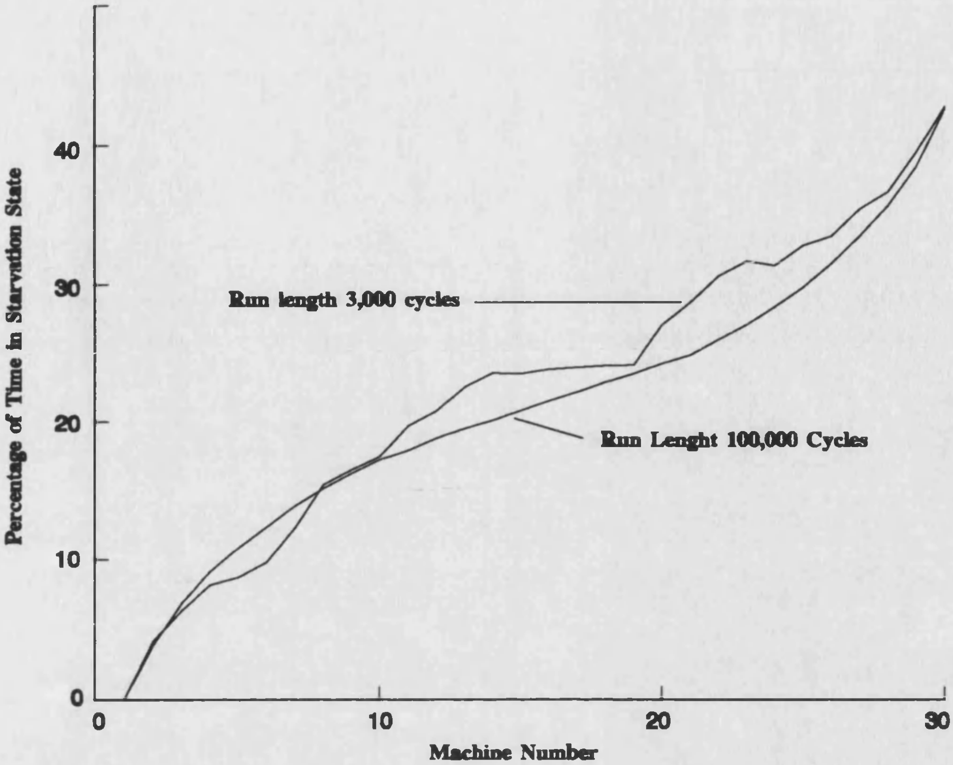


Fig. 4.3 The effect of run length on the accuracy of simulation results

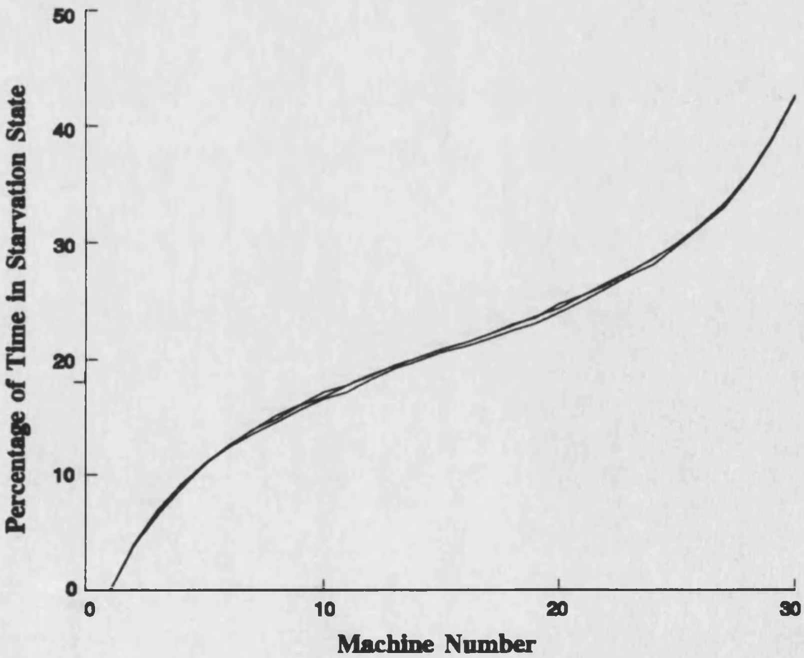


Fig. 4.4 The effect of different random number streams on simulation runs of 100,000 cycles

Chapter 5 - Machine Stoppages

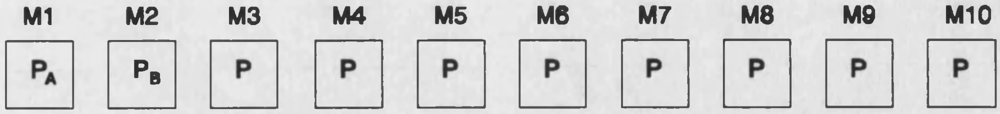
5.1 Why machines stop

It can be said that certain factors that cause lines to stop are organizational. These include manning levels, provision of raw material and tools and shift patterns. All these aspects together with others, such as power failure, are defined as external influences. In the same way that it is not the intended purpose of this research to examine the reliability of the processes themselves, it is not the intended purpose of this research to examine ways of dealing with shortcomings in line performance caused by external influences.

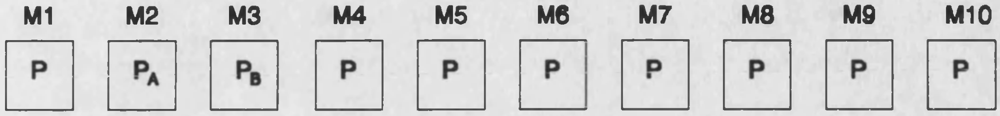
What is of greater concern is internal stoppages caused by machine failure and for tool changes. For it is these stoppages that occur most frequently and cause the greatest disruption. Data from existing lines [Crosby and Murton, 1990] suggests that an individual machine typically spends approximately 10% of its time stopped due to internal problems. This does not, however, include short stoppages of up to 3 minutes which are rectified locally. Overall line uptime for longer lines can be as low as 40%, with breakdowns on individual machines being a maximum of 10%. Although some of this lost time can be accounted for by shorter unrecorded stoppages, the majority is caused by the stoppages of one machine affecting otherwise "healthy" machines on the line. Descriptions of the distributions of the breakdown patterns and repair times used throughout this work can be found in Section 4.5.3.

5.2 What happens when a machine stops

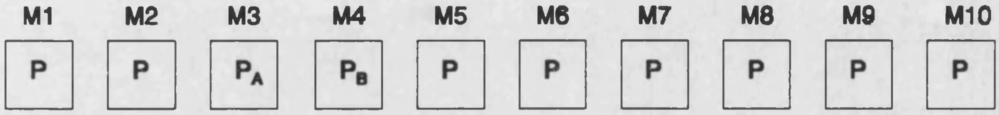
In order to understand how the line output is so drastically affected by what are outwardly minor breakdown levels, it is important to consider exactly what happens when a machine stops. Consider the example non-synchronous transfer line of 10 machines shown in Figure 5.1. When the line is functioning normally, parts are sequentially processed by M1 through to M10, each machine having a cycle time C.



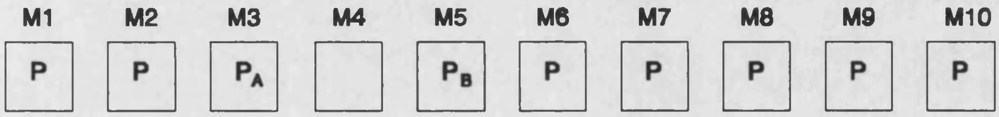
a) Time T=0



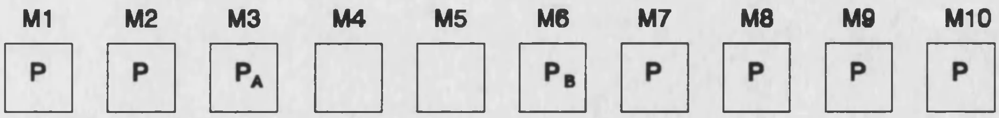
b) Time T=C



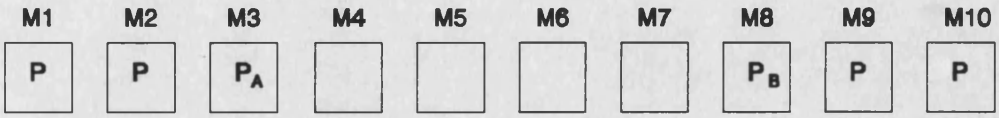
c) Time T=2C



d) Time T=3C - Machine 3 breaksdown

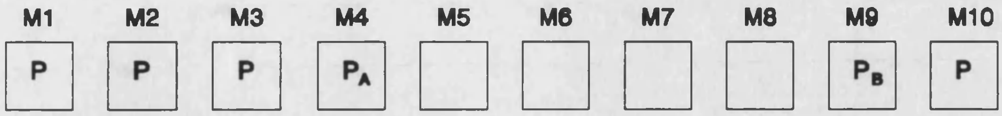


e) Time T-4C

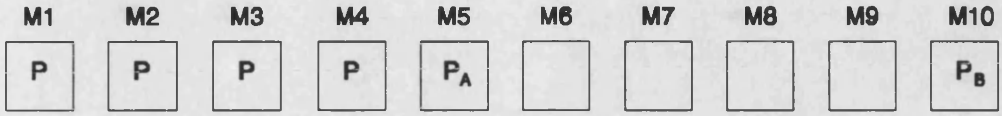


f) Time T=6C

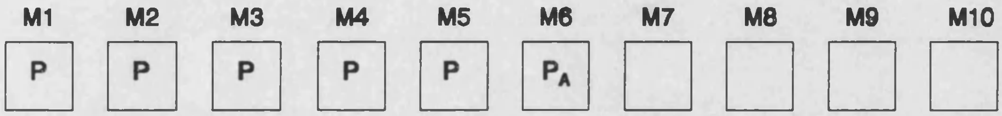
Fig 5.1 Flow of parts through an example line with a breakdown. (Continued overleaf)



g) Time T=7C M3 now repaired



h) Time T=8C



i) Time T=9C

Fig 5.1 (g-i) Flow of parts through an example line with a breakdown.

For the first three diagrams (a-c) the parts are flowing normally through the line. It can be clearly seen that the example parts (P_A and P_B) have moved two machines down the line. At the end of the cycle shown commencing in c) machine M3 breaks down. As a result, the part it is processing P_A can not be fed into machine M4. This effects the other machines on the line in 2 ways.

Upstream of M3, machines M1 and M2 are also unable to feed their parts as their output route is not free. As a result, although these machines are perfectly capable of producing components (i.e. they are not themselves broken down), they are forced to stop by the effects of other machines stopping on the line. Machines in this situation are described as being *Blocked.*

Downstream of M3, the supply of components to M4 has stopped. Like those upstream, this machine is also unable to cycle through no fault of its own. It is described as being *Starved.* Further downstream, machines that have components are able to process them and so continue to do so. The breakdown at M3 lasts for a total of 4 cycle times as shown in e) and f). During this time machines M5 to M7 also become starved as they run out of available components.

At the end of the cycle shown in f) machine M3 is repaired, having spent a total time of 4 cycle times stopped. M3 now feeds the part P_A to machine M4. The machines upstream are now no longer blocked and commence processing once more. They too have spent a time equal to 4 cycle times stopped, but in their case they have spent the time blocked.

As the line has continued to process parts, the gap that was produced between P_A and P_B remains at the size it was in f). The length of this gap (known as the *Starvation Window*) is also 4 cycle times, the length of the stoppage at M3. The starvation window travels along the line at the same rate as the components, as shown in g), h) and i). As the starvation window passes over a given machine, that machine is starved for a time equal to 4 cycle times, the length of the stoppage. When the starvation window reaches the end of the line, machine M10 becomes starved. Note this is at a time 6 cycle times after M3 broke down. It has taken this time for the starvation window to work its way down the line.

A different perspective of how a stoppage produces a window is to use Goldratt's [1984] example of a scout troop walking through a wood in single file. One scout stops (say to do up his laces) and this represents a breakdown. All the scouts behind him (upstream) are forced to stop and wait for him to tie his laces (he is thus 'repaired'). While he has been stopped, the front of the troop has carried on walking. A gap has appeared in the line which, once the lace is tied and the scout again walks at the same pace as the rest of the troop, now remains and continues at the same pace as the scouts. This gap is the same as the starvation window described above.

5.3 Window Interference

The above description assumes that only one breakdown has occurred. On a line, if each machine is down for 10% of the time and there are more than 10 machines on the line, the machines must be breaking down at the same time or at least be stopped in parallel. Blocking and starvation windows are produced by all the stoppages. The question is what happens when the effects of two stoppages travelling along the line in opposite directions meet (i.e. A blockage effect going up the line meeting a starvation window travelling down the line)? The phenomenon, known as *Window Interference*, has the effect of making the two stoppages cause the disruption of only a single stoppage. Thus output is improved by the fact that in terms of the interference of stoppages 1+1=1.

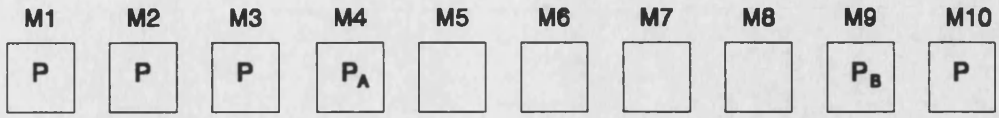
Window interference is best explained with reference to Figure 5.2.a) (the same as Figure 5.1.g)). As described in the example in Section 5.2, a starvation window between parts P_A and P_B is moving down the line. If instead of continuing to process as is shown in Figure 5.1.h) and i), machine M10 breaks down, the effect would be as is shown in Figure 5.2. Machine M9 now becomes blocked and part P_B is forced to wait until M10 is repaired but as the starvation window is resident from M5 to M8, M5 is not blocked and is thus affected by both stoppages at the same time. If the repair times of M3 and M10 are equal then by the time M10 is repaired, the starvation window travelling down the line will now have been filled and effectively disappeared, as is shown in Figure 5.2.b)-d). Parts P_A and P_B are now on adjacent machines as they were at the beginning of Figure 5.1 and move down the line together as shown in Figure 5.2.e). The end of the line has, of course, been affected by the stoppage of M10 and has been forced to stop for 4 cycle times. The time the line has been producing, however, has not altered from the scenario in Figure 5.1. The end of the line has still only been affected by the effects of one stoppage. To repeat from above - 1+1=1.

If the line were extended past M10, it is quite conceivable for this second stoppage effect to also interfere. This results in 1+1+1=1. The longer the line the more the chance of interference. When two stoppage effects of different size meet, they will obviously not cancel each other out. The larger one will be reduced by the smaller one. The net effect being that the small one will disappear.

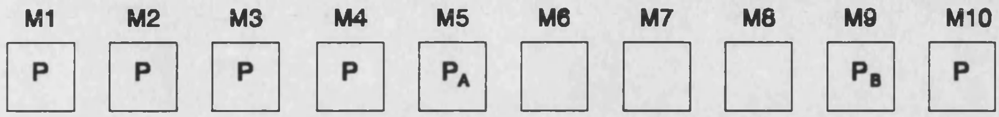
As with the single stoppage the effect of window interference can also be modelled on the scout group walking single file through the woods. The analogy went that a starvation window was formed towards the back of the troop when someone stopped to do up their laces. If now a second scout stops to tie up his laces, this time at the front, the gap at the back of the troop will close up. A second gap will appear at the front ahead of the second stoppage. Thus we have had 2 stoppages but only one gap in the troop.

As with the transfer line, the second stoppage must happen at the end of the line compared with the first. To be more precise, in the case of a transfer line the second stoppage must occur before the starvation window passes the machine which is about to breakdown. This demonstrates how important the timing of breakdowns is. Obviously it is impossible to get breakdowns to occur when we like, and thus maximise output. What is, however, possible is to time other stoppages (tool changes and maintenance for example) to interfere with the effects of a breakdown. An example of this is a maintenance window. Following a machine breakdown, the maintenance staff could travel along the line using the starvation window to carry out maintenance on subsequent machines. As the window moves down the line all machines can be dealt with. It is more difficult to carry this out upstream as all the machines must be dealt with in parallel and they are not empty of parts.

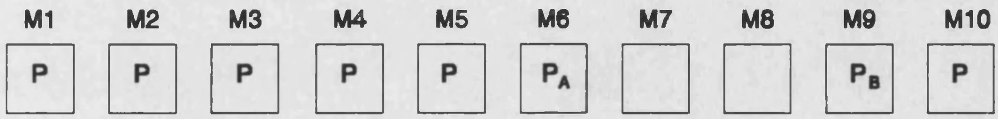
The idea of window interference has not been discussed by any other authors. It is difficult to understand why this is so as it is the only way to explain the difference between the theoretical and actual models which is described in Chapter 6.



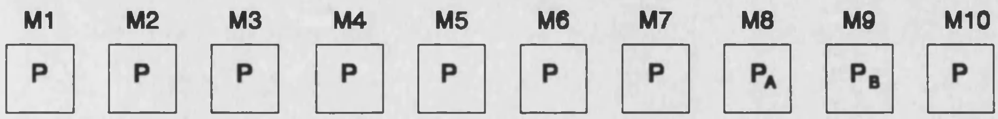
a) Time T=7C Machine M10 Breaks down



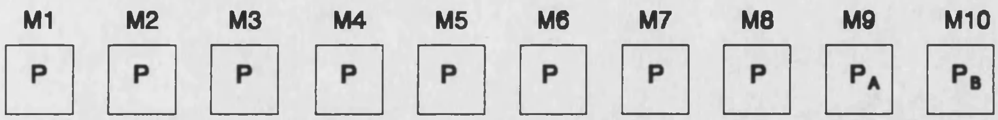
b) Time T=8C



c) Time T=9C



d) Time T=11C Machine M10 Repaired



e) Time T=12C

Fig. 5.2 Window Interference on an example line.

5.4 The effect of blocking and starvation

As has been mentioned earlier in this chapter, the effect of stoppages on the rest of the line can be very significant. Figure 5.3 shows a simplified breakdown of the time each machine spends Up, Down, Blocked and Starved for an example line. In this example, a 11% machine down time has resulted in an output of 40%. The remaining 50% of the time is spent effectively wasted. Curves showing the amount of time blocked and starved (and not necessarily showing up time and down time) will be frequently used throughout this thesis. It is, therefore important for the reader to clearly understand their meaning before proceeding.

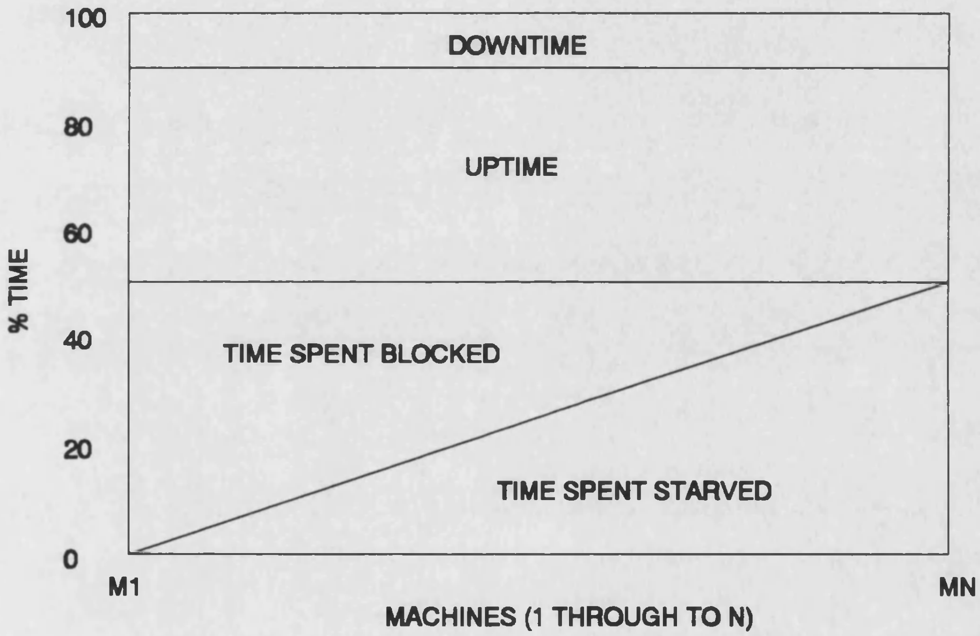


Fig. 5.3. Individual machine times for a line of N machines

5.5 Theoretical examination of the effects of blocking and starving.

Consider a fully balanced line of N machines (i.e. All machines have the same cycle time, breakdown frequency and repair time). On such a line the time each machine spends operating in a given period will be equal, as will the time each machine spends broken down. Assume also that Machine 1 is never starved of components and machine N is never blocked and that there is no window interference.

If the effects of the machines breaking down are studied during a time T which is equal to the time taken to produce a number of components equal to the breakdown frequency (measured in components). In terms of the starvation windows moving down the line, one at a time, it can be seen that a window from machine 1 causes all the other machines down stream to stop (due to starvation) for a time equal to the repair time. This is also true when machine 2 stops except that machine 1 is blocked and the stoppage at machine 2, therefore, does not contribute to the starvation of machine 1. This argument is applied to all the machines on the line. The result is that the starvation curve shown in Figure 5.3 is thus built up from the effects of these windows as shown in Figure 5.4. By examining these two graphs an estimate for the output of the line can be developed.

Consider the last machine on the line. MN will only be starved by the effects of other machines on the line (it can not be blocked). The time MN spends starved is equal to the number of starvation windows multiplied by their length (equal to the repair time R). Using symbols defined in Section 4.5.2.

$$\textit{Total Starvation Time} = R \times (N-1)$$

The last machine also breaks down for a time R. Thus

$$\textit{Total Non-operating time} = R \times (N-1) + R = R \times N$$

Total time T can be expressed as

$$\begin{aligned} T &= \textit{Producing time} + \textit{Stopped time} \\ &= (B \times C) + (R \times N) \end{aligned}$$

The time spent non-producing due to breakdown and starvation can be expressed as

$$\textit{Percentage non-busy time} = \frac{R \times N}{(R \times N) + (B \times C)}$$

Therefore

$$\textit{Theoretical output,} O(\%) = \left(1 - \frac{(R \times N)}{(R \times N) + (B \times C)}\right) \times 100$$

This derivation has been based around the starvation of the last machine. It can be easily repeated to give the same results for the blocking of the first machine (or the blocking and starving of any other machine) since all machines must process the same number of components in a given period (assuming the start and finish conditions are equal).

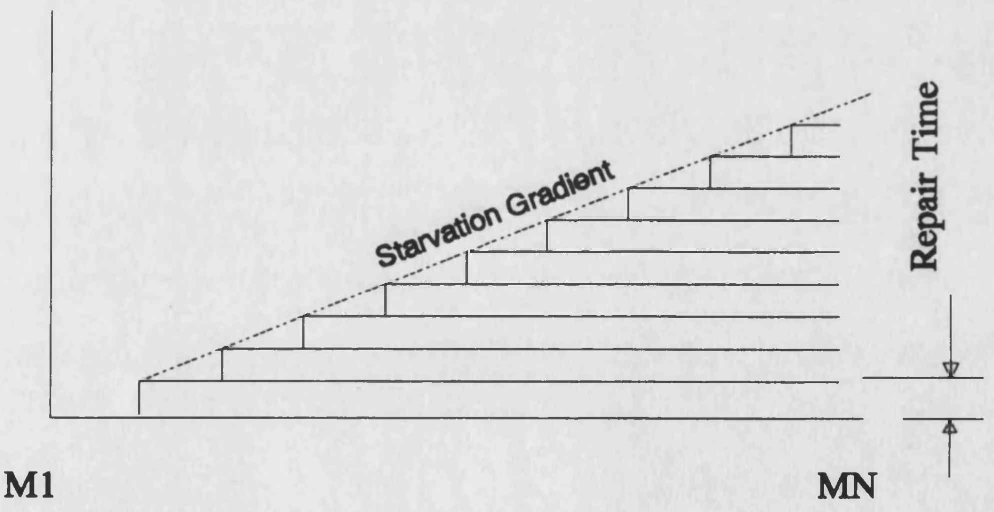


Fig. 5.4. Theoretical make up of the starvation curve

An additional consideration is the gradient of the starvation (and blocking) graph. This represents the additional starvation (or blocking) of a machine caused by stoppages of the previous machine.

The variation of output efficiency with line length is shown in Figure 5.5. The longer the line the less the effect of adding extra machines. This is, however, theoretical and assumes no window interference. A comparison between this theoretical model and actual results are presented in the next chapter.

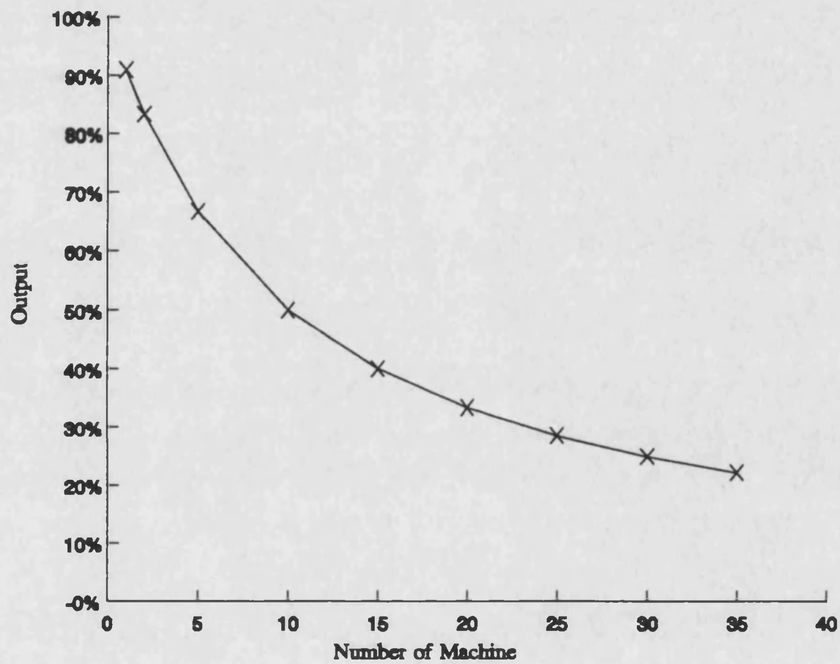


Fig 5.5. The theoretical output of a transfer line

5.6 Definition of %Downtime

Downtime is expressed throughout this thesis as a percentage of %uptime, *not* as a percentage of overall time. For example, consider a single machine with a %downtime of 10%. This is 10% of the %uptime. Thus

Total time = %uptime + %downtime = %uptime + 10% of the %uptime

Therefore the actual %uptime is 91% and the actual time spent down is 9% not 10%.

Chapter 6 The Effect of Stoppages on Unbuffered Lines.

Before studying the influence of buffering on lines, it is important to fully understand how stoppages affect the output of a line without any buffers. By doing this, the affects of different line parameters and details of how phenomenon such as window interference can be examined. There are two ways that understanding can be gained from simulations. Firstly by viewing the events as they occur on the computer screen, and secondly by analyzing the statistical results of various simulation runs.

In order to investigate the affect of parameters on line output a series of tests was carried out. The initial tests were carried out on fully balanced lines (all machines having equal breakdown patterns as well as cycle times). More complex lines were then examined by adding a bottleneck at various points on the line (the line was still, however, balanced in terms of cycle time).

Throughout this chapter, machines are referred to as having a given %downtime. The %downtime for a machine is defined as the proportion of time that the machine would spend broken down if it was a stand alone machine. When a given machine is placed in a line of machines, due to blocking and starvation the output will be reduced, and hence the actual amount of time spent broken down will be less than the %downtime.

6.1 The effect of length and other parameters on output

For an unbuffered line, the 4 main line parameters that can be varied are the number of machines or workstations (the length of the line), N; the Cycle Time, C; the Mean Breakdown Frequency, B; and the Repair Time, R.

To examine the effects of line length on output, a base model of a balanced line (both in terms of cycle time and machine reliability) was constructed where -

C - 1 min

R - 10 min (fixed value)

B - 100 cycles (mean value, Strian distribution)

N - varied from 1 to 35

A series of short simulation runs were carried out to establish the performance of unbuffered lines. These simulations were for 5000 cycles after the line had reached a steady state. As such the use of different random number streams for different runs was significant for these short tests and the results do not yield smooth curves. The significance of the run length can be seen in Figure 4.4. Although these results lack the accuracy of longer runs, they allow trends to be exposed without using excess computer time.

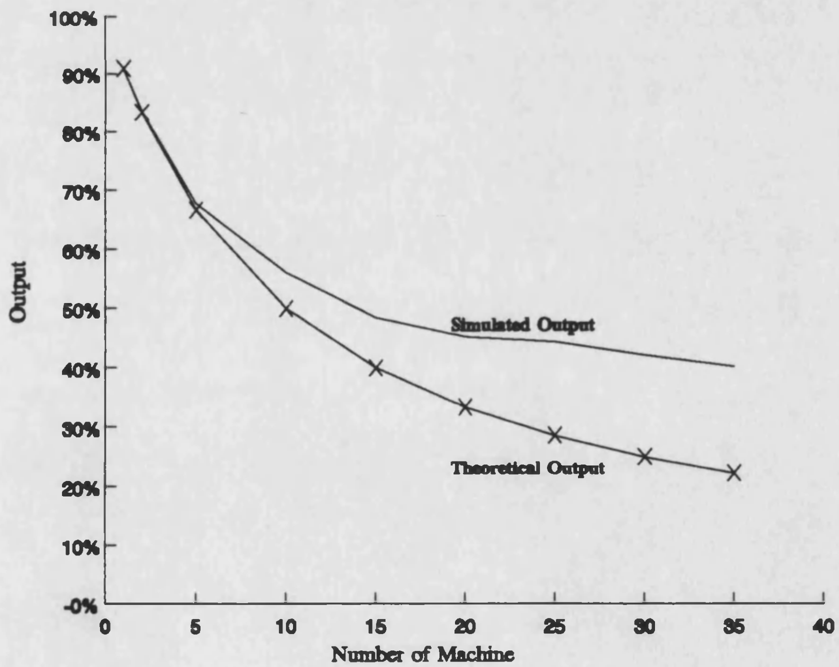


Fig. 6.1. Variation of actual and theoretical line output with line length

Figure 6.1 shows how the line output from the simulation runs varies with line length. Also shown, as a comparison, is the theoretical output derived in Section 5.5. Several important features can be noted from the graph:

- That the addition of extra machines causes a fall in output, but the effect of adding each machine diminishes the longer the line is. This point has been shown by many authors, see Chapter 4. The reason for the fall in output is that machines are affected by the stoppages of other machines. The more machines there are, the greater number of times that they are affected by blockages and starvation windows.

- That for a balanced line without buffering where each machine has a mean %downtime of 8%, then one machine will be up for approximately 93% of the time, a 5 machine line for approximately 68%, a 10 machine line for approximately 58%, a 20 machine line for approximately 45% and a 30 machine line for approximately 42%. Thus the results obtained, using short lines (less than 5 machines), as is the case for many previous researchers (see Chapter 3), will probably be different for those with 30 machines, but results achieved for 20 machines should still be reasonably valid for a 30 machine line. This again demonstrates the problems of analytical study of short lines, which can be overcome by the use of simulation.

- Output tends towards a minimum limiting value where adding extra machines will have almost no effect.

- There is a significant difference between the simulated graphs and the theoretical one. Although for short lines the values of the simulated output are nearly identical to the theoretical model, as the line gets longer, the difference between them increases significantly. This can be explained by window interference. If there is no interference then the output of an simulated line and that of the theoretical line will be identical. On a short line, few windows are produced in a given period. The windows that are produced quickly reach the end of the line and do not interfere with other effects. The result of little or no interference on short lines is that theoretical and simulated outputs are similar. As the lines get longer, there are more windows produced in a given period. These windows also take longer to reach the ends of the line. For both these reasons the chances of interference are increased. As interference levels increase so the theoretical model of the line where there is no interference becomes less valid, and the resulting difference between simulated and theoretical lines appears.

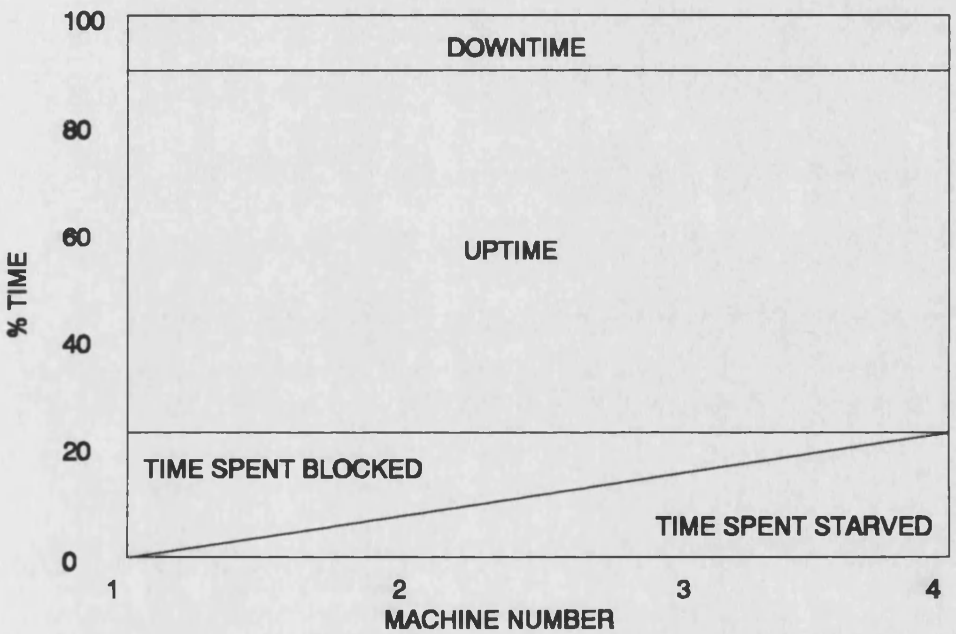


Fig 6.2a Time distribution of a line of 4 machines

This last observation can also be seen if we study the shape of the blockage and starvation curves for a 4 machine line and a 30 machine line as shown in Figure 6.2 (a and b respectively). On the 4 machine line, where little or no interference has occurred and the output is similar to that of the theoretical model, the graph appears similar to that of the theoretical model shown in Figure 5.3. (i.e. a straight line is present between the blocking and starvation area).

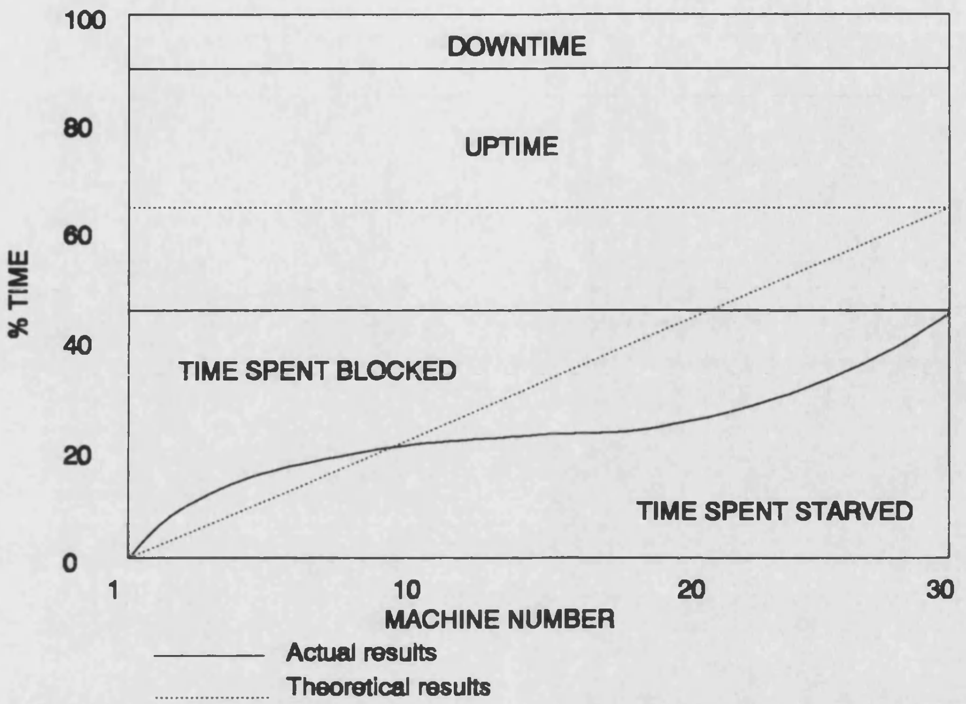


Fig 6.2b Time distribution for a 30 machine line

The gradient of this line in the theoretical model described in Section 5.5 was the additional starvation caused by a machine on the subsequent machine as a result of its stoppages. Linearity means that each machine has contributed the same extra starvation on subsequent machines all the way down the line (as would be expected on a fully balanced line).

On the longer line, Figure 6.2b, the graph is an 'S-curve'. The level portion in the middle indicates that subsequent machines have not been starved for any more of the time as a result of that machine stopping (i.e. the machine is not causing any additional starvation on the line). The reason is window interference between the two machines. Thus the window interference between the two machines is directly related to the difference between the simulated gradient of the starvation or blocking graph and the theoretical gradient calculated.

The curve is 'S' shaped as most of the interference occurs in the middle of the line, as this is where, statistically the probability of windows meeting is highest. The longer the line, then the greater the chances of windows meeting (and on a greater portion of the line) and thus the greater difference between the theoretical and simulated output levels, and the flatter the 'S-curve' obtained.

Having determined and explained the simulated output from the theoretical model, variations were then made in order to understand the effects of different parameters. C, R, and B were each varied in turn and the output efficiencies recorded for the range of line lengths.

6.1.1 Variations in Cycle Time (C)

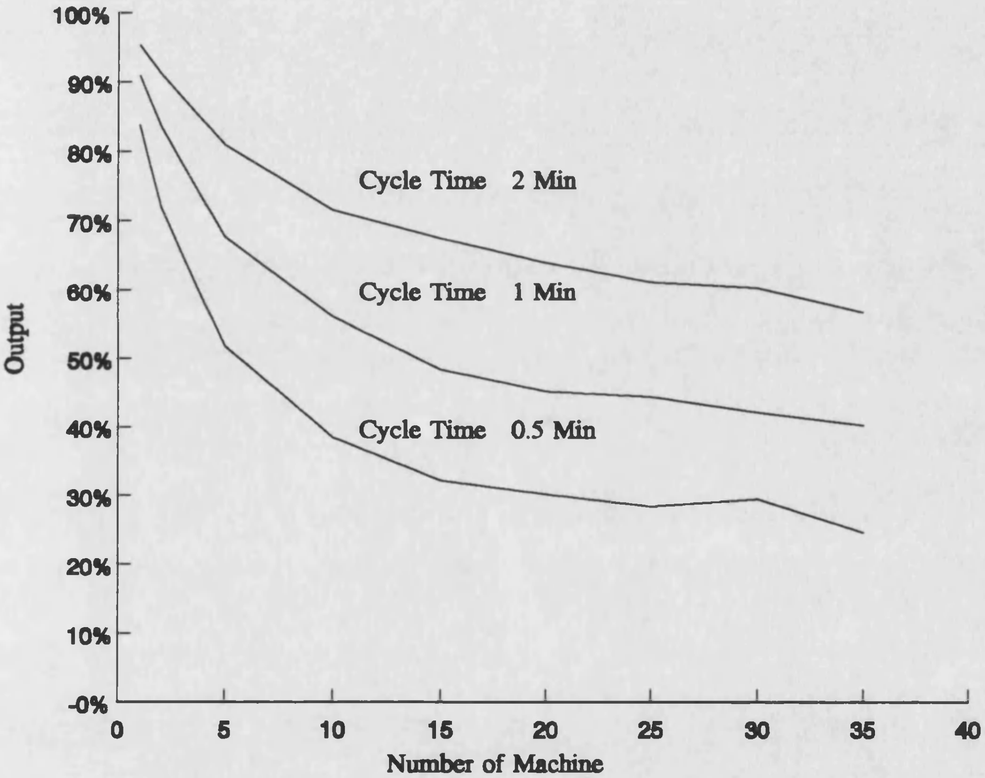


Fig. 6.3 The effect of line length on output with varying cycle times

The results of the simulation runs to determine the effect of changes in cycle time are shown in Figure 6.3. The curves obtained are similar in shape, and all tend towards minimum constant values as the number of machines, N, increases. Again these are the results of short simulation runs.

The results are as expected. Because the breakdown frequency is measured in terms of components and not time (i.e. the number of components between failures) then there is a fall in the %downtime when the cycle time is increased, and thus an increase in %output, i.e. The mean time between failures has increased. The actual number of units produced with a higher cycle time is of course reduced even though the %output is higher.

6.1.2 Variations in Breakdown Frequency (B)

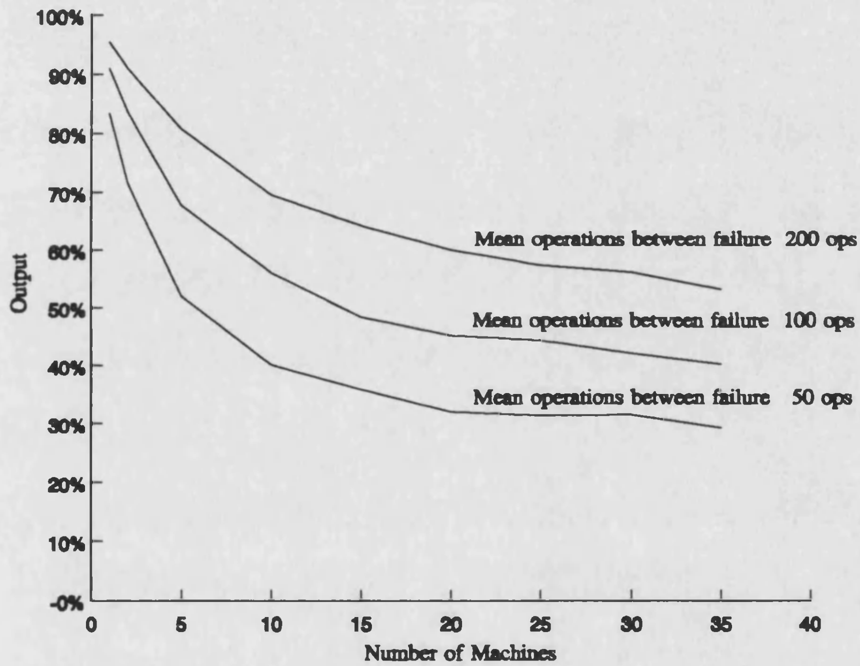


Fig. 6.4 The effect of line length on output with different Breakdown Frequencies

Variation in breakdown frequency, which is shown in Figure 6.4, yield similar results to that of varying the cycle time. If the mean number of operations between failures decreases, the %downtime increases and the corresponding %uptime falls. It follows from the theoretical model, that for constant B/N the %output is constant. i.e The output of the line is the same if the mean breakdown frequency doubles and the number of machines is halved. The results above confirm this as

Output(N=1,B=50) equals Output(N=2,B=100) equals Output(N=4,B=200)
and

Output(N=2,B=50) equals Output(N=4,B=100)

Implied from these results is the fact that if the length of the line is doubled, the machines used on the line must be twice as reliable for the output to remain the same, but this is only valid for short lines. As line length increases and the theoretical model becomes less valid, the simulated line output is higher than the theoretical model so machine reliability no longer needs to double if the length of line doubles.

6.1.3 Variations in Repair Time (R)

The effect of varying the repair time is similar to that of varying the breakdown frequency. This is to be expected since from the theoretical model, the percentage down time for each machine is the same if the breakdown frequency is doubled and the repair time is kept constant, or the repair time is doubled and the breakdown frequency is kept constant. Although the results appear to be similar, for longer lines, a shorter repair time gives better output than less frequent breakdowns. The reason for this is given in the following section.

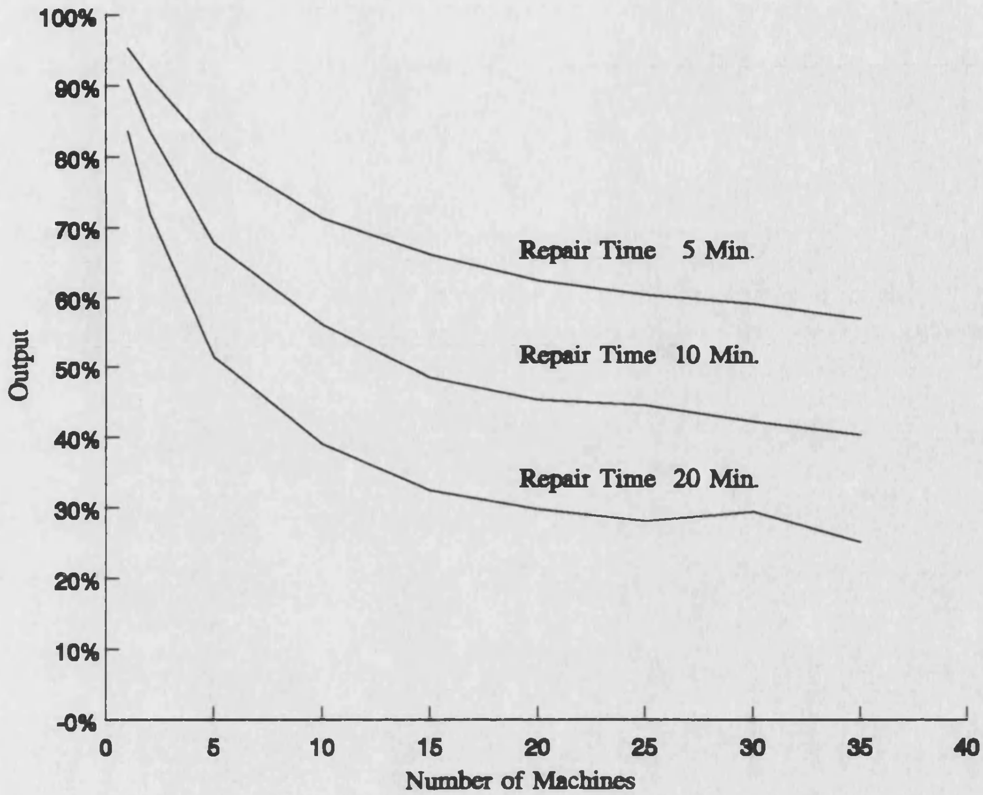


Fig. 6.5 The effect of line length on output with different repair times.

6.2 The combined effect of variations in Breakdown Frequency and Repair Time.

In order to examine the effect of different breakdown frequencies and repair times, but with the same overall downtime, three simulations with the following input parameters were run

Model	1	2	3
C	1	1	1
B	200	100	50
R	5	10	20

Each model has the same percentage downtime. Model 1 has twice the number of breakdowns, of half the length of the base model (Model 2), and Model 3 has half the number of breakdowns of twice the length.

The results can be seen in Figure 6.6. Having more frequent shorter stoppages gives a higher output for the same percentage machine down time. When line length is less than 4, the outputs for the three different lines are nearly identical, only after this does the length and frequency make a difference. Again output of short lines match the theoretical model, but as line length increases more frequent shorter stoppages give a higher output. The reason for these results is again explained by window interference. With shorter more frequent stoppages, there is more interference. This is because more windows are moving on the line which in turn increases the probability of two interfering. The analogy is trains randomly departing on journeys along a track. If there were two very long trains a day moving along the track, there is quite a low chance of an accident, but if hundreds of little trains are moving randomly, there is almost bound to be an accident.

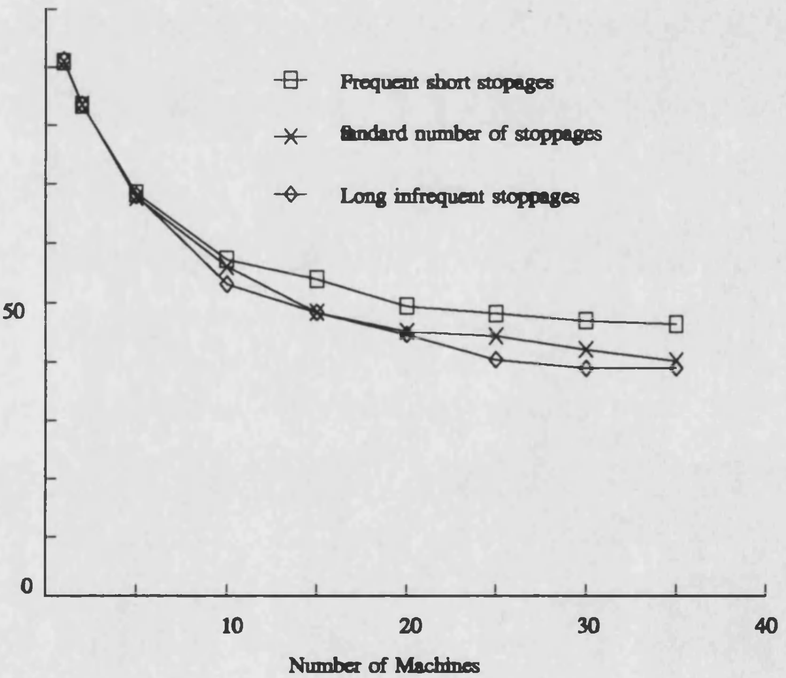


Fig 6.6 The combined effect of stoppage frequency and length on line output.

6.3 Bottlenecks

A bottleneck is defined as a machine (or other part of the line e.g. a conveyor) which constrains the output of the line to a greater degree than any other element on the line. It is in effect "the weak link in the chain".

In general bottlenecks are caused by machines as opposed to other elements such as conveyors. This is because machines carry out a process whose parameters are less easy to adjust than other elements and upgrades are consequently more expensive. The smaller capacity can be due to a longer cycle time, more frequent breakdowns, longer repair and set up times, more frequent tool changes, or indeed a combination of some or all of these factors. Lines are, however, generally built with a balanced cycle time, and so the bottleneck is usually due to increased downtime due breakdowns and tool changes at a given machine.

In the same way that there must be a weakest link in a chain, in theory there can only be one true bottleneck on a line, the lowest capacity machine. In reality, breakdowns occur randomly. The result of this is that *Local Bottlenecks* can occur at other machines on a day to day basis. A local bottleneck is defined as a secondary bottleneck machine (or machines) which causes line output to be reduced because of breakdowns that occur in a pattern which is dissimilar to the other machines.

An example might be a line whose capacity is traditionally restricted by a machine which has long tool changes twice a day resulting in a 10% downtime. There is also another machine which has a smaller stoppages at a random frequency (say approximately hourly) due to say tool failure which result in a 8% downtime. The first machine is the bottleneck on the line, but the second machine however, becomes a frequent local bottleneck. Thus whether a machine is a bottleneck or not may depend on the time span being examined. As a result, simulation runs for lines with bottlenecks have been conducted over a time period equal to 100,000 cycle times.

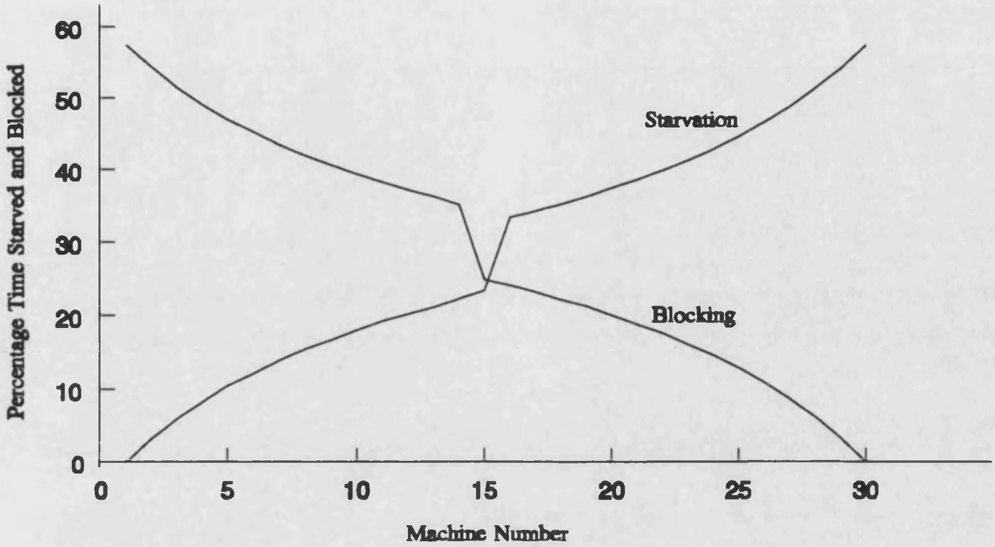


Fig. 6.7. The effect of a 70% bottleneck on a 10% downtime line of 30 machines

In order to understand the effects of a bottleneck on a line it is important to study the effect that a bottleneck has on the blocking and starving of machines. An example of this is shown in Figure 6.7 where a bottleneck of 70% (see Section 6.3.1 for definition of bottleneck size) is placed at Machine 15 of a 30 machine line where all other machines have a 10% downtime. The result of the bottleneck is to generate 'steps' in the starvation and blocking curves which would otherwise be smooth as described in Section 6.1.

The reason for the step is easily explained if the starvation curve gradient discussed in Section 5.5 is considered. The gradient of the line between two machines represents the additional starvation at the second as a result of stoppages at the first. A bottleneck machine will be down for a greater percentage of the time, the result of which is either more frequent or larger starvation windows being generated. These windows pass down the line and as a result the next machine is subjected to additional starvation. Together with the additional windows, there may be additional interference which will of course help reduce the amount of disruption to the lines overall output.

6.3.1. Bottleneck size

It would seem apparent that the greater the size of the bottleneck the greater the disruption to the line and hence the lower the overall line output. Before examining the effect of bottleneck size it is important to define a measure of the magnitude of a bottleneck.

On an otherwise balanced line, the size of a bottleneck, Z, is defined as the additional downtime compared with other machines. The %downtime of the bottleneck machine D_B can be described in terms of the %downtimes of other machines D_{Mean} as:-

$$D_B = \frac{D_{Mean}}{(1 - Z)}$$

For example, consider the line described in Figure 6.7. The bottleneck of 70% and a normal machine downtime of %10 means the bottleneck %downtime is 33%.

The bottleneck size can also be expressed in terms of the increase in the mean failure rate of the bottleneck machine compared with those around it, or an increase in repair time, or an increase in both the repair time and the mean failure rate. For the variation of failure rate alone, the failure rate of the bottleneck machine B_B, can be described as:-

$$B_B = B_{Mean} - B_{Mean} \times Z$$
$$= B_{Mean}(1 - Z)$$

and therefore

$$Z = 1 - B_B / B_{Mean}$$

In order to examine the effects of bottlenecks on the line, a series of simulation runs were carried out on a 30 machine long line based on the model line used in Section 6.1. The middle machine (M15) was then varied as the bottleneck. The resulting variations of the lines output with different size bottlenecks (0-70%) and overall line downtime (0-16%) are shown in Figure 6.8.

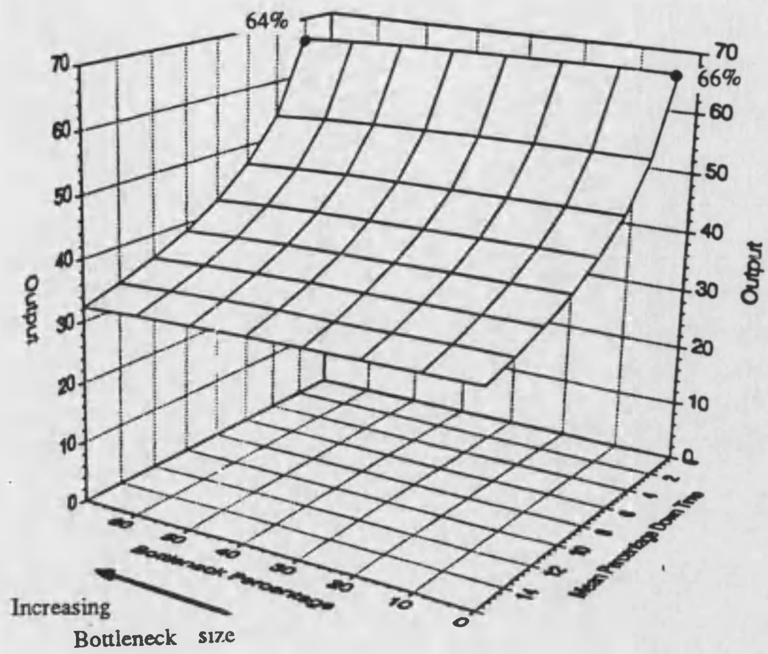


Fig. 6.8. The effect of bottleneck size on output with various breakdown frequencies

As expected, the output of the line falls when a bottleneck is placed in the line. What is surprising is how little the output falls, less than 2% in each case.

This is explained if we examine how each machine's time is spent (shown in Figure 6.8). With the exception of the bottleneck machine, the machines spend between 50% and 60% of their time either blocked or starved, compared with a downtime of only 10%. It is clear that the reason for the decrease in output at a given machine is due to stoppage effects rather than actual stoppages themselves. In this environment, interference levels determine output. There are so many starvation windows travelling downstream and so much upstream blocking, that the addition of more windows caused by the bottleneck is not significant. If we consider the actual number of windows being created in a given period, the addition of a 70% bottleneck only produces 3 times the windows at one machine.

Therefore, in a time t, where t is the time it takes the line to do B cycles, where B is the Mean Breakdown Frequency, on a line with n machines, the number of windows produced on a line with no bottleneck is nt. On a line with a bottleneck of say 70%, the number of windows rise to (n+2)t. For a line of 30 machines, there is only an extra 5% more windows. The results presented in Section 6.1.1 show that on a 30 machine long line a 100% change in the number of windows produced on the line only results in a change in output of 10-15%. Remember also that these additional windows are at the centre of the line where interference is more likely to occur.

Similar results were obtained for a line with a mean downtime of 1%. A 90% bottleneck at the middle machine causes output to fall from 84.0% to 82.5%. A difference of 1.5%. This is an extreme example because with such a small mean %downtime, there are few windows with which the bottleneck's extra windows can interfere. These values confirm the above ideas.

6.3.2 The effect of line length

The results (shown in Figure 6.9a-c) of simulation runs for 3 different length lines show that bottlenecks have a greater effect on short lines. The results are from three lines where the mean %downtime is 4% with an 80% bottleneck at the middle machine (M5, M10 and M15 respectively). The difference in the amount of starvation at the end of the line (which represents the difference in output) between the bottleneck line and the line with constant downtime (x on the diagram) clearly decreases the longer the line gets. This is to be expected since on a long line, the percentage increase in the number of windows due to the addition of a bottleneck is smaller. Also, on a long line there is a greater chance of interference between effects happening before those effects reach the end of the line. Thus the bottlenecks have a greater effect on short lines and reduce output more.

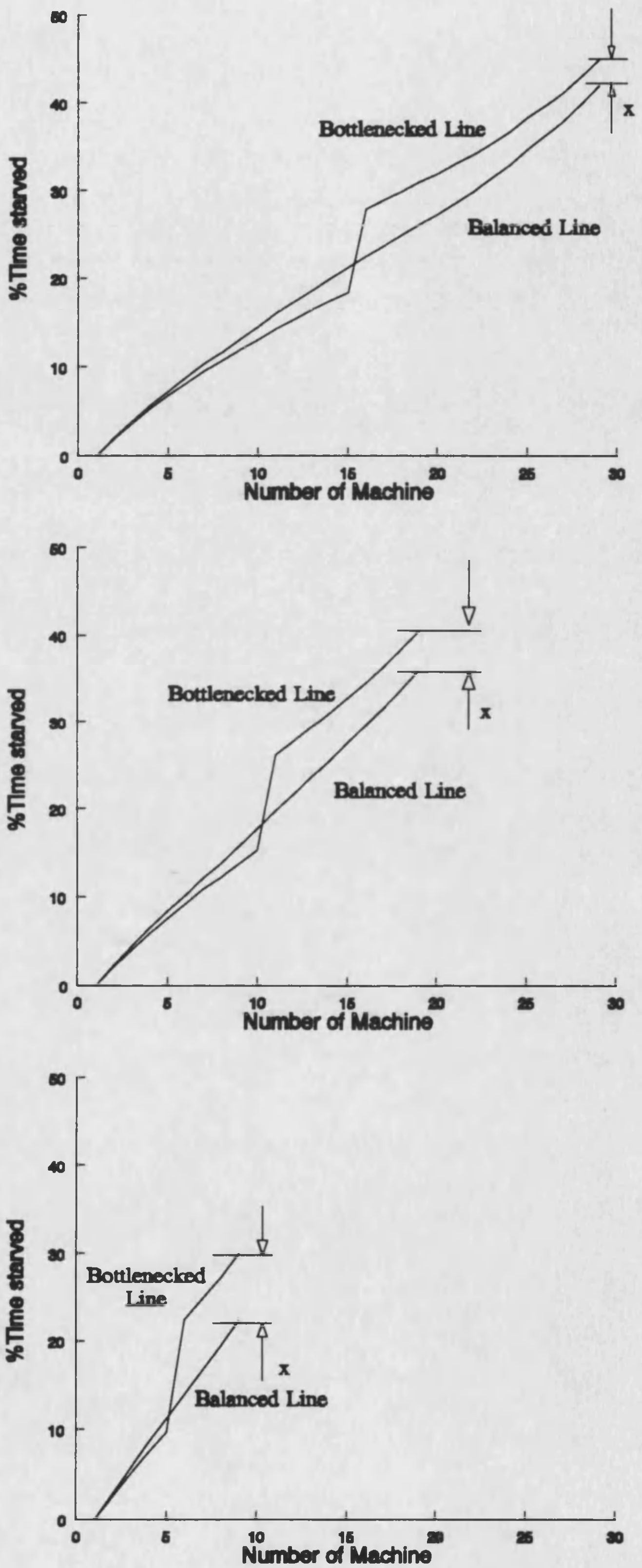


Fig. 6.9. Starvation curves for different line lengths

6.2.3 Bottleneck Position

To determine the effect of the position of the bottleneck in the line, another series of simulation runs was carried out. A line of 29 machines, each with a downtime of 4% was subjected to a 90% bottleneck at different positions along the line. The results are shown in Figure 6.10.

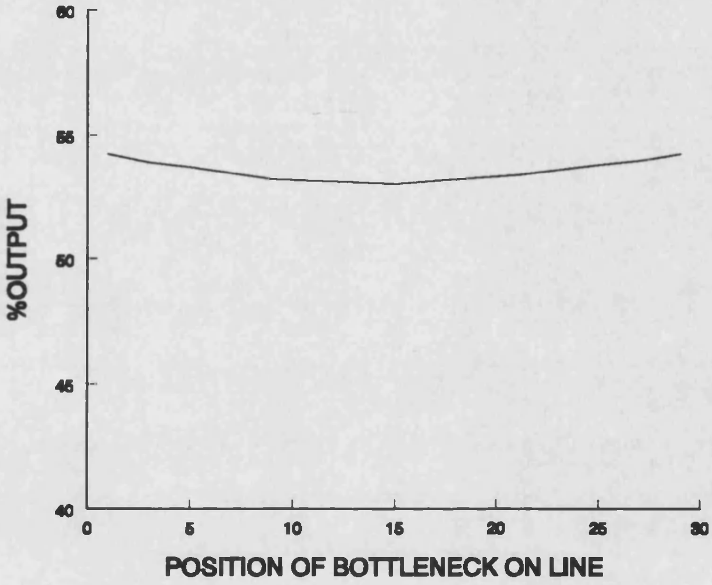


Fig. 6.10. The effect of bottleneck position on a 29 machine line

The results support the idea of the 'Bowl Theory' discussed in Section 3.2 in that the output from the line is higher if the least reliable machine is placed at the end. This may be true, but the difference in output is at maximum 1.2% with a bottleneck of 90%, with a smaller bottleneck the effect of position would be even less significant.

Another point of interest is that the Blockage and Starvation curves for the bottleneck at machines 5 and 25, are mirror images (shown in Figure 6.11 a-b). This together with the symmetry around the centre of the graph in Figure 6.12 supports the reversibility work of Yamazaki & Sakasegawa [1975] which was discussed in Section 3.1.

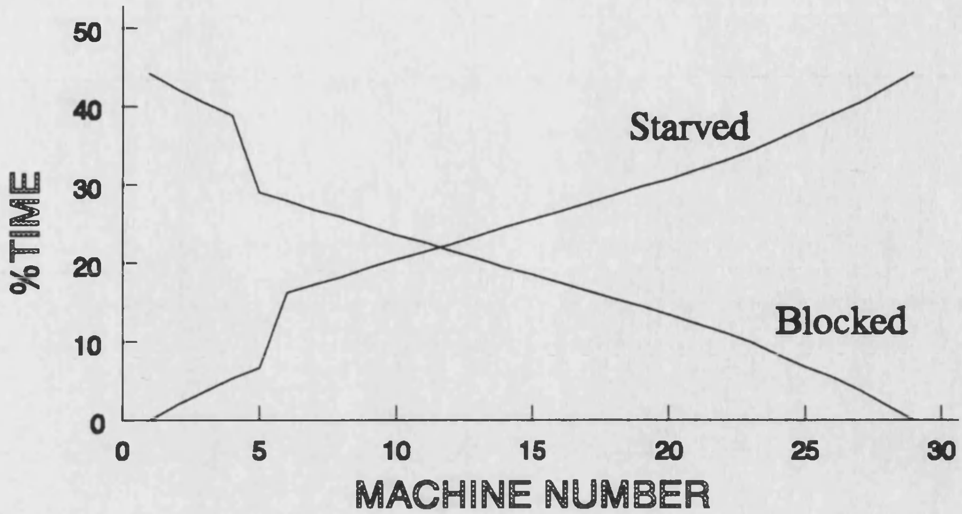


Fig. 6.11a Blocked and Starved curves with a bottleneck at M5

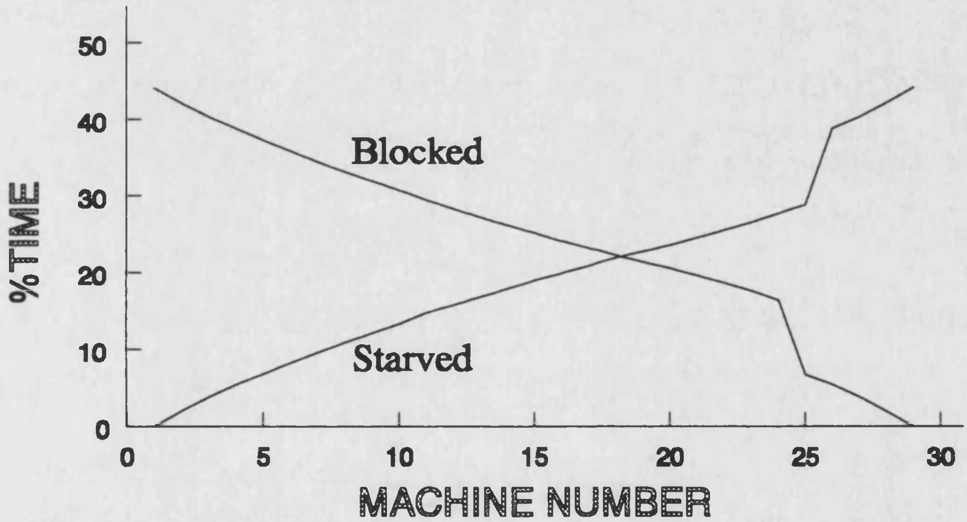


Fig. 6.11b Blocked and Starved curves with a bottleneck at M25

Chapter 7 - The Effect of Buffering Transfer Lines.

A buffer (or interstage storage) is a means by which components can be stored between sequential operations. This, as described in Section 2.5, can take the form of manually loaded pallets, queuing conveyors or automatic racking. The traditional view of buffers as a way of improving output is that an empty buffer will allow machines upstream to carry on working when machines downstream are stopped and similarly a full buffer will allow downstream machines to continue to process when supply from upstream machines has stopped. In both of these cases a machine can continue working when otherwise it would have been forced to stop. Thus the buffer is used to partially decouple different operations from each other.

The need to deal with stoppages both up and downstream does, however, place conflicting demands on the buffer. The ideal is that the buffer should be full in one case and empty in another. In reality the problem cannot be overcome and a buffer before a bottleneck will always tend towards being full, and visa versa for buffers after a bottleneck.

7.1 An alternative view of how buffers work

Although the above description of the way buffers work is true, it only operates at a local level in terms of filling and emptying. By looking at buffers in a different way it is possible to gain a greater understanding of the way they affect the whole line.

Instead of considering buffers as the component store, which they physically are, they should be considered as *a means of retarding the movement of stoppage effects (starvation windows and blockage effects) along the line.* The retardation is done by storing the stoppage effects in the buffer, rather than the view of storing components. Starvation windows are stored in full buffers by emptying the buffer by the size of the window. Blockage effects are stored in empty buffers by filling them by an amount equal to their size. Thus a half full buffer can be said to be a combination of blockage effects and starvation windows all stored in one place.

For example, consider a starvation window equivalent to 5 components travelling down the line. As it reaches a full buffer of 20 components, the buffer will supply the machines downstream. The downstream machines will not be starved as a result of the window, but the buffer stock will fall by 5 components. The buffer now contains 15 components and (more significantly) a starvation window of 5 components. As the line now continues to process, the starvation window does not move down the line since the supply of components to, and the demand for components from the buffer are equal. Thus the window's movement has been halted and it is stored in the buffer. A similar phenomenon happens to blockage effects travelling upstream when reaching an empty buffer, i.e. the buffer has components added to it as upstream machines are not effected and the movement of the blockage effect is stopped. The significance of the retardation can be seen if the buffer of 15 components and a 5 'component' starvation window is again considered. Having stopped the original window from reaching the end of the line, the line continues to process for a further time, t, in which no stoppages occur on the line. Where t is such that the original window would have reached the end of the line and halted line output. A second breakdown now occurs downstream of the buffer. The blockage effect immediately travels upstream until it reaches the buffer. The buffer now fills with components (from the upstream machines which have not been forced to stop) until the downstream stoppage is repaired (say after a time equal to 5 cycle times). The buffer is now full of components and the starvation window has disappeared. Window Interference has occurred in the buffer and, as is always the case with interference, the result is that the line has produced more components since only one window has reached the end of the line. Had the buffer not been present, the interference could not, however, have occurred as the first starvation window would no longer have been on the line.

Should a buffer not be large enough to contain all the effect of a stoppage, it can contain part of it. For example, if a buffer is half full, and a starvation window reaches it which is bigger than the remaining number of components in it, then the buffer can store part of it (i.e. until it is empty). The remainder will travel down the line until it either reaches another buffer, the end of the line or it is interfered with.

This method of considering how buffers operate differs from any other published methodology. Although the buffers operate in the same way, in so much as they take components from or give components back to the line, the effect of buffering on the line is more easily understood as it can be related to the increases in interference discussed in the previous chapter. The lack of published information on this idea of the way buffers work is not surprising as no details of the mechanism of window interference on which it is based have been published either.

It is important to note that the size of a buffer is the maximum number of components that can be stored, not the actual number in the buffer at a given time. A useful measure is the average size of the buffer on the line, S. Where

$$S = \frac{\textit{Total number of buffer spaces on the line}}{\textit{Number of machines} - 1}$$

7.2 The difference between queuing conveyors and dedicated buffers

A consideration at this stage is the difference between automated buffering and queuing conveyors. If two machines are separated by an automated buffer, components can be placed into and taken from the buffer and this a fixed amount of time which is independent of the capacity of the buffer and how full it is. With conveyors, however, there is a cycle time associated with each space on the conveyor (i.e. the time it takes a component to move along the conveyor). The presence of the cycle time results in the effectiveness of the buffering capability of the conveyor being reduced, particularly when long conveyors contain few parts (i.e towards the end of the line).

The conveyor can not contain any part of a starvation window unless

$$\textit{No. of parts on caveyr} > \frac{\textit{Throughput time of conveyor}}{\textit{Machine cycle time}}$$

This can be demonstrated if the example conveyor between two machines A and B shown in Figure 7.1 is considered. The cycle time for both machine A and B is 6t, and the conveyor cycle time is t per space. Parts are moving down the line in Figures 7.1a-c. After a time t=7, Machine A breaksdown at the end of its cycle for a time 10t. Parts continue to move down the conveyor and are processed by Machine B (Figure 7.1 d-f) until a time t=18 when Machine A is repaired and pushes a part onto the conveyor (Figure 7.1g). At this point a starvation window of 17 spaces (10 for the stoppage plus the normal 7 due to the difference in the cycle time of the conveyor and the machines), can be seen between the parts on the conveyor. Parts continue to move down the conveyor and be processed by the machines (Figures 7.1h&i) until a time t=28 when Machine B finishes processing the last part before the starvation window. Machine B is now starved until the next part on the conveyor reaches it at a time t=39 (Figure 7.1l). Machine B has been starved for a time of 10t, the same time that Machine A was broken down. Thus the three components present on the conveyor at the time t=1 did not act as a buffer.

Had the machines fed directly into and out from a buffer which had three components, Machine B would not have been starved as the whole starvation window could have been stored. The result of this disparity when applied to a real line can be significant. Consider an example of a queuing conveyor with space for 150 components (derived in the case study in Chapter 9). The conveyor throughput time is 7.5 minutes, where the machine cycle time is 1/3 minute. Thus by applying the above equation, 22 of the 150 buffer (approx 15%) spaces do not contribute to storing starvation windows.

Conveyors, however, must not be dismissed as a type of buffer. It must be remembered that in reality there is often a minimum distance by which machines must be separated for which conveyors must be used even if it is only to and from an automated buffer. Also, dedicated buffers are generally more expensive than an equivalent conveyor, particularly for small amounts of buffering. This difference between conveyors and buffers has not been found described in any other research. It must be assumed that this, again, is due to the lack of study into the movement of starvation windows along the line.

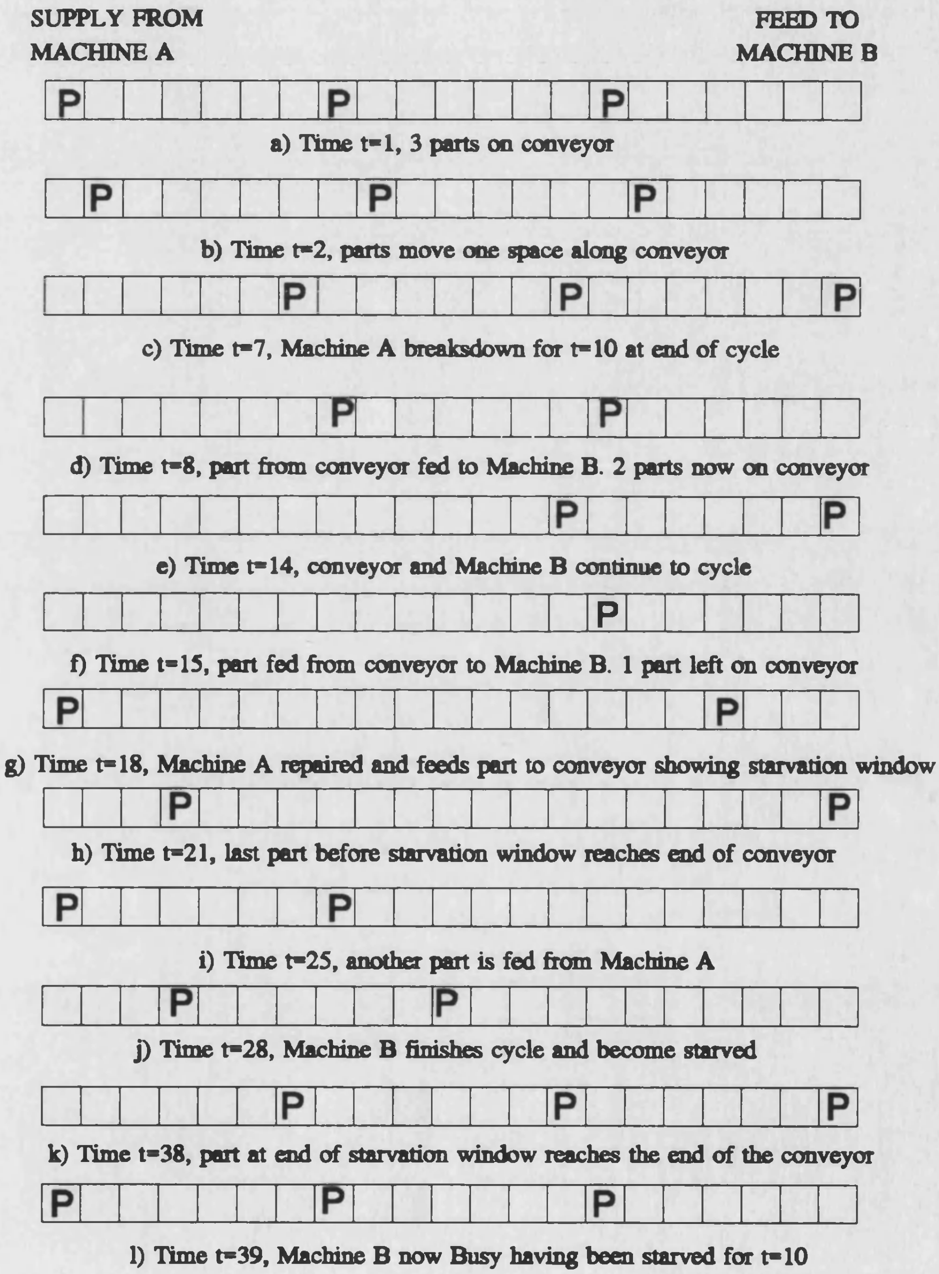


Fig 7.1 The passage of parts down a queuing conveyor

7.3 The effect of buffering a balanced line

7.3.1 Even buffering

From previous published work it is clear that buffers improve output, and that the position of the buffering is as important as the amount. In order to gauge the improvements in output made, a series of simulation runs were carried out on a balanced line with a machine downtime of 8%. Equal size buffers were placed between each machine on lines whose length varied from 2 to 35 machines long. The results are shown in Figure 7.2. It can be seen from the graph that the addition of what must be considered as quite small buffers has a large effect on output. For example, in a line of 35 machines the increases in output due to placing a single buffer between machines is over 30% (%uptime rises from 41% to 54%). Placing two buffer spaces between each machine yields a 50% improvement (41% to 62%), whilst an 87% improvement is achieved by placing five buffer spaces between each machine, raising the output of the line from 41% to 76%.

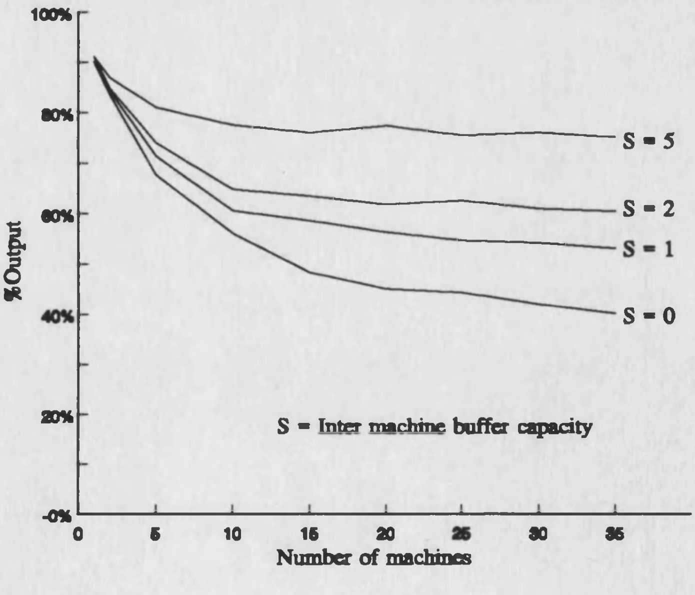


Fig. 7.2 The effect of even buffering on balanced lines of various lengths

The gains made clearly increase with line length, since the potential to increase interference is greater, as is the ability to accomodate stoppage effects over several buffers. This latter point is important because all stoppages in this series of tests were for a period equal to 10 cycle times, thus for S=1, a starvation window would be spread over 5 buffers and 5 machines (a machine also acts as a buffer because it too can store a component).

The gains made also increase as the breakdown frequency of each machine increases i.e. the machines become more unreliable. This is shown in Figure 7.3 which shows how the output of a 30 machine line with no bottleneck varies with different buffer capacity and different machine downtimes.

Readers are reminded that %downtime is expressed as a percentage of uptime, not total time (See Section 6.0). With no buffers the output of the line falls as described in the previous chapter. It is clear that the even buffering improves the output of the line substantially.

The addition of 16 buffers between each machine (the largest amount tried during this series of simulation runs) was sufficient to reduce the total amount of time each machine spent blocked and starved on a line with machine downtimes of 2% to less than 1%, giving an output of over 97%.

Since a machine can not perform better than if it is never blocked or starved, this represents an output within 1% of the maximum. Without buffering each machine spent over 30% either blocked or starved, giving an output of 68%.

With the %downtime of each machine at 16% (a figure higher than would be expected in most cases), an inter machine buffer of 16 components increased the output to within 4% of the maximum possible (i.e. Total blocking and starvation time less than 4%). Although this could be improved by the addition of further buffering, it still represents a doubling of the output compared with the unbuffered line (81.9% compared with 33.6%).

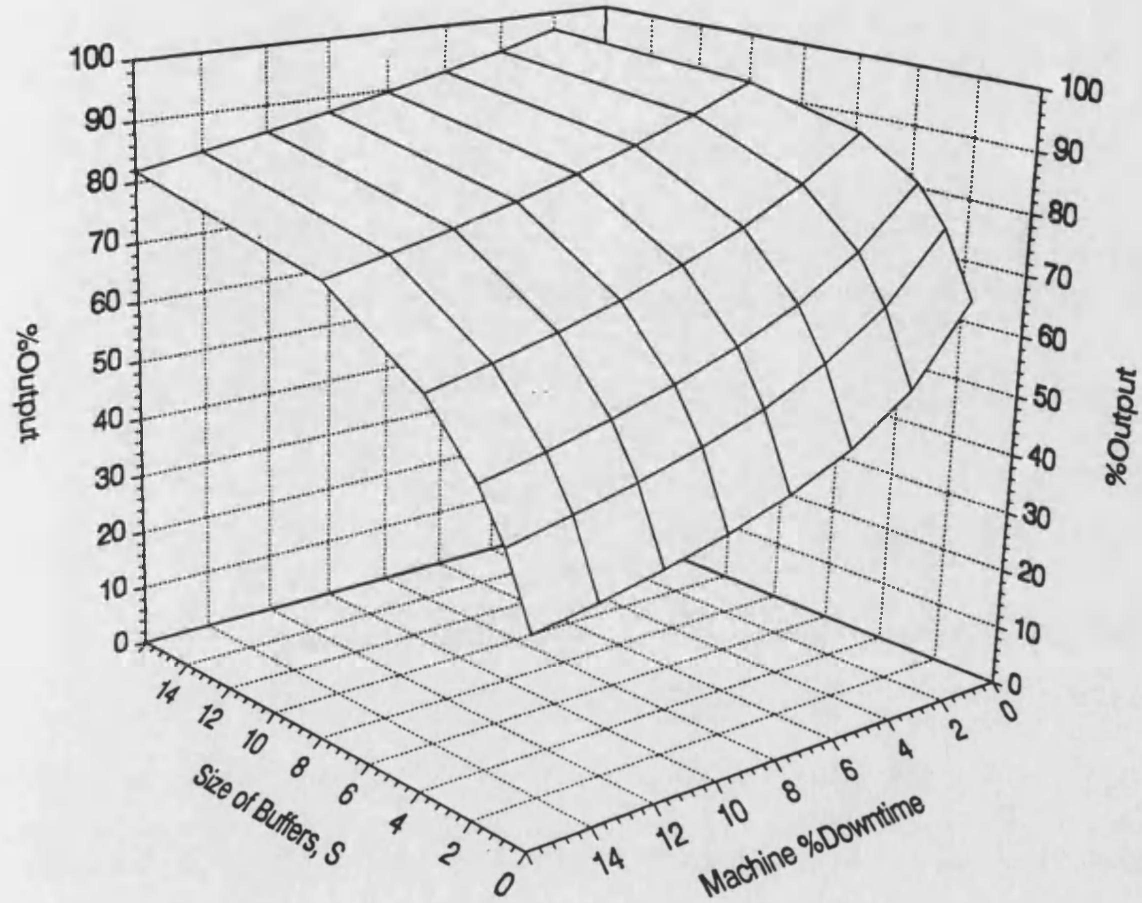


Fig. 7.3 The effect of even buffering and %downtime on a line of 30 machines

The effect on machine starvation of the buffering of the line with a 16% downtime is shown in Figure 7.4. The same effect was seen when comparing the difference between actual and theoretical outputs in Section 6.1. The interference is greatest at the middle of the line, and this results in the flattening of the starvation curve. This effect is also experienced in unbuffered lines where there is more interference present (i.e long lines compared with short ones). However, it is exaggerated in the buffered environment (more buffering) and is clearly demonstrated in the graph.

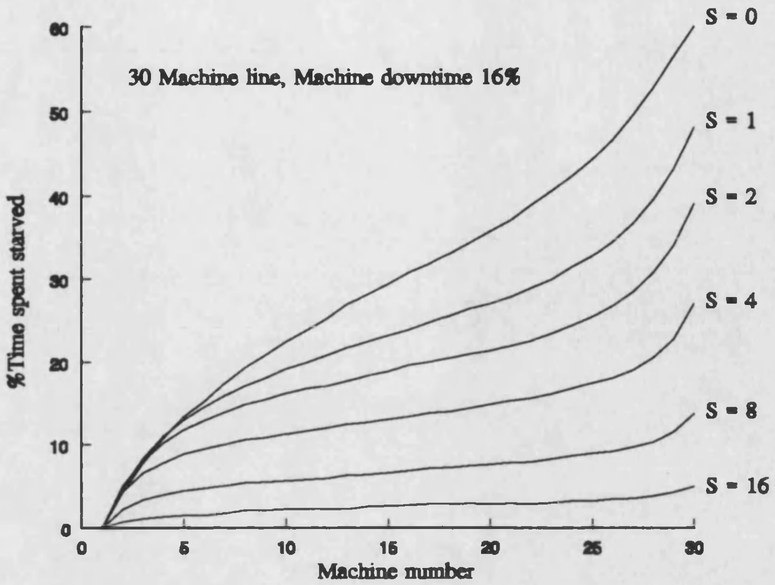


Fig. 7.4 The effect on the starvation curve of adding buffers

7.3.2 Other buffering patterns

The majority of study carried out by the author on balanced lines has been directed at even buffering. There is, however, no reason why this should yield the highest output. In order to examine whether other buffering strategies could produce better results a series of simulation runs were carried out putting a 'square' buffering pattern on to a line of 30 machines each with a 10% downtime. Square buffering is buffering placed in rectangular patterns as shown in Figure 7.5a in which S=1.

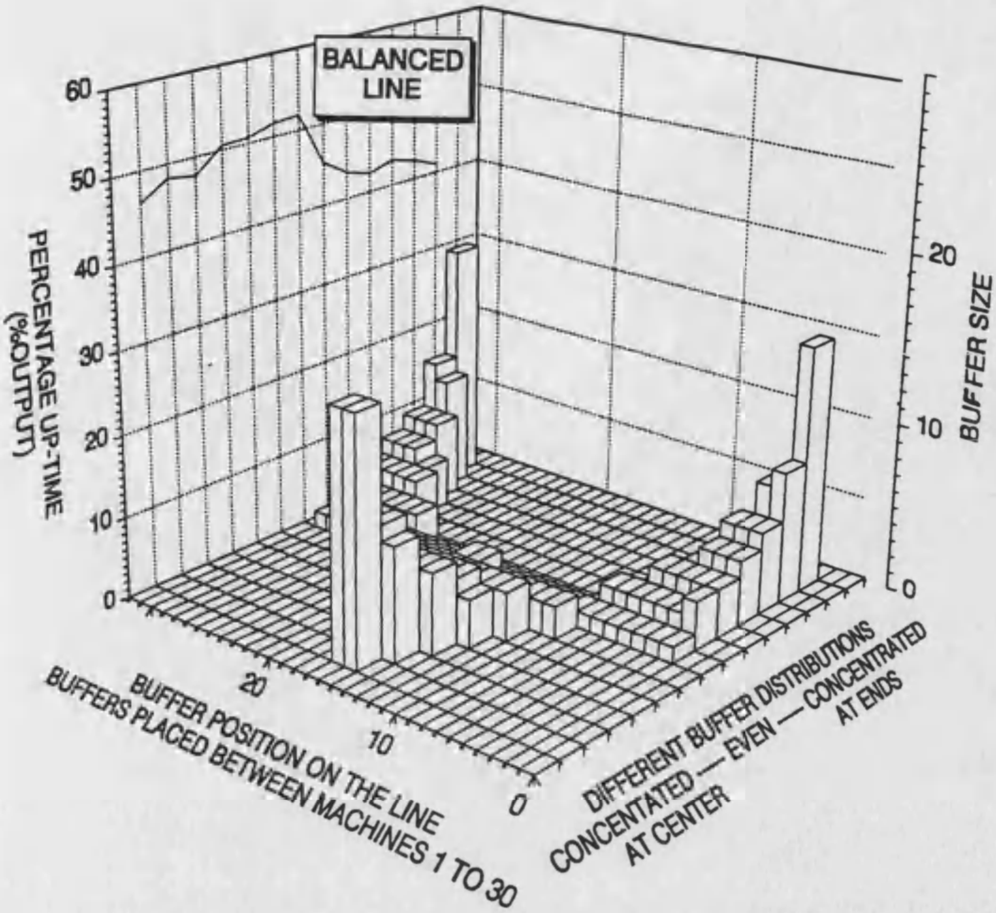


Fig. 7.5a Output for different 'square' buffer distributions for a balanced line (S=1)

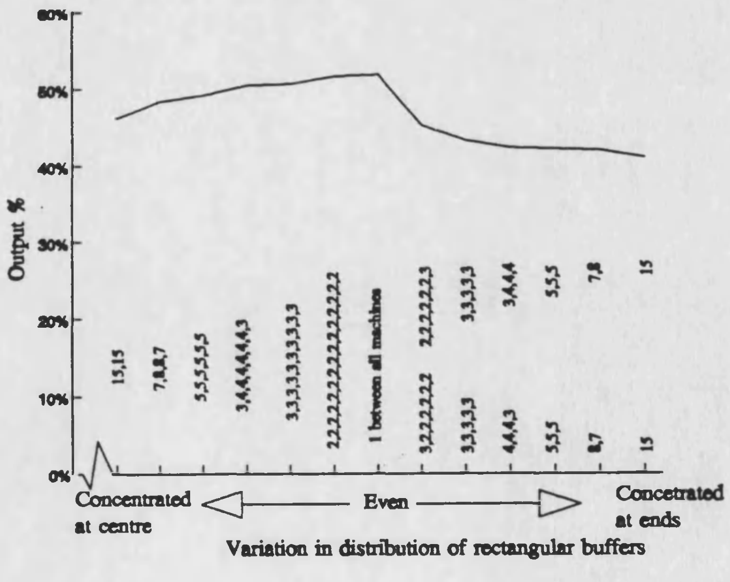


Fig. 7.5b Variation of output (buffer distribution shown in figures) S=1

The rectangular patterns are a mirror around the centre of the line. When buffering is concentrated at the centre (The foreground of Figure 7.5a and the left hand side of Figure 7.5b), buffers of 15 spaces each are placed either side of M15. This can be said to be a rectangle 15 spaces high by 2 positions wide (15x2=30). Other rectangles are also used and get progressively wider but less tall using the same number of buffers (e.g. 5x6 and 3x10). In some cases exact rectangles can not be made. In these cases the closest to the required rectangle is used (e.g. 7,8,8,7 around the middle machine also has a total of 30). The lowest and widest buffer pattern is an even buffer pattern of one buffer space between each machine.

Other patterns which concentrated buffering at the ends of the line were also used. Again rectangles were used, one at either end of the line with similar sizes to those rectangles where buffers were concentrated at the centre. These patterns are shown in the background of Figure 7.5a and on the right hand side of Figure 7.5b.

Outputs corresponding to these buffer patterns are shown in the graphs. Note the significantly lower output with buffering concentrated at the ends of the line compared with the centre. The even buffer pattern gives the highest output (53%). Other strategies show that buffering towards the centre is preferable to that at the ends of the line. For example, the case of buffers of 3 spaces each. Referring to Figure 7.4b, when a buffer pattern of 5 x 6 is used in the middle of the line, the output is 50%. When it is separated in a pattern of 3 x 5 at each end the output is lower (44%). It was stated in Chapter 6 that interference occurs mainly in the centre of the line (the flattening of the S-shaped starvation curve). Placing buffers at the ends results in a lower number of windows being retarded where interference is most likely. Consequently output is lower.

At this point it is worth attaching some criticism to these experiments. With 30 machines, there is no middle machine for the buffers to be put either side of. Also, where the buffering is even, there are only 29 buffers but in all other cases 30 buffers are used. Although these tests favour the uneven 30 buffer patterns, even buffering was still found to be significantly more effective. It must also be pointed out that these are interim results. As such they were conducted to identify trends and even allowing for the short comings of the techniques used, they have enabled even buffering of balanced lines to be identified as the best pattern of those tested.

7.3.3 Conclusions on buffering a balanced line

The results presented above show that on a balanced line, even buffering gives a higher output than any other buffer pattern tested. The gains in output with different machine downtimes have been shown to be significant, even with low levels of buffering (S less than 4). In the cases tested, larger buffers (16 components between each machine) were found to be sufficient to ensure that the output from the line was within a small % of the theoretical maximum which is dependent on downtime.

No further examination was carried out on the balanced line as it is a much simplified case, and can be considered as a bottleneck line with a 0% bottleneck.

7.4 Buffering a line with a bottleneck

7.4.1 Even buffering on a bottlenecked line

The effect of even buffering on lines is shown in the graphs in Appendix II. These graphs (a-h) show the output for 30 machine long lines with buffering in the range S=0 to 16, bottlenecks of 0% to 70% and machine downtimes of 2% to 16%. In all cases, the bottleneck is at M15.

As would be expected, all the graphs show the gains in output made by the addition of buffers are significant and the addition of buffering is again seen to follow the law of diminishing returns, i.e. as more buffering is placed on the line, then even more is needed to be placed on the line again to achieve the same gains.

With low buffering levels (S less than 4), the effect of the bottleneck is very small. As described in Chapter 6, the output of the line in this environment is dependent on the effects of stoppages on other machines and on the levels of interference. The addition of extra stoppages at the centre of the line creates only a small increase in the effects on other machines.

As the buffering is increased, the level of interference increases which in turn increases output. Sufficient buffering in this situation means that nearly all stoppage effects interfere and the output approaches the maximum. In the case of the balanced line this means all machines are never blocked or starved and the output is solely dependent on machine downtimes. The addition of a bottleneck to such a line which has nearly complete interference results in a fall in output. This is because the output of the line now becomes determined by the output of the bottleneck machine. Thus it can be seen that for a given line, bottlenecks are more significant when there are high levels of interference (accepting the point made in Section 6.3.2 that bottlenecks are more significant on short lines).

On the lines of machines with small %downtimes (e.g 2% - Appendix IIa), the effects of imposing a bottleneck are not easily seen when there is no buffering. With buffering of S=16, the output of the line is maximised for both the balanced and bottlenecked line (i.e. the machines are neither blocked or starved). In this environment, the output of the bottleneck machine dictates output and increases in bottleneck size result in a decrease in output, and the bottleneck is therefore significant.

Increases in %downtime result in the bottleneck being more evident with lower levels of buffering. For example, the 70% bottleneck is only evident for S=16 for a downtime of 2%, yet a buffer value of S=4 is sufficient to expose a 70% bottleneck at a downtime of 16%.

For all these simulation runs, the repair time for the machines was a fixed value of 10 times the cycle time. In order to achieve maximum output in the unbuffered environment, there must be no blocking and starving. As such, stoppage effects can not be allowed to spread over several buffers as described in Section 7.1. The minimum values for buffering to achieve this is obviously S >10. Tests were run with S=8 and S=16, but the maximum output could only be achieved with S=16.

Although in all cases the buffer must be greater than S=10 to achieve the maximum output, in the bottlenecked environment tested, a smaller buffer achieves a greater improvement than it does on a balanced line. This can clearly be seen in the results for the downtime of 16%. At S=8 the larger the bottleneck the closer the output is to the reduced maximum, i.e the smaller the gradient of the line from S=8 to S=16. The reason for this is twofold. Firstly, in the bottlenecked situation, effects spread over several machines have little affect, provided they are not on the bottleneck machine. This is because these machines must spend a certain amount of time blocked and starved at some stage, and secondly, if a stoppage occurs near the bottleneck and its effects try and spread over the buffers either side of the bottleneck, the chances of interference are higher due to the increased number of opposite effects produced by the bottleneck machine.

The effect of the changes in output for the line with a machine downtime of 16% and a bottleneck of 70% can be seen on the starvation curves as shown in Figure 7.6. As buffering is added, the gradient decreases as a result of increased interference. The buffer of S=16 required to ensure maximum output, results in the starvation of the machines up to M15 being 0% - a level graph up to M15 (i.e. starvation windows do not move down the line), a step increase as a result of the bottleneck, and another level portion over the second half of the line.

The level area over the first half of the line shows that these machines are never starved, thus all starvation windows are interfered with before reaching the next machine. The size of the step increase in starvation past the bottleneck is the difference between the downtimes of the bottleneck machine and the other machines on the line. Downstream of the bottleneck the machines are subjected to additional starvation as a result of the increased number of windows produced by the bottleneck.

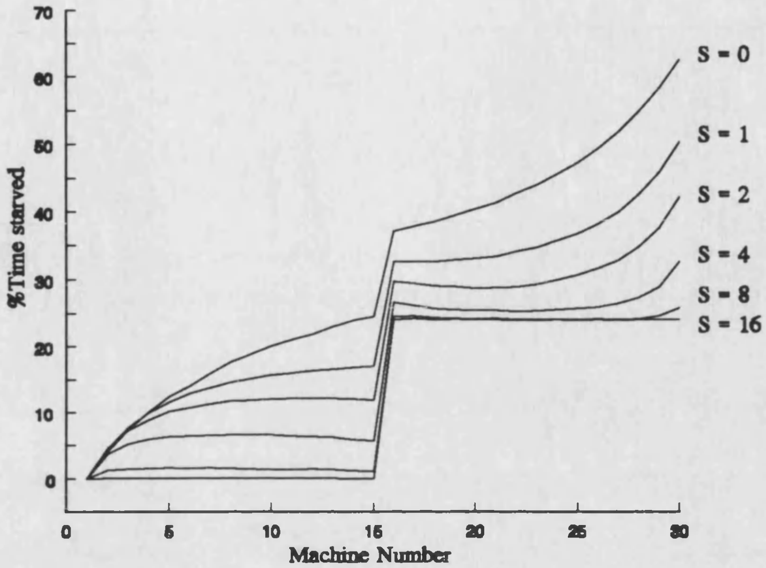


Fig. 7.6 Starvation curves for a line of 30 machines with a 70% bottleneck at M15 and various levels of even buffering.
(note 0% on the y-axis is offset)

The blockage graph is a mirror image of this graph.

Further increase in downtime and bottleneck size would exaggerate the effects described in this section.

7.4.2 Square buffering

The effect of square buffering (identical to that used in Section 7.3.2 (Figure 7.5a&b)) on a line of 30 machines with a 10% downtime and an 80% bottleneck is shown in Figure 7.7a&b. The difference between the balanced and bottlenecked line can clearly be seen. On the balanced line, output falls as the buffering becomes more concentrated towards the centre. With a bottlenecked line, however, the highest output is midway between even buffering and all the buffering concentrated at the centre.

The reason why different buffer patterns produce different outputs is due to the level of interference. The buffering levels in this example are comparatively small, and as such the effect of stoppages on other machines determines output.

By increasing the chances of interference where it is most likely to occur, the output is greater. Thus it is a combined effect of encouraging interference where it is more likely to occur (in the middle of the line), and increasing interference where the greatest number of windows are produced.

The implications of this are that, for a bottleneck, the buffering requirements are two fold. A general buffer on the line to increase interference, particularly in the middle of the line, plus additional buffering around the bottleneck to increase the interference with the extra stoppage effects being produced there.

These requirements vary as buffering is placed on the line, in so much as the more buffering you have evenly spread on the line, the more effect the bottleneck machine has on output. Consequently more buffering will be required around the bottleneck to ensure (particularly as the maximum output is reached) that the amount of time it spends blocked and starved is minimised.

These results also demonstrate the importance of correct buffer positioning. For the bottlenecked line, the difference in output between putting all the buffering at the ends (the classic Raw Material Store and Finished Part Store) and buffering near the bottleneck amount to an increase of uptime from 34% to 46%, an increase in actual output of 35% achieved entirely by moving buffers around the line.

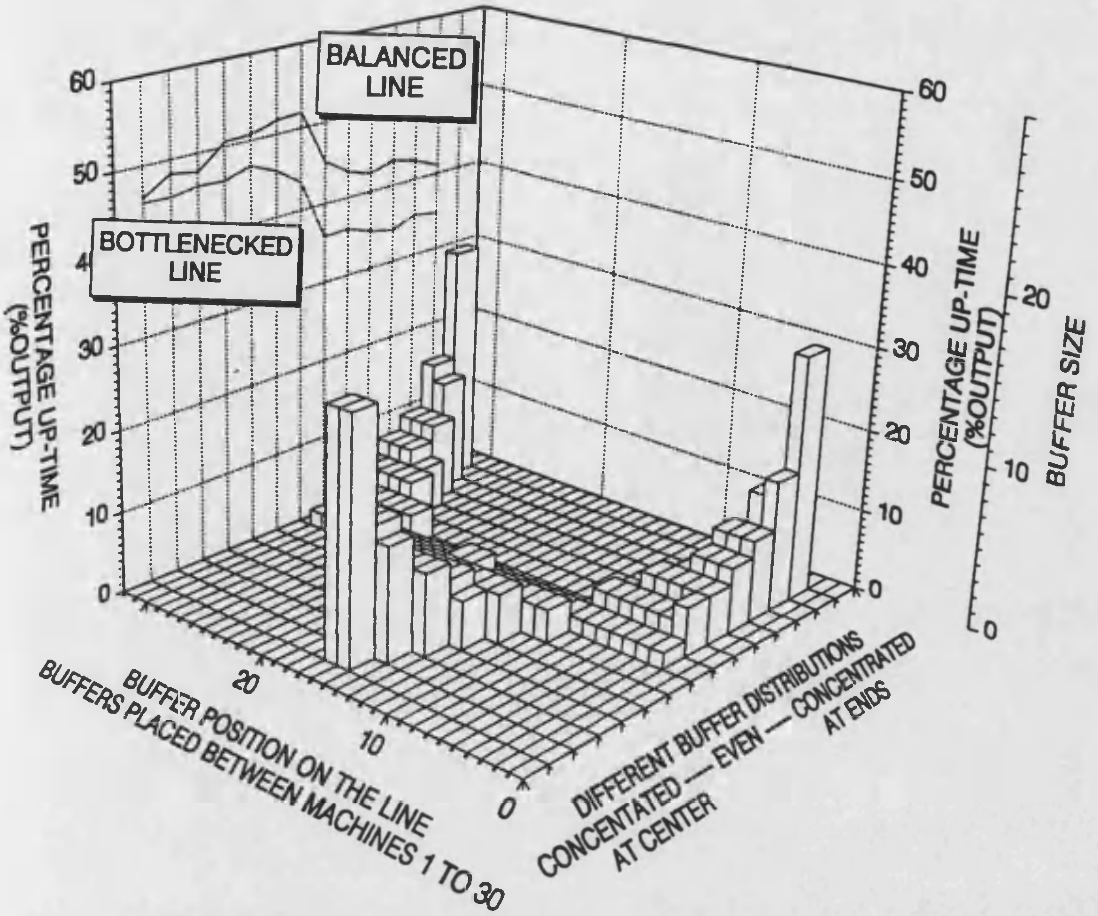


Fig. 7.7a Output with different 'square' buffer distributions for a bottlenecked line

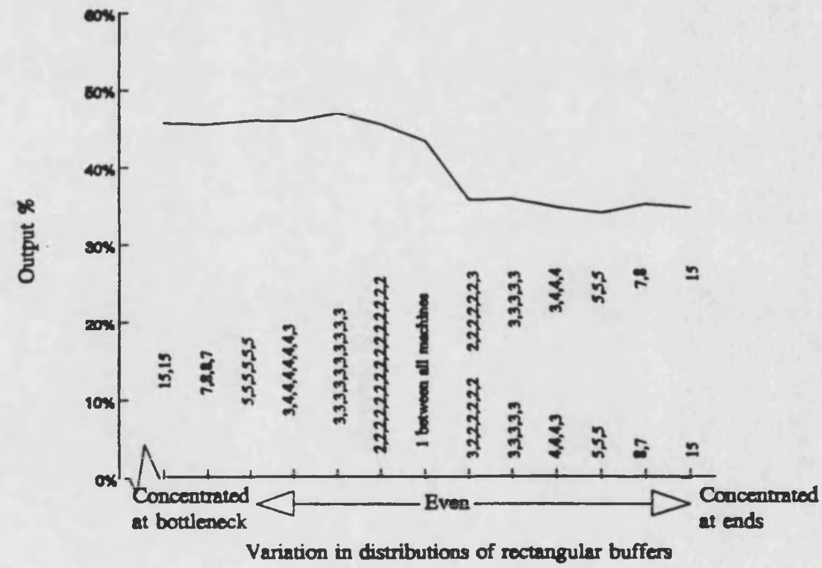


Fig 7.6b Variation of Output (buffer distribution shown in figures)

7.4.3 Conclusions from buffering a bottlenecked line.

The results presented above show that for a line with a central bottleneck, the bottleneck has little effect when the line is unbuffered. If even buffers are placed on the line, not only does the output increase, but the bottleneck begins to become significant in dictating the output of the line. Sufficient buffering on the line will ensure that the bottleneck machine is never blocked or starved. In this case, the output of the line is maximised.

Having tested other buffer patterns on a bottlenecked line, it is clear that even buffering is not the best buffering to use in order to maximise output. Concentrating buffering around the central bottleneck results in a rise in output. The degree to which the buffering should be concentrated at the centre is, however, a complex problem which is dependent on the many line parameters.

7.5 'Near Ideal' buffer pattern

From the above results it is clear that buffer position can greatly affect output. Although the square buffer distributions have shown some patterns to be better than others they fail to show if these are the best. There is, of course, no reason why they should be. In order to determine an ideal pattern an iterative technique can be used by adding buffers to the best current line to try and determine a 'next best' solution.

7.5.1 Technique to find 'Near Ideal' buffering patterns

In order to determine the near ideal buffer pattern the following method was used:-

Starting a 29 machine line with no buffering and a bottleneck at the middle machine. Simulate the line with a buffer between M1 & M2 and also between M28 & M29 (i.e. One buffer either side of the line). Re-simulate the line with the buffers between M2 & M3 and M27 & M28. Continue testing different buffer positions until all 14 possible buffer positions are tried (the last one being with buffers either side of the bottleneck machine M15). Select the result of the simulation run with the highest output and use that line as the input for the next series of simulations where a further two buffers are added.

Thus as each series of simulations are carried out, 2 buffers are added to the line (one either side of the bottleneck) and an iterative solution is generated. The resulting build up of buffer spaces will of course be a mirror image around the bottleneck.

Although this method can be described as finding the 'ideal' pattern, it relies on two assumptions. Firstly, that a mirror image is best. This may not be the case since starvation windows and blockage patterns travel down the line at different rates. Starvation windows take time to travel downstream even in an unbuffered line. The movement of blockage patterns is, however, dependant on the buffering levels. On an unbuffered line, blockages move upstream instantaneously. As buffers are added, however, their movement is slowed as they must wait for buffers to fill. The second assumption is that by sequentially adding 2 buffers, the best buffering will be achieved. Local optima can mean that the overall result can be distorted. For example consider cutting up a cake for different numbers of people. Starting with a whole cake for one, then add another person, and the cake is divided up, i.e. the cake is cut in half. A third person comes along, a further cut to provide three pieces results in two quarters and a half, thus the optimum share out is not achieved by sequentially adding people and extra cuts into the cake.

Buffer patterns derived from this technique of sequentially adding buffers must, therefore, be described as 'near ideal' buffer patterns. There are also drawbacks in terms of experimentation. The two main problems are as follows:-

• **Accuracy.** The difference in output levels between different buffer position can be very small. In cases where the results of the two options are equal, assumptions have to made and the decision was taken to place the buffer nearest the bottleneck. Tests were carried out to see if this rule was valid and the buffering technique was found to self correct i.e. given the choice of locations X and Y, if you chose to buffer at X, Y would be the highest output during the next iteration, and vice versa.

• **Experimentation time.** The main example presented later in this chapter involved placing 56 buffers on a line of 29 machines. Although for much of the time all 14 possible buffer places were not compared (the end positions yielded far lower outputs, particularly with low buffer levels), the result for this line alone represent over 2 months of simulation work. This is why a mirrored pattern of adding buffers on a symmetrical line was used, since to add the buffers individually (as would be required on an unsymmetrical line or when using a non-mirrored pattern) would have increased the number of simulations 4 fold.

7.5.2 Application to a bottlenecked line

The near ideal buffer pattern has been developed for a line of 29 machines. Each machine on the line had a downtime of 2% with a 90% bottleneck at the middle machine, M15. Figure 7.8 shows how the output improves as the buffers are added, together with the resulting buffer patterns. The addition of buffers is continued until a value of S=2 is reached.

Table 7.1 shows, in more detail, how the buffering was built up during successive iterations. The buffering can be seen to build up in different stages. These are as follows :-

a) Initially, the buffering is added at the bottleneck until 5 buffers are either side of M15.

b) Buffers are then placed at machines going away from the bottleneck until 16 buffers have been placed on the line.

c) A small addition is made to the buffering at the bottleneck.

d) The buffering continues to spread from the bottleneck until 40 buffers have been placed on the line.

e) Again more buffers are added at the centre.

f) And again more buffers are added away from the buffer.

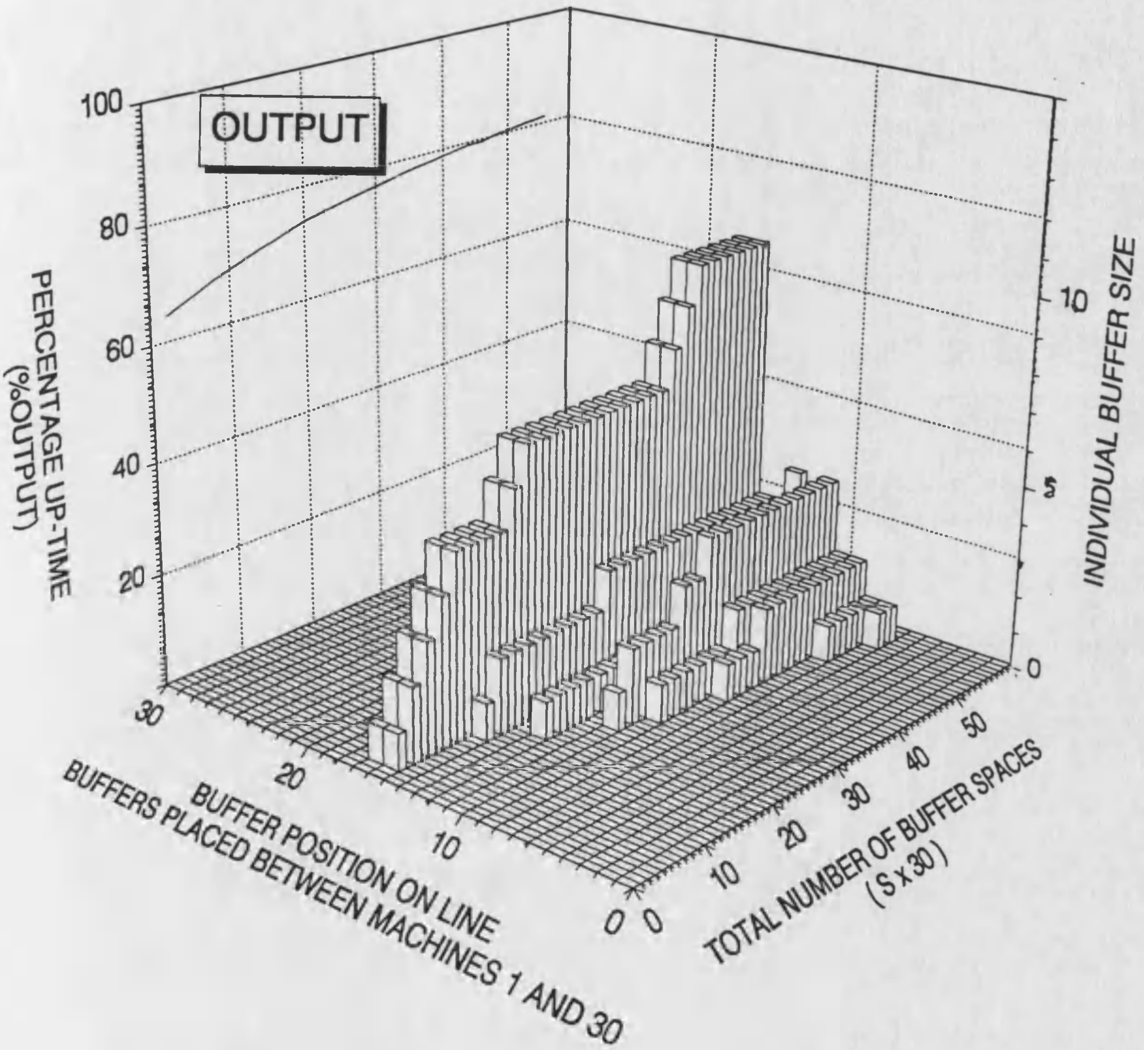


Fig. 7.8 The increase in output with the build up of the 'Near Ideal' buffer pattern.

This recurring pattern of adding buffering at the bottleneck and then away from the bottleneck follows the TOC, in that initially the bottleneck is the constraint on the line. Following buffering (5 either side in this case) the bottleneck is no longer the constraint. The effects of stoppages of other machines is now determining output. This results in buffering being placed away from the bottleneck, which in turn increases output and causes the bottleneck to become the constraint on the line once more.

This cycle of buffering the bottleneck until it is no longer a constraint and then buffering other points on the line until the bottleneck re-emerges as the constraint can be repeated until the maximum output of the line is reached when the bottleneck machine is never blocked or starved. The example shown in Figure 7.8 and Table 7.1 does not, however, add sufficient buffers for maximum output to be reached. The reason for this was that the amount of simulation time required to do this would have been too great.

A point worth noting is that the transition made between points a) and b) above occurs when the output of the buffered and bottlenecked line is similar to the output of the same line without buffers or bottleneck. Thus the buffering at the bottleneck has compensated for increase in stoppages. The line becomes 'equivalent' to the balanced line which should be buffered evenly. As this even buffering is placed on the line, however, the interference around the bottleneck increases, output rises and the bottleneck once more becomes the constraint on output.

Figure 7.9 shows the starvation curves for the line at the turning points between bottleneck buffering and buffering the rest of the line. Figure 7.9a is an exception in that it shows the buffering affecting the step caused by the bottleneck which is not a transition point. At 5 buffers either side of the bottleneck (Figure 7.9b) the gradient of the starvation curve across the bottleneck is very small i.e. The amount of time the machine after the bottleneck spends starved is only marginally greater than that of the machine before the bottleneck. As such the bottleneck is not causing any additional starvation on the line. It is therefore no longer the constraint on the line since it contributes no extra stoppage effects on the line.

The buffering then added around the bottleneck results in an increase in interference away from the bottleneck. This affects the starvation curve by creating a level portion either side of the bottleneck (Figure 7.9c). The level portion indicates that these machines are not adding extra stoppage effects on the line.

Machine number
(buffer spaces are between machines)

Number of buffers		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	Output (%)
2															**1**	**1**															64.29
4															**2**	**2**															65.61
6															**3**	**3**															66.92
8															**4**	**4**															68.14
10															**5**	**5**															69.22
12														**1**	5	5	**1**														70.22
14														**2**	5	5	**2**														71.19
16												**1**		2	5	5	2		**1**												72.12
18												1		2	**6**	**6**	2		1												73.05
20												1		2	**7**	**7**	2		1												73.89
22										**1**		1		2	7	7	2		1		**1**										74.48
24										**2**		1		2	7	7	2		1		**2**										75.04
26									**1**	2		1		2	7	7	2		1		2	**1**									75.58
28	(S=1)								1	2	**1**	1		2	7	7	2		1	**1**	2	1									76.15
30									1	2	1	1		**3**	7	7	**3**		1	1	2	1									76.64
32									1	**3**	1	1		3	7	7	3		1	1	**3**	1									77.13
34								**1**	1	3	1	1		3	7	7	3		1	1	3	1	**1**								77.62
36								1	1	**4**	1	1		3	7	7	3		1	1	**4**	1	1								78.04
38								1	**2**	4	1	1		3	7	7	3		1	1	4	**2**	1								78.44
40								**2**	2	4	1	1		3	7	7	3		1	1	4	2	**2**								78.85
42								2	2	4	1	1		3	**8**	**8**	3		1	1	4	2	2								79.21
44								2	2	4	1	1		3	**9**	**9**	3		1	1	4	2	2								79.55
46								2	2	4	1	1		3	**10**	**10**	3		1	1	4	2	2								79.90
48							**1**	2	2	4	1	1		3	10	10	3		1	1	4	2	2	**1**							80.23
50							1	2	2	4	1	**2**		3	10	10	3		**2**	1	4	2	2	1							80.50
52						**1**	1	2	2	4	1	2		3	10	10	3		2	1	4	2	2	1	**1**						80.77
54						1	1	2	2	4	1	**3**		3	10	10	3		**3**	1	4	2	2	1	1						81.04
56	(S=2)					1	1	2	2	4	1	3	**1**	3	10	10	3	**1**	3	1	4	2	2	1	1						81.28

Buffers added during each iteration are shown in bold
2% downtime 90% bottleneck

Table 7.1 'Near Ideal' buffer pattern build up for a 29 machine line with a bottleneck

During this operation, the gradient across the bottleneck has increased and the bottleneck has once again become the constraint. At this point buffering is again placed either side of the bottleneck until it ceases to be the constraint and causes no additional disruption on the line i.e. the starvation curve either side of the bottleneck are level (Figure 7.9d). The cycle is again repeated, creating a bigger level portion step increase in the starvation at the bottleneck (Figure 7.9e). This is again levelled by buffering at the bottleneck (Figure 7.9f) before further levelling (Figure 7.9g).

The 'Near Ideal' buffer pattern for S=2 contained localized peaks and troughs. In order to test to see if these were as a result of local optima and/or the effects of random number streams, the pattern was tested against a smoothed pattern and with different random number streams. The smoothed pattern uses buffer spaces in similar places to the 'Near Ideal', the difference being that local peaks and troughs in the buffering shape are smoothed. The two patterns are shown in Figure 7.10. Using the smoothed pattern gave an output of 81.25% compared with the 'Near Ideal' patterns output of 81.53%. The 'Near Ideal' pattern has the greater output, and the difference is equivalent to the addition of 2 buffers (i.e. taking two buffers off the 'Near Ideal' pattern will result in almost identical outputs). Although the local peaks in the 'Near Ideal' pattern appear to be correct, when different random number streams were used, the output from the line with the smoothed buffer pattern was virtually unchanged (a 0.01% fall in output). When different random number streams were used on the line with the 'Near Ideal' pattern the output decreased by 0.36% to 81.17%. Although these differences appear to be small, they represent a 16% increase in time spent blocked or starved by the bottleneck machine. This implies that the 'Near Ideal' pattern was sensitive to the random number streams used, and although it gives the highest output in one case it is far from perfect. It does, nevertheless, give a good indication of the best buffering since the steps involved in producing the smoothed pattern from the 'Near Ideal' were relatively straightforward.

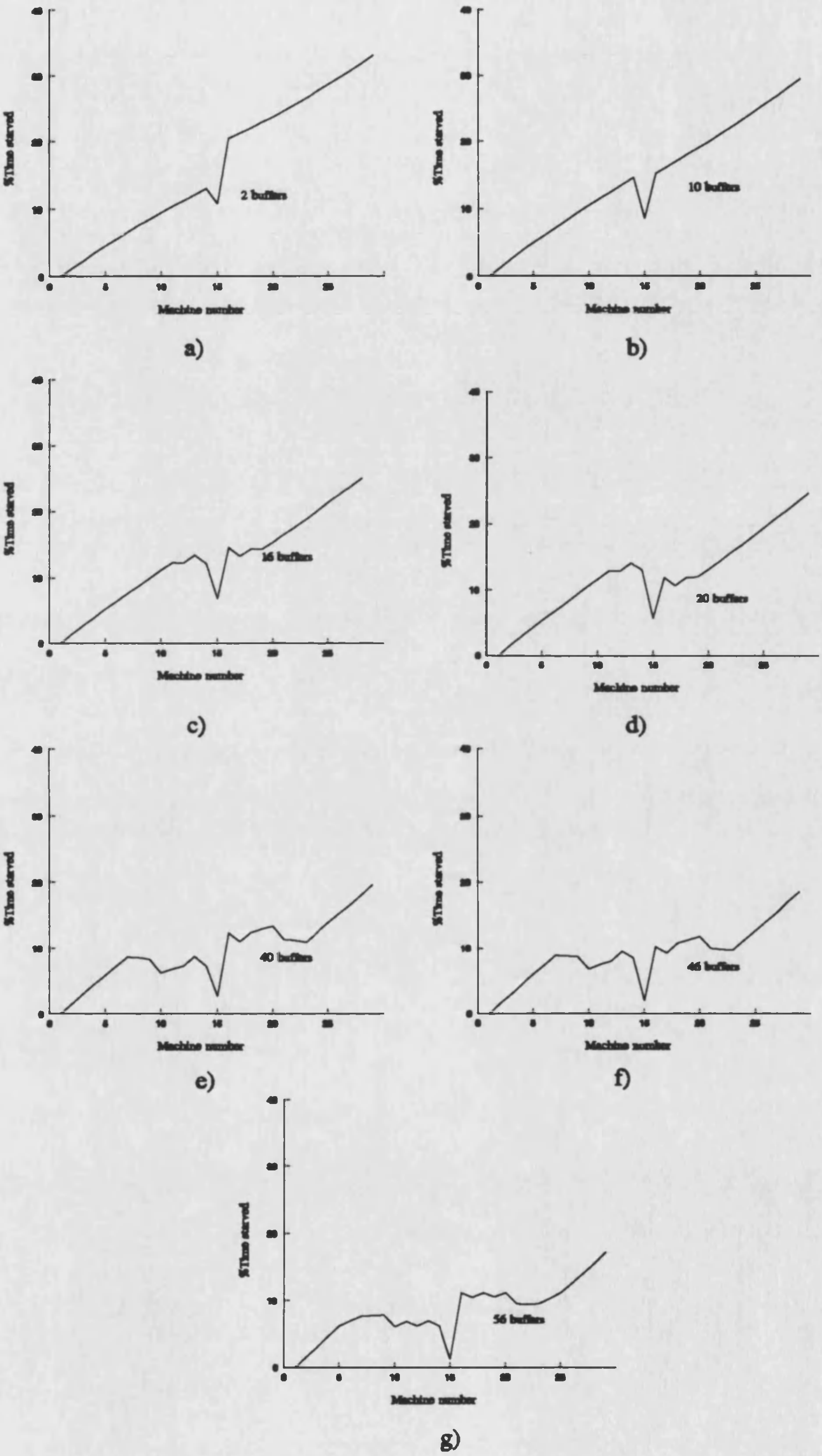


Fig. 7.9 Starvation curves for line with 'near ideal' buffering

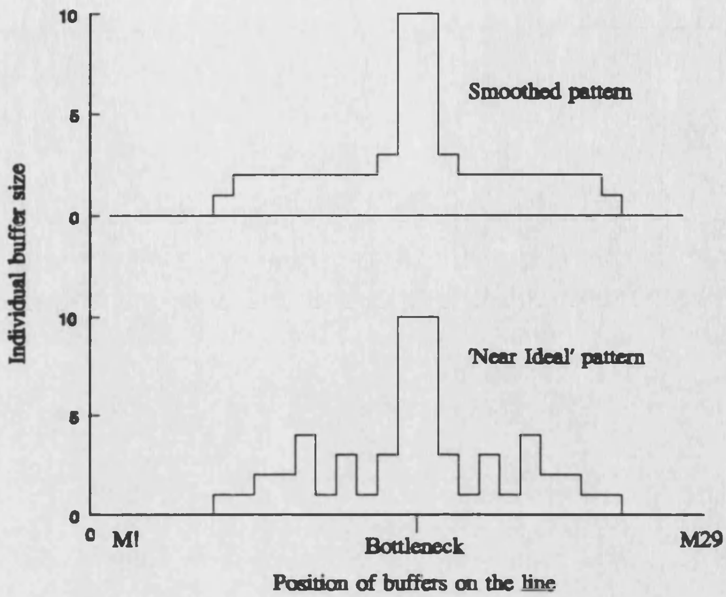


Fig 7.10 'Near Ideal' and smoothed buffer patterns for S=2

7.5.3 Other examples

In order to determine how line parameters affected the 'Near Ideal' distribution, the buffer patterns for other lines were found. Although it would have been preferable to study the results of several lines, where a similar level of buffering had been applied to that in the example in Section 7.5.2 (Referred to as Example 1), the excessive amount of simulation time precluded this. Instead, lines were simulated until a general 'feel' of how the buffering would build up was achieved. In most cases this was at a buffering level of approximately $S={}^1/_3$.

Consider a line of 29 machines with 4% downtime and a 80% bottleneck at the middle machine (M15). This model (Example 2) has an increase in %downtime from Example 1, with a corresponding decrease in the bottleneck size (the bottleneck machine has remained the same). The resulting build up of the ideal buffer is shown in Table 7.2. It can be seen that this model gives a far more spread out pattern than Example 1. Example 3 (Table 7.3) gives further evidence of this with more downtime (10%) and a smaller bottleneck (50%). The buffer pattern in this case is even more dispersed. The reason for this spreading out is twofold. Firstly, the bottleneck is relatively small. This means that it imposes less of a constraint on the line with a correspondingly smaller step in the starvation and blocking graphs, which in turn means that less buffers are required to make the rest of the line become the constraint. Secondly, the overall %downtime is higher. This means that there is a higher general level of interference and effects produced at the bottleneck are more easily interfered with before they affect the whole line.

This later point is borne out by Example 4, another 29 machine line with an 80% bottleneck but with a 0.8% downtime. The 'Near Ideal' buffer pattern for this line is shown in Table 7.4. Although the number of buffers placed on the line is small compared with the other examples, the overall trend can be seen. With the decrease in downtime there is a decrease in the number of stoppage effects to be interfered with, the need for a greater amount of buffering at the bottleneck to prevent it being the constraint is needed.

The need for general stoppage effects with which the bottleneck stoppages can interfere is also shown in Example 5 (Table 7.5). This is a line with the same machine parameters as Example 2, but it is only 9 machines long. In such a short line, fewer stoppage effects are present on the line at any one time, and those that are soon reach the end of the line (before being interfered with). The result is that interference levels are low and bigger buffers are needed to obtain the interference required for the bottleneck to cease being the constraint.

7.5.4 Conclusions drawn from example of 'Near Ideal' buffering

It is evident that there is a need to buffer the capacity constraint resource in order to gain the greatest improvement in output. In general on a bottleneck line, the bottleneck will be the constraint before any buffering is applied. After adding a given amount of buffering, however, the bottleneck is no longer causing additional starvation or blocking of other machines. This means it is no longer the constraint on output and as such any further buffering should be directed elsewhere on the line. In the cases examined the optimum place to put this buffering is in such a way so as to create, on the starvation curve, a level portion either side of the bottleneck. This in turn ensures that these machines are no longer the constraint. The reason the buffer is placed here is that this is where interference can be increased the most by the addition of buffers (The S-shaped starvation and blocking curve philosophy). And because of the tendency towards a level S-curve, this is where a level portion equating to no additional starvation and blocking (and thus no constraint) can be most easily generated.

The addition of this buffering to create the level portion, increases interference and hence output. The result of both of these facts is that less windows reach the buffers either side of the bottleneck, interference there is reduced and the bottleneck once more becomes the constraint. The cycle is then repeated.

From the above, the mechanism of buffering can be understood. The problem of how best to buffer each individual case is, however, still unresolved. Machine reliability, repair times, bottleneck size, line length and position will all effect the 'Near Ideal' pattern. To determine the 'Near Optimum' pattern for all different combinations of parameters would be a huge undertaking in computing time. Although the number of line parameter combinations could be reduced using some form of optimized experimental design (e.g. Taguchi's orthogonal arrays) the work could still possibly take years to complete and would really be pointless since the work would still be exclusively for lines where the bottleneck is the only machine to have a different downtime from all the others. In reality lines are far more complex, and although these results give useful pointers, they only scratch the surface of the problem.

Machine number
(buffer spaces are between machines)

Number of buffers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
2														**1**	**1**														
4														**2**	**2**														
6													**1**	2	2	**1**													
8													**2**	2	2	**2**													
10										**1**			2	2	2	2			**1**										
12									**1**	1			2	2	2	2			1	**1**									

Buffers added during each iteration are shown in bold
4% downtime 80% bottleneck

Table 7.2 'Near Ideal' buffer pattern build up for a 29 machine line with a bottleneck

Machine number

Numer of buffers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
2														**1**	**1**														
4											**1**			1	1			**1**											
6											1		**1**	1	1	**1**		1											
8										**1**	1		1	1	1	1		1	**1**										
10								**1**		1	1		1	1	1	1		1	1		**1**								
12								1	**1**	1	1		1	1	1	1		1	1	**1**	1								
14								1	1	1	1	**1**	1	1	1	1	**1**	1	1	1	1								
16							**1**	1	1	1	1	1	1	1	1	1	1	1	1	1	1	**1**							
18						**1**	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	**1**						
20						1	1	1	1	1	1	1	1	**2**	**2**	1	1	1	1	1	1	1	1						

Buffers added during each iteration are shown in bold
10% downtime 50% bottleneck

Table 7.3 'Near Ideal' buffer pattern build up for a 29 machine line with a bottleneck

Machine number
(buffer spaces are between machines)

Number of buffers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
2														**1**	**1**														
4														**2**	**2**														
6														**3**	**3**														
8														**4**	**4**														
10													**1**	4	4	**1**													

Buffers added during each iteration are shown in bold
0.8% downtime 80% bottleneck

Table 7.4 'Near Ideal' buffer pattern build up for a 29 machine line with a bottleneck

Machine number

Number of buffers	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
2														**1**	**1**														
4														**2**	**2**														
6														**3**	**3**														
8														**4**	**4**														
10														**5**	**5**														
12													**1**	5	5	**1**													
14													**2**	5	5	**2**													
16													**3**	5	5	**3**													
18													**4**	5	5	**4**													

Buffers added during each iteration are shown in bold
4% downtime 80% bottleneck

Table 7.5 'Near Ideal' buffer pattern build up for a 9 machine line with a bottleneck

Chapter 8 A Methodology for Buffering Transfer Lines.

It is clear that the method used in Chapter 7 to produce a 'near ideal' buffer pattern on a transfer line is far too time consuming to be applied to a real, complex line. Other methods of buffering could rely on the 'expert knowledge' of someone who has buffered other lines and/or studied the results of the buffering of simple lines such as those in Chapter 7. This would be another unsatisfactory solution to the problem since it would not provide a consistent answer, and would not be available for all to use. To overcome these problems, a methodology has been developed which allows a line designer to buffer the line by applying a set of rules. These rules are based on trying to ensure that the bottleneck machine is never blocked or starved by preventing the stoppage effects of other machines affecting the bottleneck. It is loosely based on constraint theory whereby the bottleneck is buffered until it is no longer the constraint on the line. Secondary bottlenecks are then highlighted and buffered. The buffering of the secondary bottlenecks causes the original bottleneck to become a constraint once more. This buffering cycle is then repeated until the output is maximised. In order to distinguish the buffer pattern produced using the methodology from other patterns, it is referred to as the '***Near Optimum***' pattern. Another technique to apply buffering to the line is also presented in this chapter. This is referred to as the '***Proportional Downtime***' pattern, and is described in Section 8.6.

The methodology has been developed with the aim of allowing line designers to place 'near optimum' buffers on a line which has no buffering. There is, however, no reason why the initial line set up can not have a given amount of buffering already in place. This provides three main advantages. Firstly, additions to existing lines can be planned where none of the existing buffering can be moved or removed. Secondly, when designing a line it may be that there has to be a certain level of buffering at a given point. Generally this would be due to physical constraints such as the shape of the line (For example, a long conveyor is needed to go round the end of a U-shaped line) or a given minimum distance between two successive machines. Finally, buffering already on the line reduces the number of steps required to add buffering to the line.

This reduces the amount of work on the part of the designer and reduces the simulation time, both of which help reduce the design lead time. The disadvantage of having buffering on the line before applying the methodology is that the buffers may not be in the most effective place. As such the buffering levels are increased for a given output. This gives an associated increase in cost both in terms of installation and in increased WIP. Further discussion on the cost aspects of buffering are presented in Section 8.4.

8.1 Outline of methodology

The methodology operates around repeated simulation runs of a model of the transfer line to be buffered. From the simulation runs results showing the percentage of time each machine spends *Blocked, Starved, Up and Down* are needed. After each run analysis is carried out on the amount of time spent blocked and starved and the methodology then identifies points on the line to be buffered. Buffering is repeatedly added at the point identified until the methodology indicates sufficient has been added. Following that buffering, the line is re-simulated and further points identified and buffering added. This cycle is repeated until the bottleneck machine is never blocked or starved or a pre-required output is reached.

Although the ideal methodology would be applicable to all lines, the following assumptions about the line are made :-

- Work is always available to the line, so machine 1 is never starved.
- Output from the line is always free so the last machine is never blocked.
- Parts travelling down the line visit all machines sequentially (i.e. there are no parallel machines or operations missed).
- Parts are never scrapped on the line itself.
- Buffer spaces are placed between machines, and once fixed they cannot be moved from one place to another.
- Machines are arranged such that buffers of any given size can be placed

between them.

• All machines have the same cycle time.

Although some of these assumptions impose constraints on the application of the methodology, it does allow the buffering to be developed in many situations. Discussion on the buffering of more complex lines (primarily with splitting and cycle time imbalance) is presented in Section 8.5.

8.2 Description of steps involved in the methodology

The flow chart in Figure 8.1 show the main steps in the methodology. To apply the methodology, the steps shown in the flow chart should be followed until either:-

a) Output is maximised (the bottleneck machine not blocked or starved), or

b) A required output level is reached, or

c) A maximum number of buffers to be used on the line is reached.

Throughout this an subsequent chapters, there are references to getting A/B= (said as "A B Level"). This refers to getting the values of starvation (or blocking) on the machines either side of a bottleneck equal (i.e. the points on a graph are level). A more detailed explanation on page 118.

The boxes on the flow chart are each numbered. An explanation of the method of application of each box and how any actions are to be carried out is given below.

Box 1 - *Simulate line and obtain blocking and starvation graphs* - Before any buffering can be added to the line a simulation model of the line must be constructed. The model built should be as realistic as possible (the Garbage In = Garbage Out cliche is very applicable). The time distributions and causes of machine stoppages (both breakdowns and tool changes) are particularly important.

Simulation runs using the model should be conducted such that results showing the amount of time spent Up, Down, Blocked and Starved are recorded. The latter two are

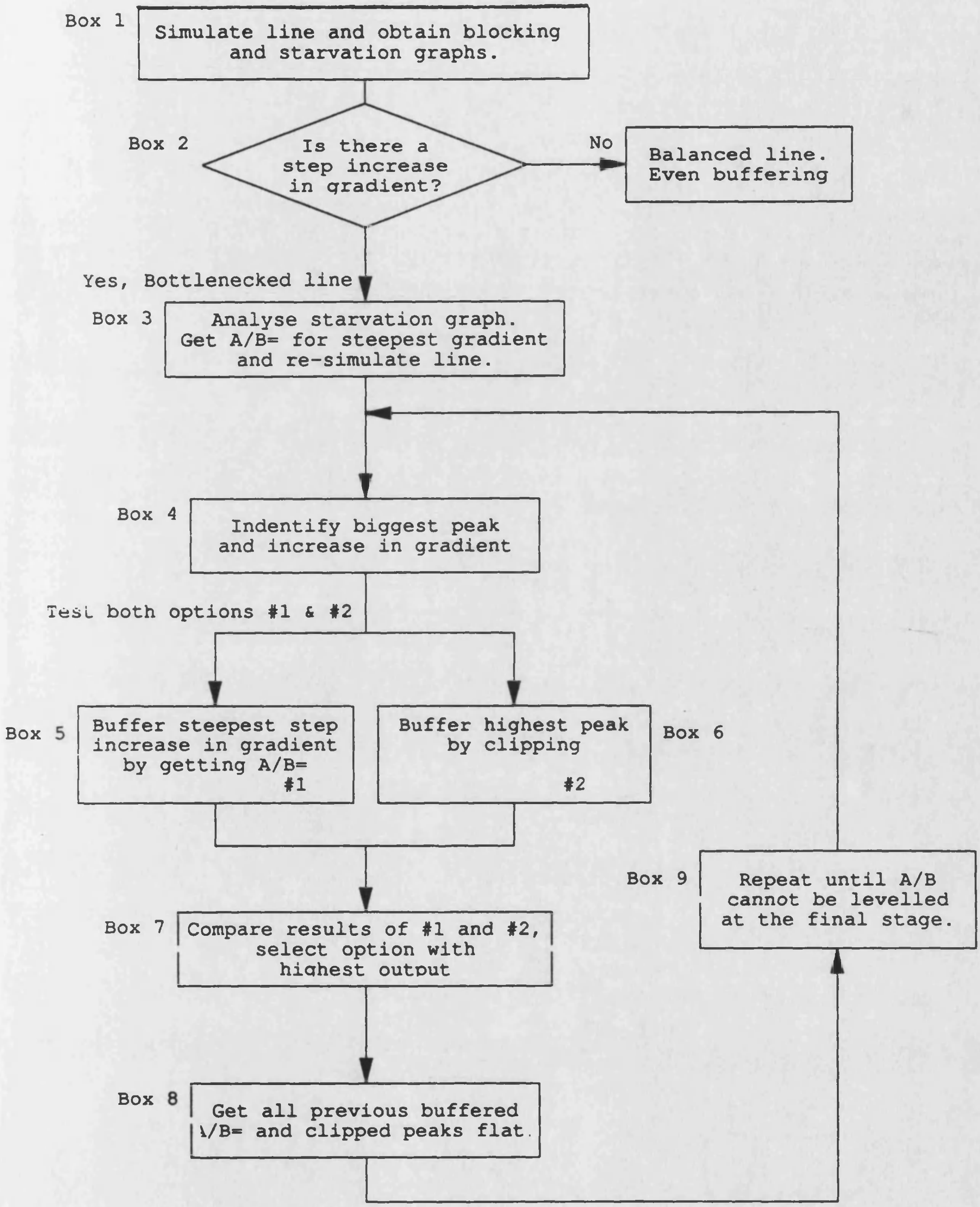
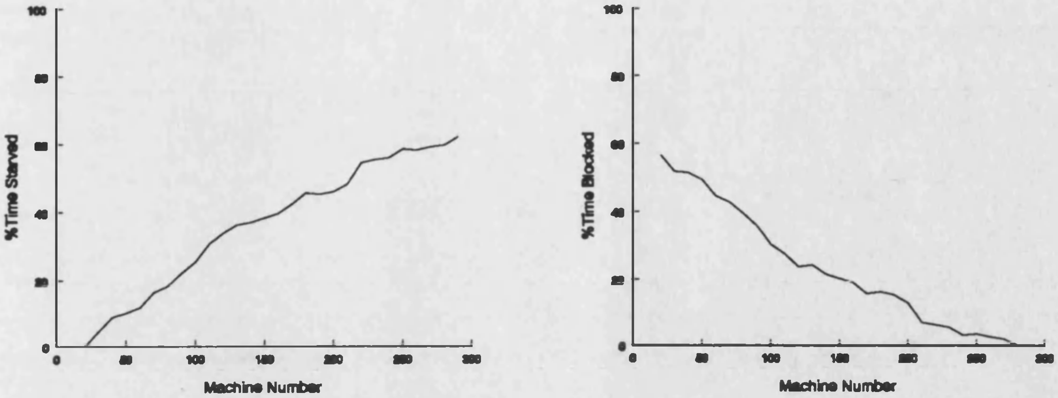


Figure 8.1 Flow chart of methodology used to place 'Near Optimum' buffer patterns on a transfer line



a) b)

Figure 8.2 Examples of Starvation and Blockage graphs for a complex line. (These graphs are the unbuffered line used in the case study in Chapter 9 and correspond to the figures in Table 8.1)

%TIMES FOR THE MACHINES					GRAPH GRADIENTS	
MACHINE NUMBER	STARVED	BUSY	BLOCKED	DOWN	STARVED	NEGATIVE BLOCKED
20	0.000	37.751	56.651	5.598	4.467	
30	4.467	37.752	51.915	5.866	4.464	4.736
40	8.932	37.750	51.615	1.703	1.157	0.300
50	10.088	37.749	49.498	2.664	1.830	2.117
60	11.918	37.748	44.723	5.610	4.577	4.775
70	16.495	37.747	42.825	2.933	1.749	1.898
80	18.244	37.746	39.446	4.564	3.925	3.379
90	22.169	37.745	35.712	4.375	3.212	3.734
100	25.381	37.743	30.394	6.482	5.559	5.318
110	30.940	37.742	27.663	3.655	3.002	2.731
120	33.941	37.741	23.791	4.527	2.366	3.872
130	36.307	37.739	24.330	1.624	0.868	-0.539
140	37.174	37.738	21.624	3.463	2.483	2.706
160	39.658	37.737	19.181	3.425	2.753	2.443
170	42.410	37.736	15.895	3.959	3.478	3.286
180	45.888	37.735	16.377	0.000	-0.485	-0.482
190	45.404	37.734	15.362	1.501	0.722	1.015
200	46.126	37.732	12.946	3.195	2.321	2.415
210	48.447	37.731	7.090	6.732	6.243	5.857
220	54.690	37.730	6.363	1.216	0.826	0.727
230	55.516	37.730	5.580	1.174	0.680	0.783
240	56.197	37.730	3.151	2.923	2.538	2.428
250	58.735	37.729	3.537	0.000	-0.419	-0.385
260	58.316	37.728	2.680	1.276	0.920	0.857
270	59.236	37.728	2.038	0.998	0.569	0.642
280	59.805	37.727	0.000	2.468	2.468	2.038
290	62.273	37.727	0.000	0.000		0.000

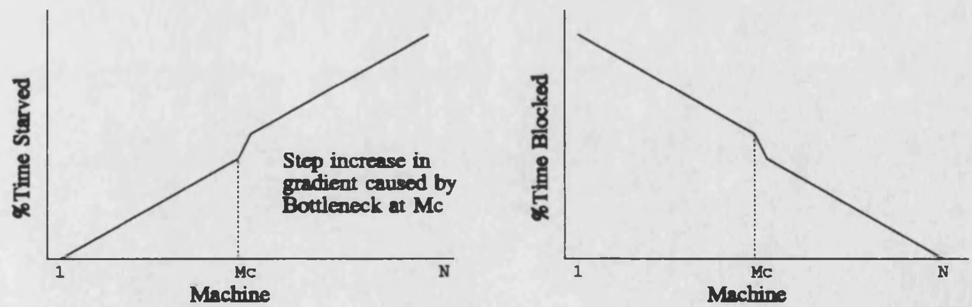
Table 8.1 Recorded data for a complex line
(unbuffered example of case study in Chapter 9)

used as a base for most of the analysis and should be displayed graphically (Examples in Figures 8.2a&b) and numerically (Table 8.1). The gradient of the starvation curve is also required and must be calculated (also shown in Table 8.1).

After any changes to the buffering on the line, the line must be re-simulated and the graphs and figures re-calculated as they are the basis for checking buffering already added (i.e Has sufficient buffering been placed at a given point?) and for planning further buffer additions.

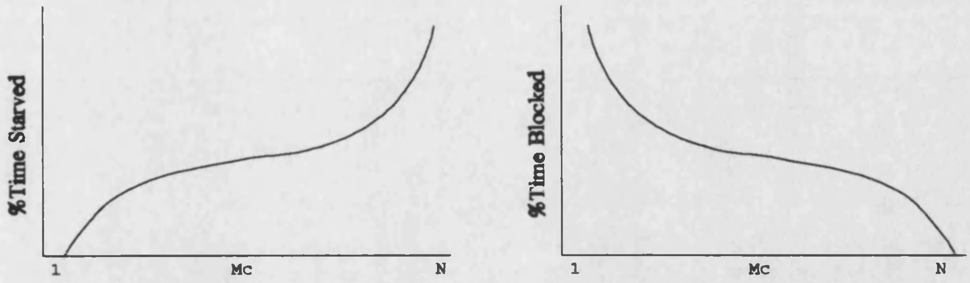
Box 2 - *Is there a step increase in gradient?* - A step increase in gradient indicates a bottleneck. Figures 8.3a&b shows simplified starvation curves for a lines with a bottleneck in the middle of the line. The gradient is not significant, it is the step increase that is important in determining the buffering.

If there is no step increase, as shown in Figures 8.4a&b, then the line should receive even buffering (i.e. equal buffers between each machine). It is, however, important to note that following a given amount of even buffering on a line, which initially appears not to have a bottleneck, a small bottleneck may be revealed which will be indicated by a step increase. If this is the case, it is necessary to revert to the main buffering methodology.



a) b)

Fig. 8.3 Simplified Starvation and Blockage graphs showing a step increase in gradient.



a) b)

Fig. 8.4 Simplified Starvation and Blockage graphs showing a balanced line (no bottleneck) with no step increase in gradient

Box 3 - *Analyze starvation graph. Get A/B= for steepest gradient and re-simulate line* - If step increases in gradient are found then determine the steepest step increase (this is associated with the primary bottleneck). This is then buffered until it no longer provides a constraint on the line. This is achieved by ensuring that the bottleneck machine provides no additional starvation down the line compared with the amount of time it itself is starved (i.e the gradient across the bottleneck is zero). By a similar reasoning, no additional blockages must be passed up the line.

Although in most cases the bottleneck machine (the machine with the greatest amount of downtime) will have the steepest step increase, this has not been proved. Two machines, with similar overall %downtimes but with vastly different distributions will have different gradients on the line. It is for this reason that the methodology deals with blockage and starvation times and not directly with downtimes which might give a different result.

The zero gradient across the bottleneck is achieved by placing buffers immediately before and after the bottleneck machine. These buffers are placed either side in proportion to the amount of time spent blocked and starved. In general, a bottleneck machine which is predominantly blocked has more buffers placed after it. This, however, need not always be the case as distributions of machine stoppage times can have a big effect.

The effect of this buffering on the starvation graphs is shown in Figure 8.5 (the effect on the blockage graph is the mirror image of that on the starvation graph). The amount of time the bottleneck machine (M_C) spends starved is indicated by the point C, with the amount of time that the machines immediately before and after M_C spend starved being indicated by A and B respectively. As buffering is added around the bottleneck machine (M_C), point A tends to rise slowly, while point B tends to fall more quickly. When the two points are level, the gradient across the bottleneck is zero and the bottleneck is not contributing to line disruption. In short hand this is written as A/B= (said A B Level). Although this type of graph shows the trend, it is difficult to determine the exact point that A/B becomes level. To do this it is necessary to study the amount of time the two machines either side of M_C (corresponding to points A and B) spend starved or blocked numerically. Obviously for the starvation curve to get A/B=, the time spent starved for A must be greater than or equal to the time spent starved by B, and vice versa for the blocking curve. It must be noted that A/B= must be achieved for both the blockage and starvation curve for any given bottleneck. If A/B is level in the starvation curve but not in the blocking curve, buffering must be added after the bottleneck machine, and vice versa. To get A/B= may take several iterations but this must be completed before moving on to other steps in the methodology.

There are some complications at this stage. In some cases A/B cannot be levelled as adding additional buffers have no effect to A & B (A & B reach a given point and cease to move). The reason for this is not fully understood but it is based around a lack of interference at the buffer. If this happens, the level of buffering placed at the bottleneck should be restricted to the minimum needed to reach the values for points A and B where they stopped moving. A further problem occurs when the steepest gradient is found on the first two or last two machines. In this case it is not possible to get both points A and B. The general method of buffering this situation is to add buffers around the machine concerned until it is no longer the steepest gradient.

Box 4 - *Identify biggest peak and increase in gradient* - Having got A/B level for the bottleneck machine, the next step is to identify the next biggest step increase and the biggest peak. The biggest step increase is defined in a similar way as described for

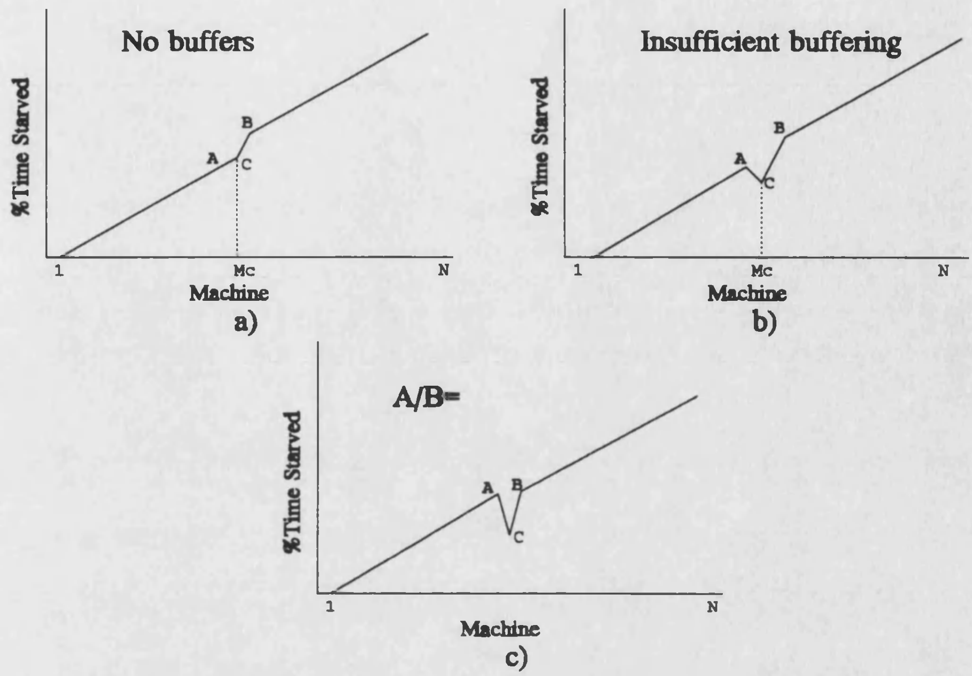


Fig. 8.5 Adding buffers at a step increase in gradient to get A/B=

Box 2. The only difference is that gradient from point C to point B in Figure 8.5c is not considered to be a step increase.

A peak is defined as the crest at point A in Figure 8.6a. The height of the peak (z) is the %time difference between points A and C. Figure 8.6b shows the height of a clipped peak (see below) is measured from the plateau to the bottom of the trough.

When finding the steepest gradient and biggest peak, again both the starvation and blockage graph should be considered.

Having determined the steepest gradient and biggest peak, the two options (buffering the steepest gradient and buffering the biggest peak) are then tested seperately and the one giving the highest output per buffer added is chosen - see *Box 7*. If there is no secondary step increase, then the only option is to buffer by clipping a peak.

Further developments of the methodology include making a decision about which of the two options is the better before trying them. This has not been tested, but is

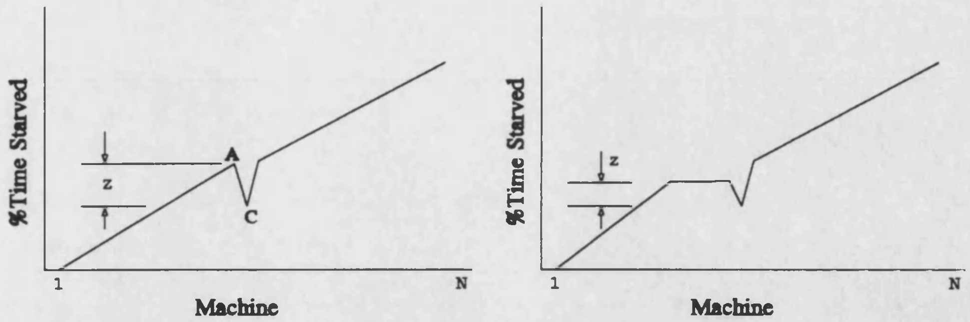


Fig. 8.6 The height of a peak on a starvation curve

discussed in Section 8.5

Box 5 - *Buffer steepest step increase in gradient by getting A/B= (#1)* - This is a repeat of *Box 2.*

Box 6 - *Buffer highest peak by clipping (#2)* - Peaks are clipped by placing even buffers incremental between machines next to the bottleneck which caused the peak. A peak in the starvation curve will result in buffering being placed in between machines before the bottleneck that was buffered to cause the peak. A peak in the starvation curve caused by a bottleneck at M_C means buffers should be placed upstream between M_{C-1} & M_{C-2} and M_{C-2} & M_{C-3} etc. A peak in the Blockage curve means buffers should be placed downstream of the bottleneck machine. The buffering should extend along the line until a flat plateau is achieved, as shown in Figure 8.7a. If the even buffering does not extend far enough from the bottleneck machine, the plateau will slope towards the bottleneck (Figure 8.7b). Figure 8.7c shows how the plateau slopes away from bottleneck if the buffering extends too far from the bottleneck.

The same peak may be clipped more than once. This is done by placing bigger buffers between the consecutive machines working away from the bottleneck machine. When this is done, the plateau will become bigger, extending further from the bottleneck, and as such requires more buffering.

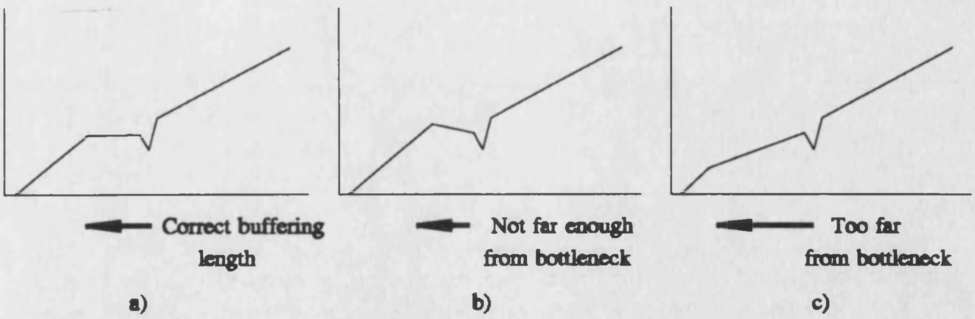


Fig. 8.7 The correct distance away from a bottleneck to buffer a peak

Box 7 - *Compare results of options #1 and #2, select option with highest output* - The better of the two options is selected by comparing the gains made with the number of buffers used i.e. The gain in output made for each option is divided by the number of buffers used. The option with the highest gain in output per buffer is then selected.

Short cuts can be made in testing the two options by adding an equal number of buffers to the line for both options at a part way stage (i.e. before A/B are completely level and before a flat plateau is achieved). The option with the highest output with the same number of buffers is then considered the best. Although the validity of using this short cut has not been thoroughly tested, an example of using it to reduce the number of simulations is given in the case study in Chapter 9.

Box 8 - *Get all previously buffered A/B= and clipped peaks flat* - Previous constraints that were buffered to ensure they did not limit output will, as a result of the buffering done on each iteration of the loop, no longer have A/B= or have flat plateaux. This means they may now constrain output once more. The last stage of the repeated loop of the methodology is therefore to get all A/B= and all plateaux flat.

This is done by adding buffers in the same way as described for *Box 2* and *Box 6*. The more complex the line, the greater effort that will be required to complete this stage. In the case study presented in Chapter 9, during a particular application of the main loop 120 buffer spaces were added, but to execute this further step of the methodology required another 130 buffers.

Box 9 - *Repeat until A/B cannot be levelled at the final stage* - The main loop of the methodology can be repeated until, in the limit, A/B can not be levelled for the bottleneck machine. The reason that they cannot be levelled is that point C has reached zero and output is maximised. When the buffering levels approach this point, the methodology can become unstable and large buffers can be needed for very small gains. When this point is reached, great care must be exercised.

In each stage where buffers are to be added, the methodology describes how buffering should be added until a given point is reached (either A/B= or a flat plateau). This implies that buffers should be added one at a time until this point is reached. In reality great short cuts can be taken. To get A/B= in a particular instance, add the buffers in say blocks of 10 (or more) spaces. This may quickly show that 20 buffers either side of the bottleneck are too few, but 30 are too many. If so then 25 buffer spaces can be tried, etc. By doing this, the number of simulation runs is greatly reduced, but the principles of the methodology are not compromised.

8.3 A simplified example of applying the methodology

The methodology for finding the 'near optimum' pattern has been applied to a line identical to that used as the main example for finding the 'Near Ideal' pattern. (Example 1 in Section 7.5.2). This is a 29 machine line, where each machine has a downtime of 2% with a 90% bottleneck on the middle machine, M15. This simplified example allows all the steps of the methodology to be demonstrated and enables its effectiveness to be seen. Figure 8.8 shows the principle starvation curves obtained during the application.

Stage 1 (Box 1) - The first stage is to build a simulation model of the line and obtain the results for the line without any buffers. This is shown in Figure 8.8 as *NO BUFFERS.* The output of the line is 61.20%.

Stage 2 (Box 2) - The step increase in the starvation curve due to the bottleneck can be clearly seen a M15. The line is therefore not balanced, and as such should not be buffered evenly.

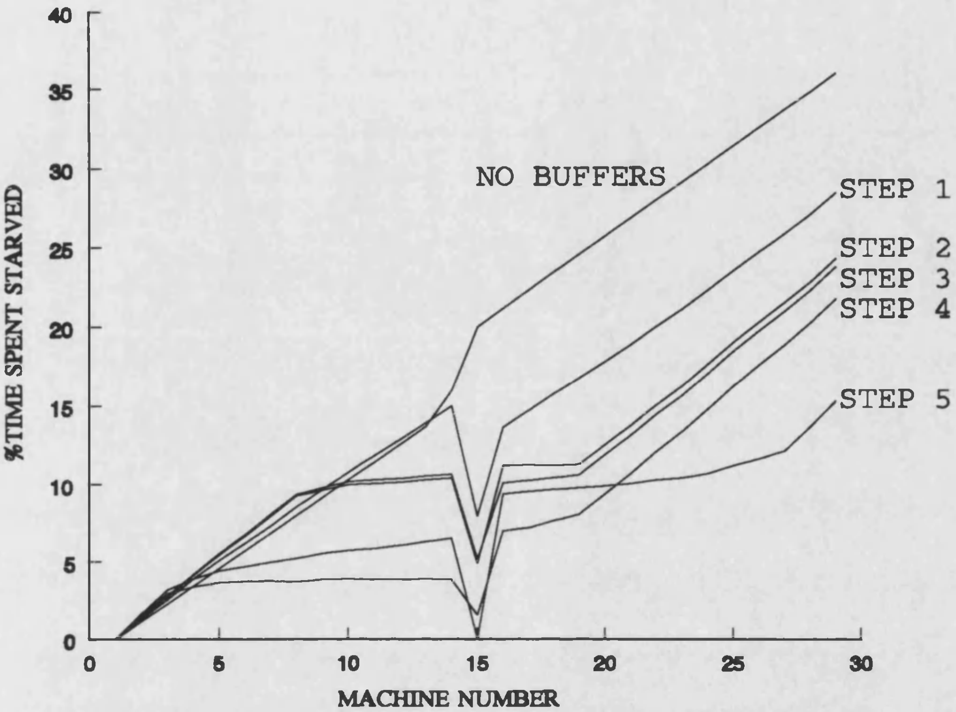


Fig. 8.8 The effect of the buffering methodology on the starvation curve.

Stage 3 (Box 3) - The line is buffered to get A/B=. This is achieved by placing 6 buffers either side of the bottlenecked machine, M15. The resulting starvation curve is labelled as *STEP 1* in Figure 8.8. The buffer pattern can be seen, together with the output level of Step 1 (70.2%), in Figure 8.9. Note that A/B= is within the limit of discrete additions and in such cases it is better to have A slightly higher than B than vice versa.

Stage 4 (Box 4) - As there are no other step increases in gradient, the next stage *must* be to clip the peak. The line is symmetrical, as such the peaks generated in the starvation and blockage graphs are equal. As this is the case, and to simplify this example, both peaks (in the blockage and starvation graphs) are clipped at the same time.

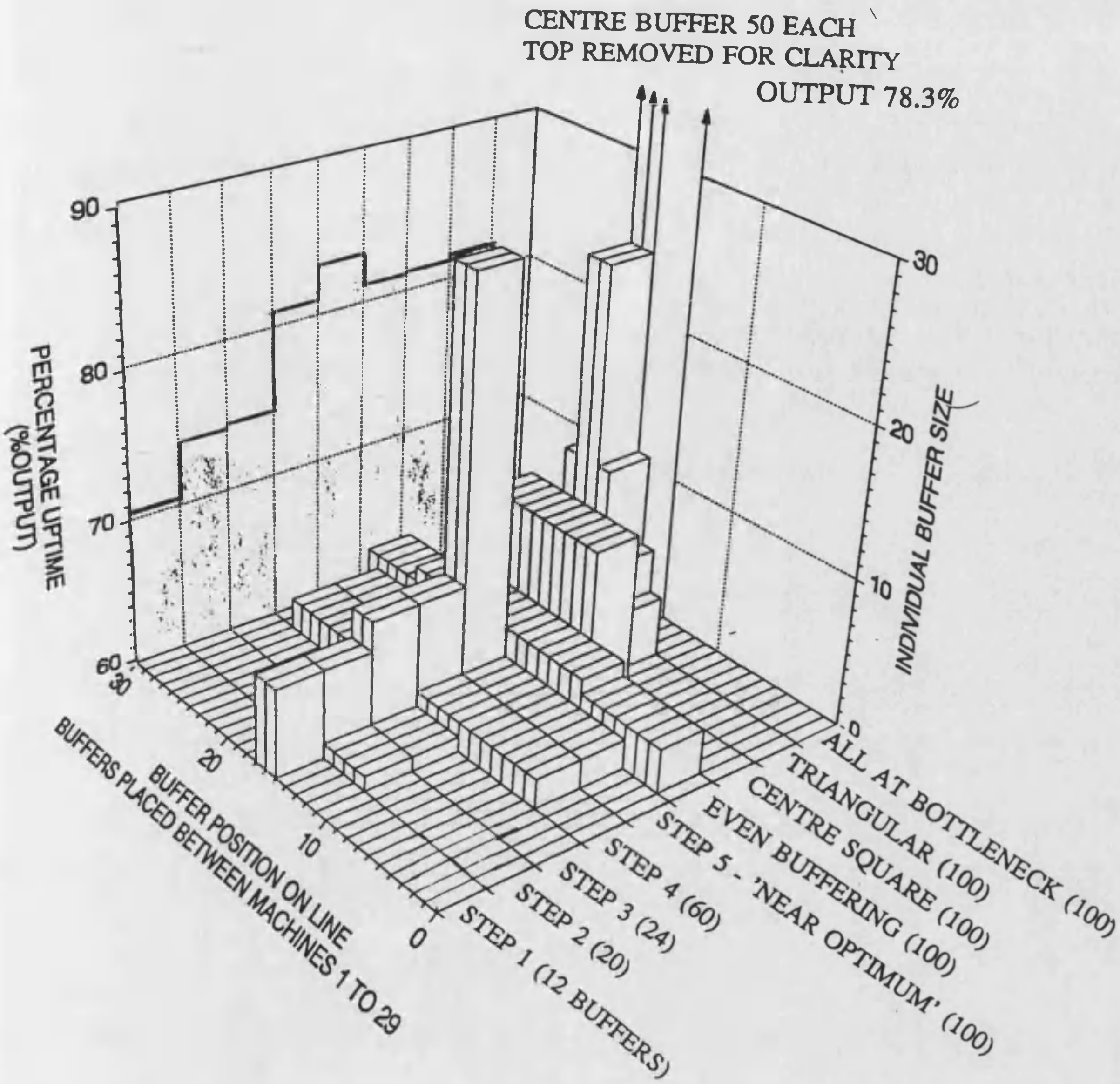


Fig 8.9 Graph showing the build up of the 'Near Optimum' buffer pattern, together with a comparison between the output of this and other strategies

Stage 5 (Box 6) - By conducting a series of incremental steps of adding one buffer at a time, both peaks are clipped by placing a series of 1 buffer spaces in between machines either side of the bottleneck. To achieve two flat plateaux, it was necessary to add 4 single buffer spaces either side of the bottleneck. The output rises to 74.0%

and the resulting starvation curve is shown in Figure 8.8 labelled *STEP 2*. The buffer distribution is 1 1 1 1 6 6 1 1 1 1 , and is shown in Figure 8.9.

Stage 6 (Box 8) - A/B is no longer level. The buffers next to the bottlenecks are increased from 6 to 8 spaces each to return A/B=. This is shown in Figures 8.8 & 8.9 as *STEP 3*. The output has risen to 74.8%.

Stage 7 (Box 4) - The main loop is now repeated, again there is no step increase in gradient so the additional buffering must be directed at re-clipping the peak.

Stage 8 (Box 6) - The peaks in the starvation and blockage curves are again clipped. The inter-machine buffer space size is increased to 2. This buffering extends from the gaps between M4/M5 to M13/M14, and from the gaps between M16/M17 to M26/M27. The output of the line has risen to 81.38%, this and the buffer pattern are shown in Figures 8.8 & 8.9 as *STEP 4*.

A point worth noting is that the total number of buffer spaces used is 60 to get an output of 81.38%. Compare this to the 'Near Ideal' pattern which had an output of 81.17% with 56 buffer spaces. The reason for the similarity in the outputs is that the buffer patterns are almost identical. Thus the results of the two techniques compare favourably.

Stage 9 (Box 8) - A/B must be got level once more by applying buffers either side of the bottleneck.

Stage 10 (Box 9) - While attempting to get A/B= by adding buffers in Stage 9, the amount of time M15 spends blocked or starved falls to zero. At this point the output of the line is maximised (83.4%) and no further buffering is needed. The total number of buffers placed on the line is 100. The starvation curve and buffer pattern are shown in Figures 8.8 and 8.9 respectively as *STEP 5*.

Also shown in Figure 8.9 are other buffer patterns using the same total number of buffer spaces. Note that with the 'even buffering', the total number of buffers does not divide up equally amongst the available spaces. The remainder have been concentrated towards the middle. The output using the methodology can be seen to give at least 1.5% higher output than any other pattern (83.4% compared with 81.8%). The lower output of 81.8% achieved using the triangular distribution can be achieved by using approximately 70 buffers using the methodology. Thus it can be said that for a given output, the buffering required by the best other distribution (triangular distribution in this case) is approximately 40% more than that required by using the methodology. To achieve this buffering on the line required 10 stages of thought corresponding to boxes in the flow chart in Figure 8.1 and only 5 iterative steps of adding buffers. Although each step of adding buffers could take up to 10 simulation runs, the maximum number of simulations is, however, only a fraction of those that would be required to place a near ideal buffer pattern on the line.

8.4 Implications of applying the methodology

The objective of the methodology presented in this chapter is to maximise output with the minimum number of buffers. Within the methodology there has been no consideration of the cost aspect of adding buffers to a transfer line. Although the examination of the cost aspect is an interesting and worthy pursuit it is beyond the scope of this project since the issues raised could easily be the base for another PhD thesis. Nevertheless it is important that some of the issues involved are raised and briefly discussed within the context of this thesis, so the reader is aware of the advantages of using the methodology together with the areas for further consideration.

Within the environment of line design there are two main factors, Output and Cost. The relationship between them is not, however, straight forward, with many parameters in the design process affecting both. The balance between these two factors is always company and product specific, with each case being different. As such a global solution is very difficult, if not impossible, to generate.

Although the methodology is designed to provide maximum output for the minimum number of buffer spaces, this is only one part of its application. As well as finding the correct pattern to maximise output, the methodology can be used to rearrange buffers on existing lines in order to increase output, or even to allow lower buffer stocks to be used on new lines to achieve the same proposed output. Thus it need not be a method for only applying buffers, but a tool to help the line designer determine the correct buffer strategy as part of the line design process.

The effects of applying the methodology are difficult to assess since each case is different. Benefits made through buffering to increase output are only of use if the extra products made are saleable. If extra products are not saleable, however, the methodology is equally applicable as it allows the line's buffering to be reduced and hence reduce cost.

Much is said about the disadvantage of buffering as it increases WIP. This increase in WIP goes against some of the trends in manufacturing which advocate lower stock levels (Lean Manufacturing and JIT for example). These philosophies, however, are mainly designed for application in the batch environment. Consider the example of Ford in the mid 1980's. Their sales of the Sierra model in Europe were restricted by the number of engines that could be built. This in turn was dependant on the number of engine blocks machined on a transfer line. The line was installed at a cost of £50m. From the figures in the case study in Chapter 9, it is estimated that the increase of the overall line buffer level from 750 to 1250 buffer spaces would yield a 5% improvement in output. This is equivalent to 20,000 units/year, which Ford believed they could sell. Each engine when built is worth approx £1000, and is placed in a car typically costing £10,000. The potential increase in turnover from adding the buffers to the line is thus in the order of £200m. This is all based on having an extra 500 buffer spaces. A very generous estimation of the installation costs would be in the order of £500k. If a profit margin of 5% of turnover is assumed, this gives a payback period of less than 20 days.

The interesting figure, however, is the increase in the cost of WIP. Assuming the buffers are always full (a very unlikely worst case scenario), the additional cost of stock at each buffer space is approximately £15 each. The increase cost of stock is therefore less than £7500. Studying these figures, the advantages of having the extra buffer stocks can be clearly seen, but possibly more importantly is the fact that the cost of the increased WIP is negligible.

A final point to be discussed is that the buffering is only one parameter that must be balanced when designing a line. Others such as tooling and machine costs also affect output. The question that is raised is, that if a line costing £100m with an output of 50% can be buffered using the methodology (much of which might be rearranging existing buffers as well as the more obvious addition of more buffering) to give 60% output, could (assuming line output and capital costs are proportional) a line costing £90m with an output of 40% be buffered using the methodology to give an output of 50%. If this is the case, restructuring the buffers (including adding some more buffers at a relatively low cost) could lead to cost reduction of up to £10m when installing the line, a significant saving. Although it is interesting to postulate this, without detailed figures the answer is unknown and the discussion is beyond the scope of this work. Nevertheless it could, however, be suggested that the reason that the £90m+buffering option is not more widely used is that most, if not all, line designers are unaware of the correct method of buffing the line. If this is the case then potential savings through improved line design by applying the methodology are substantial.

8.5 Shortcomings of the methodology and possible improvements

There are three main areas in which the methodology may be improved or expanded. They are:-

a) Modifications to the methodology to allow easier application in its current form,

b) Development of the methodology to encompass all machining type lines,

c) Development of the methodology for application to other flow environments.

Although there is some overlap between these areas, they are more easily discussed separately. The ideas presented in this section also overlap with the comments made on further research presented in Chapter 11.

8.5.1. Modifications to improve methodology in its current form

The robustness of the methodology has not been thoroughly tested. Although tests on simple lines show the methodology to be effective, there has only been one study of the methodology's effect on an actual line (presented in Chapter 9). A particular area for further testing is the technique of clipping the peaks for lines with a level of imbalance in the downtime of the non-bottleneck machines.

Before the methodology can be claimed to be fully effective further tests on real lines are required. There are, however, problems in conducting such a study. Any work would rely on accurate data on existing lines being available. Although some manufacturers (e.g. Ford) do have simulation models of their lines, these models are far from accurate as the data on which they are based is generally unrepresentative. For example, the Ford model used in Chapter 9 takes no account of stoppages for tool changes. Also a new line being modelled before being built must have any machine data based on estimates, although this can be based on similar machines on other lines.

There are, however, problems which need resolving before such a study might be undertaken. The question of how best to buffer a bottleneck if it occurs at the first two, or last two machines, on the line remains unresolved. Although in the methodology it has been suggested that they are buffered until they are no longer the steepest gradient, this has not been proved as the best option. This problem could best be solved by finding the 'Near Ideal' pattern for both a bottleneck at the first and at the second machine. Doing this should highlight the best strategy to use. The problem in doing this is that the time required to complete such a simulation based study would be a minimum of 3 months.

There are two further possible improvements to the current methodology. Although both of these are felt to be worthy additions, neither have been tested in any form.

The first concerns *Box 4* in the flow chart, where it is necessary to test the two options of clipping the highest peak and buffering to get A/B= for the steepest step increase. Using the Theory of Constraints, it is suggested that the two options can be compared numerically as they are both constraints on the line. Thus if the gradient of the biggest step increase (i.e. the numerical size of the constraint of the step increase) is greater that the height of the biggest peak (i.e. the numerical size of the constraint of the peak) then the step increase should be buffered, and vice versa. Again thorough testing of such an amendment to the methodology would be required. If this hypothesis were found to be true, it would make the application of the methodology far easier to execute and it would also help to reduce the number of simulation runs required.

The second possible amendment is aimed at improving accuracy but this may be at the cost of increasing the number of simulation runs required. The idea is that having found the correct buffer pattern to maximise output, the resulting starvation curve is used as an input for a second iteration. This would mean that the methodology would be repeated and instead of getting A/B= for a given point, A and B would be buffered until the gradient across them was the same as the gradient of a smoothed curve for that point in the first solution. An example is shown in Figure 8.10. It can be seen that the gradient between A/B is equal to the mean gradient of the starvation curve at that point. This idea is mere postulation, without any foundation at all, except for the experience gained through developing the methodology. The gains made from adopting this technique are assumed to be very small and the increase in the number of simulation required is large since the methodology is effectively executed twice.

8.5.2. Improvements to allow application to all machining lines

The application of the methodology in its current form is limited since the lines to which it may be applied are restricted. The two main obstacles to its application to all machining type transfer lines are:-

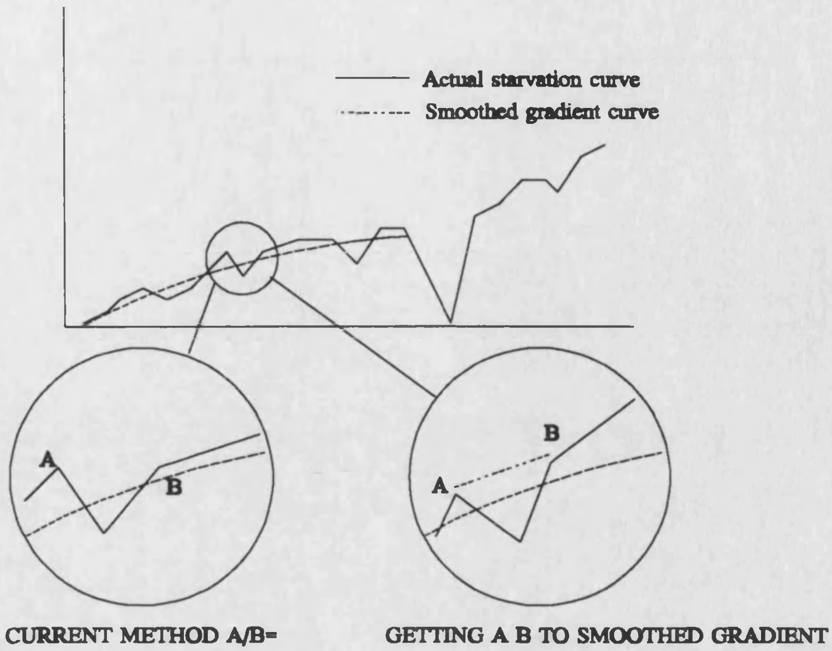


Fig 8.10 A possible alternative to getting A/B=

Need for lines to be balanced in terms of cycle time - There are two problems within this section. Firstly cycle time variation at a given machine and secondly different cycle times from machine to machine (i.e. Imbalance). Where varying cycle times are concerned, without buffering, these can cause substantial starvation and blocking. This can, however, be greatly reduced with relatively low levels of buffering. Thus it is felt the methodology can be applied to local imbalance, provided caution is exercised as small buffering levels may cause dramatic increase in output, so that A/B= and clipping peaks may be achieved with comparatively less buffering than might be expected. This idea, however, has not been tested, so although it is felt that it is applicable there is no proof.

The problem of different machine cycle times is more complex. The basis behind the methodology is that the output of the line can never be better than the least reliable machine. This is true on a line with balanced cycle times. If, however, cycle times are different this need not be the case. An unreliable machine may have a faster cycle time. If so, the least reliable machine may not be the bottleneck. To overcome this difficulty the up time of the machines must also be considered. Identifying the primary and secondary bottlenecks is the major problem. When buffering such a line, the aim is still to minimise the time the bottleneck machine spends blocked and starved. A complication comes when machines have excess capacity, as these will always have a greater amount of time spent blocked and starved than other machines. Examination as to how the methodology performs in these situations, and how it should be modified to overcome any difficulties encountered, has not been carried out. It is felt that further study in this area would be a worthy and interesting area of work.

Parallel machines (Split lines) are not accommodated - The problem of parallel machines is very complex and has not been studied in any detail. There are three main cases, which are shown in Figure 8.11. In the most simple case, if a machine has insufficient capacity, a similar machine can be placed in parallel to achieve the required capacity (Figure 8.11a). In general these machines will not be the bottleneck, but this is not always the case. This situation is difficult to resolve as either or both machines can be stopped causing differing starvation on the line. One possible solution relies on a satisfactory method of modelling the two machines as a single machine being developed.

The problem of a single parallel machine can be further complicated by extending it to a series of parallel machines. This can take two forms. Firstly, the parallel machines all feed to and feed from the same buffers. This situation can be considered as a series of single parallel machines (Figure 8.11b). The second situation is where the parallel machines form two separate lines (Figure 8.11c). The solution to this problem is more complex as there is inter-machine buffering along the sub-lines as well as at both ends. No proposed solution as to how to deal with this situation is presented.

There are other minor problems which may arise on particular lines. Since there are such a diverse range of problems, these would have to be considered on an individual basis. An example is two machines that cannot have a buffer placed between them. In this case the machines are under intermittent control. As such they can be considered as one machine with two heads and the stoppage times that are used in the simulation can be thus modified.

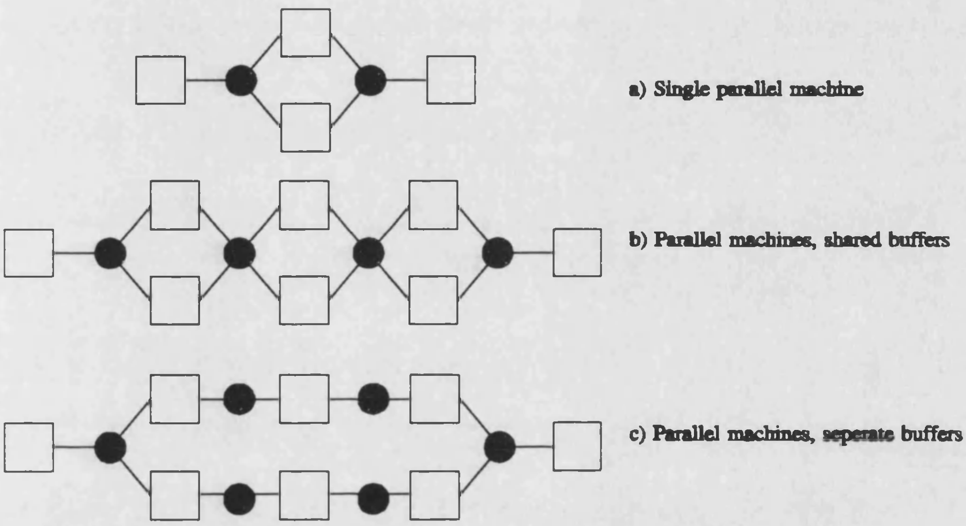


Fig. 8.11. Different formats for parallel machines

8.5.3. Application of methodology to other flow productions.

The methodology has been developed for flow type, discrete product, linear production. There is, however, no reason why the ideas used in developing the methodology can not be applied to other environments so that either modifications to the existing methodology or new methodologies are developed. The two main areas are :-

Continuous Flow Processes - This can be separated into true continuous flow processes such as steel mills and discrete part flow processes such as canning plants were the cycle time is so small that the production is considered as a continuous flow.

In steel mill type operations buffers consist of variable length accumulators.

The constrains on the design of the line are determined by the physical dimensions of the steel being moved between operations. The weight and size of the product are such that buffering levels are low. Also with such few operations buffers would have to be very large to have much effect. In such an environment, the applicability of the methodology is limited.

A different picture is found in canning and bottling plants where large stocks of components are frequently used between machines. It is not uncommon to find swirl tables holding several hundred units between each operation. The cost of a swirl table is much lower per unit than a comparable buffer space on a machining type line and with such low product value, the cost of the WIP is negligible. Thus the large buffering levels needed due to the small cycle times (normally less than 0.5sec - CMB are currently installing a canning line in Carlisle to operate at 900 units per minute) can justifiably be used. In this environment there is scope for the application of the methodology.

Assembly lines - There are two main problems preventing the methodology being applied to assembly lines. The first is that due to the typically high proportion of manual operations, there tends to be a high degree of variation in cycle time. The problems associated with this were discussed in Section 8.5.2. The second problem is that the lines are tree shaped. Although downstream blockages will affect machines upstream in a similar way, starvation effects are markedly different. Although a stoppage at a machine upstream effects all machines downstream of it, it does not starve machines in parallel streams which it will join later. Instead machines in parallel streams become blocked as they cannot cycle because the machines downstream do not need there parts. Development of the methodology for application in this area would be another interesting area of research.

A further area where the methodology may be applied is away from the typical manufacturing environment. It was stated in Chapter 3 that the flow of products can be modelled using fluids, where machines are pipes, breakdowns are taps or processes and buffers are tanks. The question to be considered is could the roles be reversed whereby the methodology developed for the manufacturing environment could be applied to fluid systems to determine optimum tank sizes?

8.6 The 'Proportion-Downtime' buffer pattern

To test the effectiveness of the buffer patterns produced using the methodology described above, line outputs must be compared with similar lines using the same amount of buffering, but in a different pattern. In Chapter 7, the 'Near Ideal' pattern was compared against Even, Square and a Triangular buffer patterns (see Figure 7.7). These patterns are rather simplistic and ineffective (as was shown in Chapter 7), so a better 'guess' for the correct buffering was developed to enable more realistic comparisons to be made. This is referred to as the 'Proportional-Downtime' buffer pattern. Although this pattern will later be seen to be reasonably effective, it must be remembered that it was developed from an educated guess by the author using an element of expert knowledge and experience gained during the course of this research work.

The 'Near Optimum' buffer pattern is based on a knowledge of the retardation of starvation and blockage effects moving along the line. Since the size of any stoppage effect is governed by individual machine downtimes, the basis for the 'Proportional-Downtime' pattern of buffering lines was that the buffer size should be proportional to the downtime of the adjacent machines. Allowance must, therefore, be made for both the machine before the buffer (to compensate for starvation of the machine and retard blockage effects from it) and the machine after the buffer (vice versa).

8.6.1 The buffering technique.

Consider a line n machine long (M_1 to M_N), where each machine's %Downtime is D_n (i.e Downtime of M_N is D_N), where each machine is separated by a buffer B_n of capacity BC_n

i.e M_1 - B_1 - M_2 - B_2 - M_3 - B_3 - M_4 - B_4 - ....- B_{N-1} - M_N

The total buffer capacity, BC_T is given by :

$$BC_T = \sum_{0}^{N} BC_n$$

and the total amount of downtime D_T is given by

$$D_T = \sum_{1}^{N} D_n$$

The capacity of a given buffer BC_n is given by

$$BC_n = \frac{BC_T \, . \, D_n}{2 \, . \, D_T} + \frac{BC_T \, . \, D_{n+1}}{2 \, . D_T} = \frac{BC_T}{2 \, . \, D_T} (D_n + D_{n+1})$$

This equation only gives an approximate answer. Manual adjustment to the buffer levels must be made because the above equation allows for buffers before the first machine (BC_0) and after the last machine (BC_N). Also the values BCn will not necessarily be integers so a degree of inspection and movement of buffers is required.

8.6.2 An example of the application of the 'Proportional-Downtime' technique

Consider a line of 5 machines where the downtimes are as follows :-

D1 = 10% D2 = 10% D3 = 20% D4 = 15% and D5 = 5%

Where a total of 240 buffers are to be placed on the line. Using the equation given in Section 8.6.2, the resulting buffer sizes are :-

BC0 - 20   BC1 - 40   BC2 - 60   BC3 - 70   BC4 - 40   BC5 - 10

The buffers at B0 and B5 are not required so their buffer spaces are spread proportionally to the other buffers, giving the final buffer sizes as :-

BC1 - 46   BC2 - 69   BC3 - 79   BC4 - 46

The simplicity of this approach can clearly be seen in this example. This example was, however, chosen so that the buffers were integers. A problem arises when the output of the integer division in the equation in Section 8.6.2 has a remainder. If this is the case, as it would normally be, the remaining buffer spaces must be reallocated in as close to a proportional manner as possible.

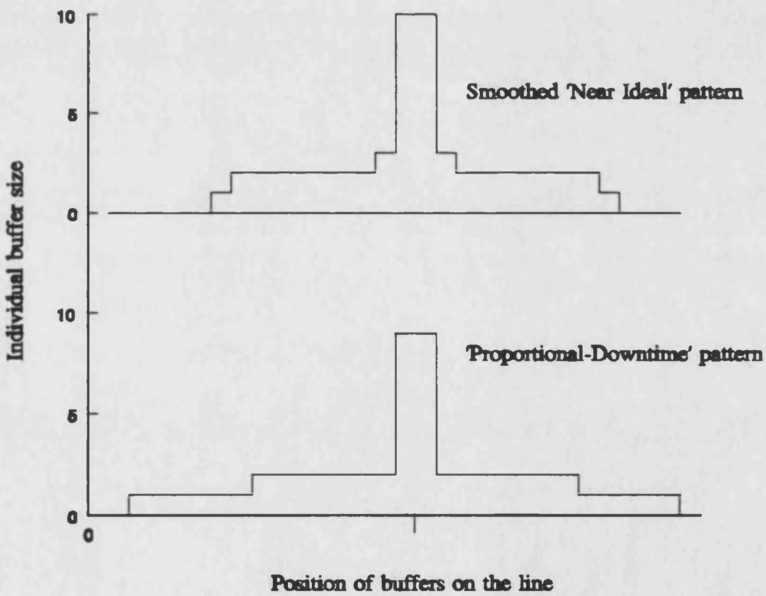


Fig. 8.12 Smoothed 'Near Ideal' and 'Proportional-Downtime' buffer patterns

8.6.3 Application of the 'Proportional-Downtime' to other lines

If the 'Proportional-Downtime' technique is applied to Example 1 in Chapter 7 (a balanced line except for a bottleneck at the middle machine) using a buffer level of S=2, the resulting buffer pattern has a similar general shape to the Smoothed 'Near Ideal' pattern. The two patterns are shown in Figure 8.12. The output of the line, however, is very much reduced. For the 'Proportional-Downtime' technique the output is 80.8% which does not compare favourably with the Smoothed 'Near Ideal' output of 81.3%. Indeed the 'Proportional-Downtime' techniques's output is more akin with those from the Even, Square, and Triangular distributions shown in Figure 8.9.

8.6.4 Implications of using 'Proportional-Downtime' buffer pattern

This buffering technique offers several advantages over the 'Near Optimum' methodology. They include:-

- Simple and easy to apply.
- Fewer simulations are needed.
- Results achieved are comparable for simple lines and complex lines with low buffering levels.
- Can be used as a good first approximation.

It does, however, have two major disadvantages:-

- The technique takes no account for different distributions of machine stoppages. It is quite conceivable for two machines to have the same %Downtime, yet have markedly different stoppage length and frequencies. The effect of different stoppage frequencies on interference levels and consequently output was shown in Section 6.2. Since the effectiveness of a buffer is determined by its ability to increase interference, the most effective pattern must also take this into account. Consider a 19 machine line which is balanced in terms of cycle time and downtime, but the middle machine (M10) has less frequent, longer stoppages (by say a factor of 10). The 'Proportional-Downtime' technique would buffer the line evenly, but looking at the starvation curve in Figure 8.12 of a buffered line, there is clearly a step increase in gradient. The 'Near Optimum' methodology identifies this feature and buffers accordingly.

• The failure of the 'Proportional-Downtime' technique to fully adapt to the line results in lower outputs when it is used on complex lines with large amounts of buffering (S > 20 in the case study in Chapter 9) when compared to the 'Near Optimum' methodology.

Even though the 'Proportional-Downtime' technique suffers these disadvantages it does represent a good first estimate of the expected output from a complex line before the 'Near Optimum' methodology is applied. As such it is a useful tool for buffering as well as meeting its original purpose of being a useful educated guess comparison to the 'Near Optimum' methodology.

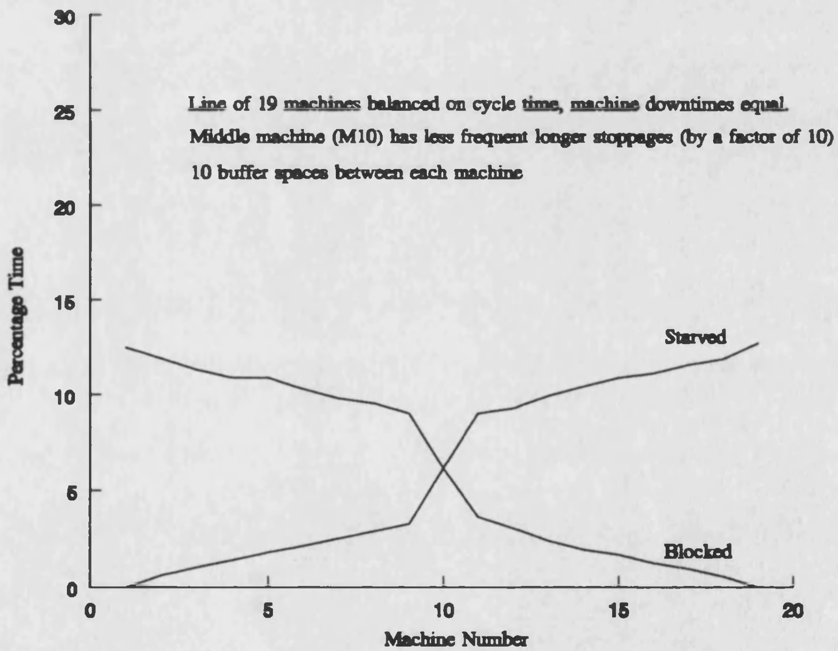


Fig. 8.12 The effect of different stoppage distributions on a line with balanced downtime

Chapter 9 - Case Study : An Example Transfer Line.

The model presented in this case study is based on the work of Crosby and Murton [1990] who conducted a simulation study of the proposed Ford Zeta engine block line. The study was part of a series of studies conducted by Ford to determine the feasibility of various layouts of their new line. Crosby and Murton's work was concerned with the correct representation of breakdown data within the simulation model.

A schematic diagram of the line showing the machines and conveyors is shown in Figure 9.1. The line consists of 33 machines operating under non-synchronous control. They are linked by queuing conveyors whose length varies depending on the required buffer capacity between the machines. The length, and associated buffer capacity, in term of the maximum number of components is also given in Figure 9.1. The length of conveyoring (i.e. the total buffer capacity) was 809 spaces.

9.1 Application of methodology

All simulation models used during the case study were run for a start up time of 1 weeks production (equivalent to 30,000 cycle times) before results were take over a ten week period. These simulation run times are twice as long as those used by Crosby and Murton who had very short initial start up times to allow buffers to reach a representative stock level.

9.1.1 Line simplifications and assumptions

Certain features of the simulation model mean the line must be modified before the methodology can be applied. These, together with other assumptions, are as follows:-

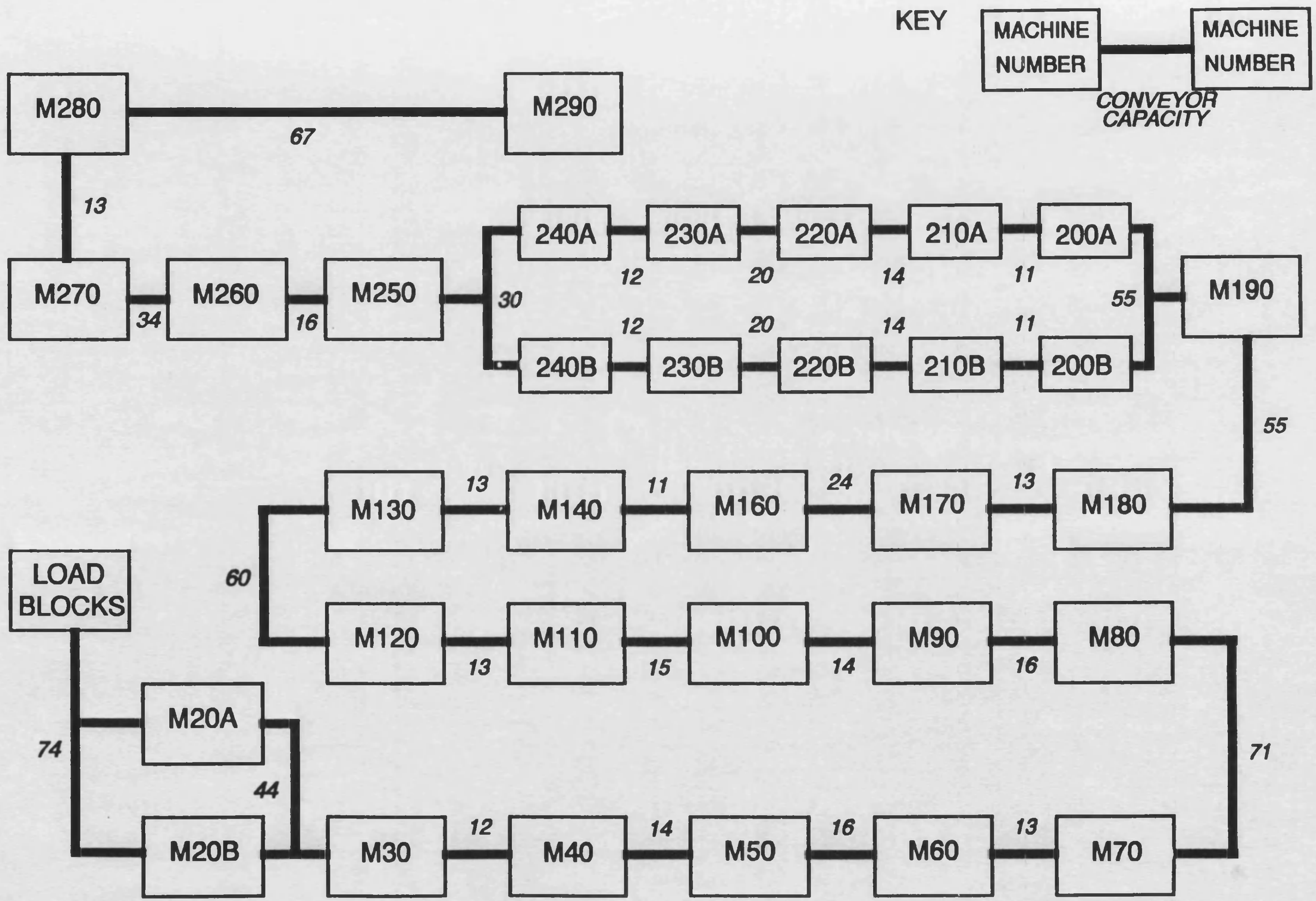


Fig. 9.1 Schematic layout of Ford Zeta I engine block machining line

Assumptions

• It is possible for buffer spaces to be added to the line by extending conveyors so that the maximum queuing length is equivalent to the required number of buffer spaces. This is not a totally satisfactory arrangement because there is a cycle time associated with each buffer space, thus a buffer of 150 components results in a conveyor cycle time of over 7 minutes. The problems with such an arrangement were discussed in Section 7.2. To overcome this problem, it is assumed that the maximum conveyor length is 20 components. Subsequent buffering is added to the line as a buffer store, at the output from the conveyor (i.e. between the end of the conveyor and the input of the next machine). The length of 20 components was chosen as it was felt that this represented a realistic maximum distance that machines could be separated in order to restrict the 'footprint' of the line. It is interesting to note that if buffers and conveyors are placed on the line in this way, the output is higher than if long conveyors are used (59.0% compared with 58.3% for 898 buffer spaces).

• The stoppage times and frequencies for breakdowns used in Crosby and Murton's model were determined from Ford's own historical data from a production line in Valencia, Italy. This same data is used for the application of the methodology.

Modifications

• It was stated in Chapter 8, that the methodology can not be applied to lines which have splitting. The Ford Zeta line has both a single parallel machine (M20A & M20B) and a length of split line (M200A&B to M240A&B). Where this is the case, the operations have been combined to enable the line to be represented by single machines. Thus M200A & M200B have become M200. The effect on line output is discussed below.

• There are further problems with the model due to cycle time imbalance. With the exception of the last machine (M290), this was only on the parallel machines which were represented as balanced machines when they were combined into single machines as described above. For the last machine, the cycle time was increased to create a balanced line. This change did not affect the results because the last machine can not be blocked (an infinite demand from the line is assumed) and it does not have any stoppages and as such it does not play an active role in determining line output.

• The first operation in Figure 9.1 is the loading of parts to the line. This operation is also considered to be 100% reliable. As an infinite supply is assumed, this operation was not modelled. This means the first conveyor is not used, resulting in the total number of buffers on the line being reduced to 735. Parts are fed directly to M20.

The last two modifications to the line are treating either end of the line differently and are as a result of using the inherited model developed by Crosby and Murton.

At the beginning of the line an operation which has no affect on the line is removed, while at the end of the line it is left in place. Although this arrangement may seem inconsistent it was considered that line output was not affected as neither operation was assumed to have stoppages. This has not, however, been proved.

The result of these modifications to the original model of Crosby and Murton's was to decrease the output from 58.5% to 56.45%. The difference in these values represents the loss of accuracy through making the simplifying changes. These changes would be significant if it were an actual change in line output. It is however, the difference between two dissimilar lines and as the output levels are similar, the effect of the modifications can be said to be small. As such it is felt that the methodology can be

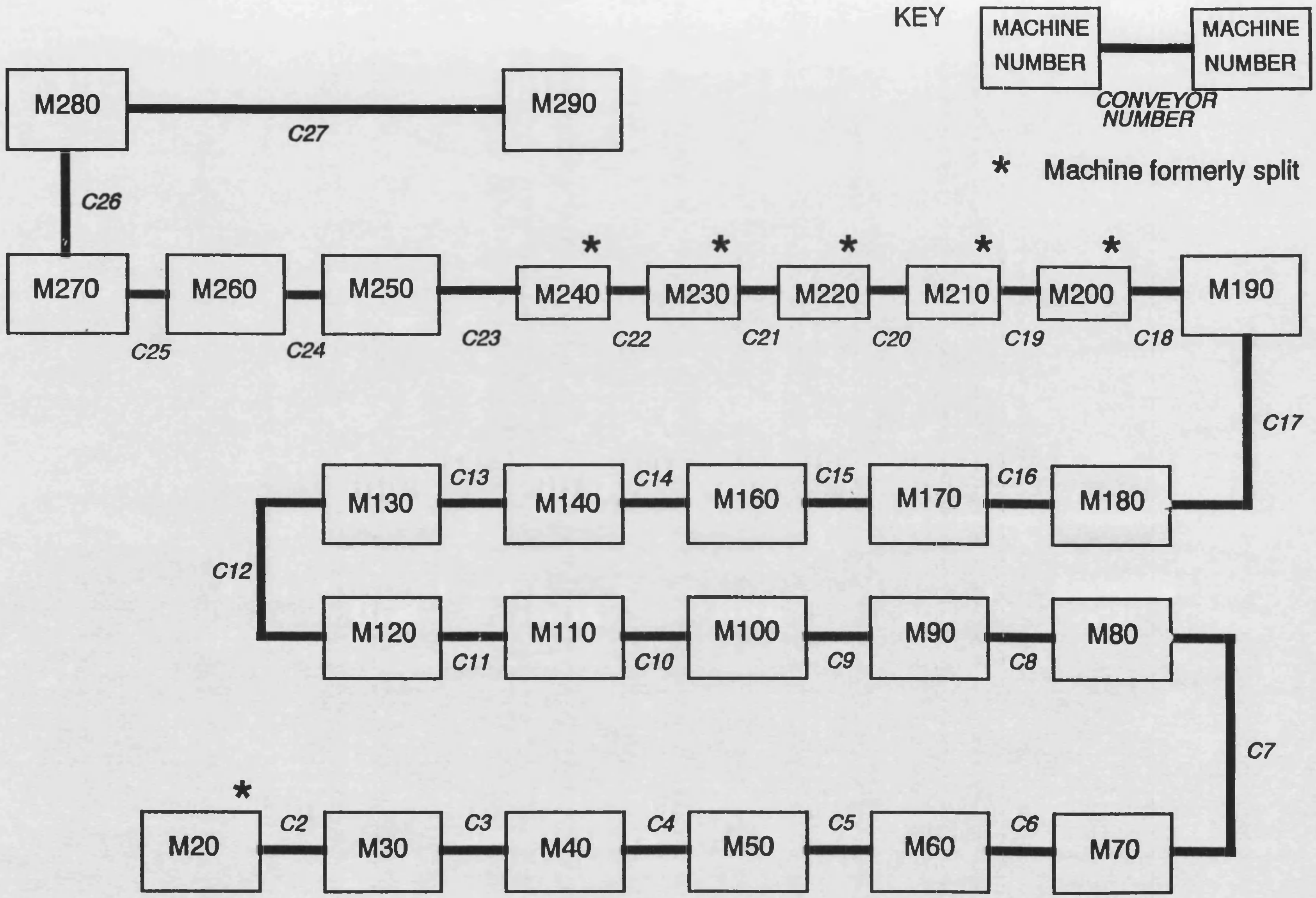


Fig. 9.2 Schematic layout of simplified Ford Zeta I engine block machining line for application of methodology

satisfactorily tested using real data on the modified line. The resulting buffering pattern, however, is for the modified line and does not represent the correct pattern for the original line.

A schematic drawing of the modified line is shown in Figure 9.2. The conveyor names used in this diagram correspond to those used in the description of the steps involved in the methodology.

9.1.2 Steps involved in buffering

The methodology has been applied to the modified line. Although several iterations were required to complete each stage, the results of these iterations are not presented. To gauge the simulation time required, to achieve the 9 stages presented below, the number of simulations carried out was 23. Six of these were, however, to complete the steps from Stage 1 to Stage 2. This is because it takes time to gain experience of the affect of adding a given amounts of buffering. Stages become progressively easier since better educated *guesses* can be made.

The simulation results together with the starvation and blockage curves for each stage are given in Appendix III. The buffering levels as a result of each iteration of the methodology together with the output level from the line are shown in Table 9.1. It must be remembered that only the first 20 buffer spaces of any one point are on a queuing conveyor, and all extra buffer spaces are in a true buffer.

The steps are described as follows:-

- **Stage 1** - The line initially has 1 buffer space between each machine. The simulation results show the greatest step increase in gradient is found at M210. (Output 38.8%)

• **Stage 2** - Since M210 spends a greater time starved than blocked (48% and 7% respectively), the buffering required to get A/B= will tend to be greater before the machine than after. With buffering of 75 spaces before and 50 spaces after A/B was not level in either curve. Addition of further buffers either side of the machine did not, however, result in an improvement in output nor did it make A/B level. The results of the increased buffering (90 before, 60 after) are given in Appendix III as Stage 2a. As a result of this the application of the methodology continued without A/B=, as provided for in the methodology, with the buffering of 75 buffer spaces before M210 at C19, and 50 buffer spaces after M210 at C20. (Output 43.3%)

• **Stage 3** - The effect of buffering the next steepest step increase in gradient and the biggest peak now need to be compared. Buffering the biggest step increase in gradient (at M100) gave a higher output than clipping the biggest peak (at M210). To get A/B level required 55 buffer spaces before M100 (at C9), and 65 buffer spaces after (at C10). Note that the majority of buffering is placed after the machine even though the machine spends more time starved than blocked and also the level of buffering either side of M100 is far more even than that for M210. This latter point is because there is less difference between the amount of time spent starved and blocked for M100 (24% and 23% respectively). The line output has now increased to 49.5%.

• **Stage 4** - By achieving A/B= for M100, at M210 A/B are now further displaced from level than they were in Stage 2. In order to return A/B= at M210 required the buffering to be increased to 100 spaces before (C19) and 90 spaces after (C20). (Output 49.8%)

• **Stage 5** - The result of getting A/B= at M210 has caused A/B at M100 to no longer be level. In order to get A/B= for both M210 and M100 level, the buffering at both had to be increased to:-

80 / **M100** / 65   140 / **M210** / 90

which is in the form Buffer capacity / **Machine Number** / Buffer capacity

The resulting line output is 51.3%, this is a 2.7% increase in output over Stage 3, where A/B= was achieved for M100 in the first instance.

• **Stage 6** - The effect of buffering the next steepest increase in gradient and the biggest peak must again be compared. Again interim tests showed that the gains in output by buffering the steepest step increase in gradient were greater than those for buffering a peak. The step increase is at M60, and to get A/B= required buffers of 100 to be placed either side at C5 and C6. (Output 55.7%).

• **Stage 7** - Returning A/B= for all the step increases that have so far been buffered, required the buffers at M100 and M210 to be increased as follows:-

105 / **M100** / 75 160 / **M210** / 100

The line output has increased to 56.6%. A point worth noting is that no additional buffers were needed at M60. This is unexpected, since the line performance has increased. This is explained by the fact that points A/B for M60 were taken beyond being levelled at Stage 6 (i.e. the was a negative gradient in the starvation curve across M60), and although these points moved during Stage 7 the gradient across them did not become positive (i.e. they were still considered level). Thus it can be said that a slight excess of buffering was added to C6 & C7 at Stage 6.

• **Stage 8** - Returning through the main loop of the methodology, again the options of clipping the biggest peak or getting the next A/B= must be considered.

Considering the two options in detail:-

Clipping the peak : The biggest peak is in the blockage graph at M210. Applying an extra 11 buffer spaces to each of the next three conveyors downstream (C21, C22 & C23 - 33 in total) gives an output of 55.5%. The results of this are shown in Appendix III as Stage 8a. The placing of 11 buffer spaces to each of the conveyors does not exactly follow the methodology, in that each buffer should have only been increased by one space. Unfortunately, adding one space at a time is very time consuming and differences are difficult to detect, so a larger amount of buffering was added to produce a flat peak.

Step increase in gradient : The two biggest step increases in gradient are both of similar magnitude, and are on the first two machines on the line (M20 & M30). As M20 is the first machine in the line, there is a problem in determining the correct buffer pattern. The best guess advised in the methodology description in Chapter 8 is to place buffering around this machine until it is no longer the steepest gradient. Since the gradient of M30 was a similar magnitude, buffers were placed either side of M30 in an attempt to address the buffering of M20 and M30 in one go. Thus 18 extra buffer spaces were added before M30 (C2) and 17 after (C3). The total number of buffers was, therefore, the same as Stage 8a. The resulting output was 55.9%. This is 0.4% higher than Stage 8a, with an increase in output from the line of 0.75%. These results are given in Appendix III as Stage 8b.

• **Stage 9** - In order to get the gradients of M20 and M30 to no longer be the steepest, the number of buffer spaces at both C2 & C3 had to be increased to 120. (Output 60.2%).

Although the output of the line has not been maximised, the simulation was halted at this point and no more buffering was added to the line. This is because slightly more buffers have now been added to the line than Ford originally used, and the output of the line exceeds that of the Ford line. Comparisons between the two buffer strategies can, therefore, be made.

It must, however, be noted that the use of this case study in demonstrating the execution of the methodology is limited because the steps involved are all aimed at getting various A/B=. At no point has there been any clipping of peaks. The effectiveness of the methodology in improving output, however, is clearly demonstrated by the results.

9.2 Results

The effect on output of buffering on the line can be seen in Figure 9.3. The maximum output obtained is 60.2%, which is achieved by adding 898 buffer spaces on the line. Note this is not the maximum output because the bottleneck machine (M100) still spends approximately 35% of the time either blocked or starved.

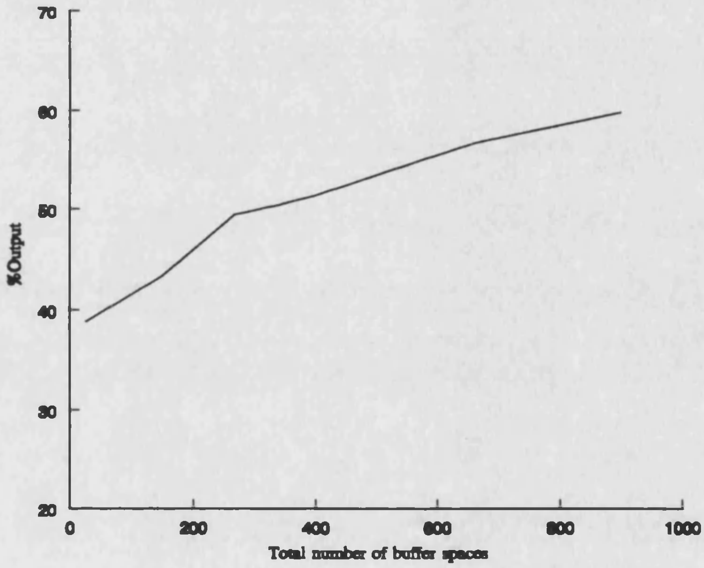


Fig. 9.3. The increase in output by adding buffers to the Ford Zeta line

Although not strictly valid due to the line modifications (splitting and cycle time balance), it is interesting to compare the results of the line buffered using the methodology with the output of the line with Ford's buffer pattern. The output of the modified Ford line is 56.4% with 735 buffer spaces.

By iterating between the results obtained when applying the methodology, the output of the line buffered using the methodology is estimated to be 57.8% with 735 buffer

Stage	Number of buffers	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	Output (%)
		Conveyor number (buffer spaces are equivelent to length)																										
1	26	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	38.87
2	149	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	**75**	**50**	1	1	1	1	1	1	1	43.33
3	267	1	1	1	1	1	1	1	**55**	**65**	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	49.46
4	332	1	1	1	1	1	1	1	55	65	1	1	1	1	1	1	1	1	**100**	**90**	1	1	1	1	1	1	1	50.35
5	397	1	1	1	1	1	1	1	**80**	65	1	1	1	1	1	1	1	1	**140**	90	1	1	1	1	1	1	1	51.31
6	595	1	1	1	**100**	**100**	1	1	80	65	1	1	1	1	1	1	1	1	140	90	1	1	1	1	1	1	1	55.71
7	660	1	1	1	100	100	1	1	**105**	**75**	1	1	1	1	1	1	1	1	**160**	**100**	1	1	1	1	1	1	1	56.63
9	898	**120**	**120**	1	100	100	1	1	105	75	1	1	1	1	1	1	1	1	160	100	1	1	1	1	1	1	1	60.24

All conveyor are one space unless stated

Table 9.1 'Near Optimum' buffer pattern build up for modified Ford Zeta engine block machining line spaces. Thus moving the buffing from the Ford pattern to that obtained by applying the methodology would result in an increase in output of 2%. Using Ford's estimation of a capacity of 500,000 units per year, this increase in output is equivalent to 10,000 units per year.

Alternatively, if the required output is 56.4%, it can be estimated from the result that the number of buffers required to achieve this using the buffer pattern obtained from the methodology is approximately 635, 100 fewer buffer spaces. This can be expressed as a 15% reduction in the number of buffer spaces required on the line.

Comparisons between the 'Near Optimum' methodology and other buffering patterns can be made. Taking the final result of the case study where the output is 60.2% with 898 buffers as a base, the two main comparisons are:-

Even buffering - As 898 buffer spaces do not divide evenly between the 27 conveyors, each buffer space was either for 33 or 34 components. The output of the line is 57.3%, this is a 5% fall in actual output (as opposed to the difference in the %outputs) compared with the line buffered using the 'Near Optimum' methodology.

'Proportional-Downtime' buffering - During the first stages of using the methodology to buffer the line, the 'Proportional-Downtime' technique yielded higher outputs. With 397 buffer spaces on the line (Stage 5 in the methodology), the 'Proportional-Downtime' technique gave an actual output increase of 3.5% over the 'Near Optimum' methodology (53.12% compared with 51.32%). As buffering levels increased, however, the performance of the 'Proportional-Downtime' technique declined until at the final buffering stage of 898 buffer spaces, the output using the methodology was higher (60.2% compared with 58.8%). These results confirm the comments made in Section 7.2 concerning the efficiency of the 'Proportional-Downtime' technique with high levels of buffering. It is expected that if further buffering were placed on the line, the difference in output as a result of using the two different buffer patterns would increase. This has not, however, been proved. No further comparisons have been made with other buffer strategies.

If the buffering pattern produced is examined, the buffering is all concentrated around 4 machines, with just one space between all the others. The biggest buffer is 160 components (a conveyor for 20 and a buffer of 140 spaces). If all the buffering had been in the form of conveyoring, this represents a very long conveyor whose empty throughput time using the conveyor speed adopted would be equivalent to 22 machine cycle times. As a way of confirming the comments made in Section 7.2 concerning the inefficiency of long conveyors as buffers, a model was run with the buffering all as conveyors. The output fell form 60.2% with buffers and conveyors to 58.3% with conveyor buffering only. This is a significant fall in output through what initially appears to be just a question of the format of the type of buffer. The subject of the relative cost efficiency has not, however, been considered.

The effect of random number streams, which was found to be significant in the 'Near Ideal' buffer pattern solutions has also been tested within the framework of this case study. The change in output was found to be an increase/decrease of less than 0.2%.

Iteration of the methodology was stopped at 898 buffers because of the uncertainty surrounding the buffering step increase at the ends of lines (the first machine in this case). The gradient of the starvation curve at M20 and M30 has clearly been decreased by the addition of the two buffers of 120 spaces. At present, however, it is unclear whether this is too many or too few, and therefore the buffering of the line has not continued.

9.3 Conclusion

The output of the line modelled can be significantly increased by changing the buffer pattern from that used by Ford to that developed using the methodology. The output from the line using this buffering was also found to be higher than other buffer strategies. Experiments showed that the use of conveyors as buffers is less efficient that the use of true buffers

The execution of the methodology to place 898 buffers on the line took less than 25 simulation runs. This compares favourably with the development of other patterns (e.g. the 'Near Ideal'), but is still significantly more than required for the 'Proportion-Downtime' technique. Substantially more iterations would, however, be required to maximise the output of the line.

It must be remembered that the buffer pattern developed could not be strictly applied to the real line because of the modifications made to the line with regard to the splitting and the lack of balance. Nevertheless, the line buffering developed by the methodology would be a valuable contribution towards the line design.

Chapter 10 - Conclusion

A substantial amount of work, using several approaches, has been done in the past concerning the design of transfer lines to improve output. Although much of it has been concentrated on the role of buffering in improving output, this body of work contains little in the way of guidelines as to how to buffer transfer lines which a line designer could use.

Previous research had tended to be analytically based, and due to the complex mathematics involved had been unable to deal with long lines. Some of the resulting ideas for the design of lines, such as the 'Bowl Theory', are completely impractical. Using these facts as a base the aim of the work was to develop a methodology with which designers could correctly and easily derive the optimum buffer pattern for a given line.

Using simulation to study relatively long lines containing up to 30 machines, an understanding of the mechanism of line stoppages was gained. This included the concept of 'Window Interference', where broken down machines cause starvation downstream in the form of a time window which can interfere with blockage effects of other stoppages. This interference results in a rise in output. Studying the movement of windows and the mechanism of window interference led to a different explanation as to the way buffers worked. Rather than mere stores of components, they should be considered as a means of arresting the movement of starvation windows and blockage effects. This leads to greater levels of interference and consequently higher output.

The clearer understanding of the operation of buffers allowed the development of a methodology to buffer lines based around constraint theory. By studying the amount of time each machine is spent blocked and starved it is possible, through a series of iterative steps, to determine the correct buffer pattern for a line.

When tested on simple lines, the resulting buffer patterns yield a higher output per added buffer space than any existing strategy that was tested. A case study using a complex line with real machine data based on a Ford Motor Co. line also showed the methodology to yield a buffer pattern more effective than any other, including the one proposed for use on the line. The increases in output gained by using the buffer pattern generated using the methodology has been found to be of the order of 2.5%. This is typically equivalent to needing approximately 20% less buffering to achieve a given output if the methodology is used.

The application of the methodology to the case study line could, however, not be fully completed because the methodology has not been fully developed to deal with all the features found on a complex line. Thus further work is needed in certain areas which have been identified. Primarily this concerns further testing of the methodology and its extension to cover such complexities as split lines.

The patterns developed using the methodology greatly contrast with those suggested by other authors. Indeed the results show that few, if any, of the general pointers suggested by other authors would give a buffer pattern resembling the patterns developed using the methodology.

The level of buffering to apply to a line will always be a complicated matter. There is still no global answer to such questions as can a reduction in machine costs of £10 million be compensated for by spending £5 million on buffering to achieve the same output. The ultimate goal of having a complete set of design rules to produce the most cost effective line is a long way away. Nevertheless it is hoped that the availability of a methodology that allows the correct buffering of a line to be developed relatively quickly will encourage these questions to be answered earlier in the design process. It is hoped that this work will prove a useful step towards the attainment of the goal of a complete set of design rules in the future.

Chapter 11 - Further Work

The first major development of the methodology that can occur in the future is its testing and use in real situations. This could take the form of a project undertaken by an individual who collects details of various lines from their users. Collation and publication of the results could then lead to the use of the methodology by others.

An alternative strategy would be the dissemination of the methodology to the end users. They would then be able to apply it directly to their own environments. Before this could happen many of the modifications and developments described in Section 8.5 would have to have been completed. This would be necessary in order to enable the methodology's global application.

The areas for further development described in Section 8.5 can be summarised as follows:-

a) Modifications to the methodology to allow easier application in its current form. This includes a thorough testing of the methodology in its present form; determining the correct procedure for buffering step increases in gradient on the first and last machine; and seeing if accuracy can be improved by re-iterating the methodology on a line so that you get A/B to a given gradient rather than try to get them level.

b) Improvements to allow application of the methodology to all machining lines. This includes developing the methodology to cope with cycle time imbalance and splitting lines.

c) Application of the methodology to other flow productions - both continual flow processes (steel mills, canning plants etc.) and assembly operations.

There are also other developments which could be considered. At present, the whole operation of the methodology is manual except for the computer simulations. Most, if not all, the operations conducted when applying the methodology are straight forward and are based on simple rules. It could therefore be possible for the whole process to be made to run automatically on a computer in such a way that the simulation model is developed and used as the input and the final buffer pattern required is then produced as output. Such a system would require extensive development, and whether the investment of the programming time could be justified on a system which would be used infrequently is, however, unlikely.

Throughout this work, although they have been appreciated, the cost implications of buffering have been largely ignored. Using the methodology developed in this work, the cost of each buffer installation would have to be calculated for each case. Even to the point of costing the difference between buffering using conveyors and automatic buffers. Any global set of rules for designing lines will have cost and output as the two main variables. A line design model that can encompass cost so that the additional cost of buffers can be traded against a reduction in machine cost is probably the most worthy next step after the full development and testing of the methodology presented.

It must, however, be remembered, that buffers are not the only way to increase line output. There are other ideas which are not strictly a development of the methodology. A typical example is the integration of tool changes. Throughout the work, tool changes have been considered in a similar way to breakdowns, in that they stop a machine operating. They can, however, be treated differently because the frequency of their stoppages can be changed, i.e if a machine becomes blocked and a tool has to be changed after another 10 components, should it be changed early to avoid another stoppage later? If question such as this can be answered then there is potential that the output of a line can be increased further at a lower cost.

References

Acherkan, N.	'Machine Tool Design', Vol. 4	MIR Publications 1969
Anderson, D.R., Moddie, C.L.	'Optimal buffer storage capacity in production line systems'.	IJPR 1969 **7** 233-240
Black, J. T.	'The design of the factory of the future'.	McGraw-Hill 1991 ISBN 007-005550-5
Buzacott, J.A.	'Automatic transfer lines with buffer stocks'.	IJPR 1967 **5** 183-200
Buzacott, J.A.	'Prediction of the efficiency of Production Systems without internal storrage'.	IJPR 1968 **6** 183-200
Buzacott, J.A.	'The Role of Inventary Banks in Flow-Line Production Systems'.	IJPR 1971 **9** 425-436
Carnall, C. A. Wild, R	'The location of variable work stations and the performance of production flow lines'.	IJPR 1976 **14** 703-710
Carrie, A.	'Simulation of Manufacturing Systems.'	Wiley and Sons Ltd, 1988.
Chapman, N.	'Cell Model Integration for Factory Simulation.'	MPhil. Thesis 1993, University of Bath
Christy D. et al.	'The Application of Simulation, a Survey of Industry Practise.'	Interfaces 13, Oct 1983, pp 47-52
Conway, R., Maxwell, W., McLain, J.O., Thomas, L.J.	'The role of Work-In-Progress inventory in serial production lines'.	Operations Research **36** 1988 229-241
Crosby, P.J., Murton, D.R.	'Simulation of Machine Tool Breakdowns'.	Project Report, University of Bath School of Mech Eng., June 1990
Dexion Conveyors	Gainsborough, Lincs	
Ding, J., Greenberg, B. S.	'Bowl shapes are better with buffers - sometimes'.	Probability in the Engineering and Informational Sciences 1991 **5** 159-169

Douglas Smith, L., Brumbaugh, P.	'Allocatin inter-station inventory capacity in unpaced production lines with hetroscedastic processing times.'	IJPR 1977 **15** 163-172
DTI Simulation study group	'Simulation in UK manufacturing industry'	Documentation produced by :- The managment consulting group, Sir William Lions Road, Coventry
Ford GB	Ford Motor Company Dagenham, Essex	
Fortran 77	IBM, Havant, Hants	
Freeman, M. C.	'The effects of Breakdowns and interstage storrage on Production Line Capacity'.	J. of Ind. Eng. 1964 **15** 194-200
Goldratt, M. E. Cox, J.	'The Goal'.	Gower Press 1984 ISBN 09045-0636X
Goldratt, M. E. Cox, J.	'The Goal - Supplement'.	Gower Press 1992 ISBN 0566-09394-3
Goldratt, M. E., Fox, F. E.	'The Race'.	North River Press, Inc. 1986 ISBN 0 88427 062 9
Groover, M.P.	'Automation, Production Systems and Computer Aided Manufacturing'.	Prentice-Hall 1980 ISBN 0-13-054668-2
Hatcher, J. M.	'The Effect of Internal Storrage on the Production Rate of a Series of Stages Having Exponential Service Times'.	AIIE Trans 1969 **1**(2) 150-156
Hillier, F. S., Bolring, R. N.	'The effect of some design factors on the efficiency of production lines with variable operations'.	J. Ind. Engrg 1966 **17** 651-658
Hillier, F. S., So, K.C.	'The effect of machine breakdowns and interstage storage on the performance of production lines'.	IJPR 1991 **29** 2043-2055
Hillier, F. S., So, K. C.	'Some data for applying the Bowl Phenomenon to large production line systems'.	IJPR 1993 **31** 811-822

Hira, D. S., Pandey, P. C.	'Simulation Study of Unbalanced Flow Line Production Systems'.	Proc. Matador Conference 1982 **23** 445-450
Hira, D. S., Pandey, P. C.	'Reliability analysis of transfer lines with finite in-process buffers'.	Proc. Matador Conference 1983 **24** 443-448
Ho, Y.C., Eyler, M.A., Chien, T.T.	'A gradient technique for general buffer design in a production line'.	IJPR 1979 **17** 557-580
HOCUS	PE Consultants, Milton Keynes	
Hopp, W. J., Pati, N., Jones, P.C.	'Optimal Inventory control in a producvtion flow system with failures'.	IJPR 1989 **27** 1367-1384
Hopp, W. J., Simon, J. T.	'Estimating throughput in an unbalanced assembly line'.	IJPR 1993 **31** 851-868
Hunt, G.C.,	'Sequential arrays of waiting lines'.	Opns. Res. 1956 **4** 674-683
IBM	Havant, Hants	
Jafari, M. A., Shanthikumar, J. G.	'Determination of optimal buffer storage capacities and optimal allocation in multistage automatic transfer lines'.	IIE Trans. 1989. **21** 130-135
Law, A.	'Simulation software for manufacturing applications'.	IE Jul 1986 18-20, 74-75.
Magazine, M. J., Silver, G. L.	'Heuristics for determining output and work allocations in series flow lines'.	IJPR 1978 **16** 169-181
MCM Conveyor Systems	Middleton Road, Chadderton Oldham	
Murphy, R. A.	'Estimating the Output of a Series Production System'.	AIEE Trans 1978 **10** 139-148
Muth, E. J., Alkaff, A.	'The bowl phenomenon revisited'.	IJPR 1967 **25** 161-173
Owen, G. W., Mileham, A. R.	'The Influence of Machine Failure on Transfer Line Performance'.	Proc. Nat. Conf. on Prod. Res. 1991
Owen, G. W., Mileham, A. R.	'The Use of Buffers to Improve Transfer Line Output'.	Proc. Nat. Conf. on Prod. Res. 1992

Owen, G. W., Mileham, A. R.	'A Method of Buffering Transfer Lines to Maximise Output'.	Proc. Irish Manuf. Committee Conf. 1993
PCModel	Simulation Software Systems, Sanjose. California	
Production Engineer	Magazine formally produced by I Prod E. Currently Titled 'Manufacturing Engineer' and produced by IEE (1994)	
ProModel PC	Production Modelling Corporation of Europe, Sir William Lyons Road, Coventry CV4 7EZ	
Raban, S., Nagel, R. N.	'Constrain based control of flexible flow lines'.	IJPR 1991 **29** 1941-1951
Rao, N. P.	'A Generalization of the 'Bowl Phenomenon' in series production systems'.	IJPR 1976 **14** 437-443
Rover	Rover Group, Longbridge Birmingham	
Ryder Machine Tools	Thomas Ryder & Son Ltd. City Road Derby DE1 3RP	
Schriber	'Simulation using GPSS.' John Wiley, Editor	Ney York 1974
Sheskin, T. J.	'Allocation of Interstage storrage along an automated production line'.	AIIE Trans 1976 **8** 146-152
SIMAN	System Model Inc. PO Box 10074, PA 16805	
Taka, H. A.	Simulation Modelling'.	Prentice Hall, 1980 ISBN 013-811621-0
Tcha, D., Lee, W., Yamazaki, G.	'Server assignments for multi-stage production systems with finite buffers'.	IJPR 1992 **30** 1637-1653
Whitt, W.	'The best order for Queues in Series'.	Man. Sci. 1985 **31** 475-487
Wild, R.	'Mass-production Management'.	John Wiley & sons 1972 ISBN 0 471 94405

Wild, R.	'Essentials of Production and Operations Management'.	Holt, Rinestat & Winston 1980 ISBN 0 03 910585 7 X
WITNESS	ISTEL Ltd. Highfield House Headless Cross Drive Redditch Worcs. B97 5EQ	
WITNESS	Version 5 User Manual 1989, Istel Ltd.	
WITNESS Modelling Notes	Version 5 Application Notes 1989, Istel Ltd.	
XCELL	Authors: Maxwell & Conway, Cornell University, New York	
Yamakazi, G., Sakasegawa, H.	'Properties of Duality in Tandem Queueing systems'.	Ann. Inst. Stat. Math. 1975 **27** 201-212
Yamashina, H., Okamura, K.	'Analysing in-process buffer for multiple transfer line systems'.	IJPR 1983 **21** 183-195
Yang, H. H., Chen, K. N., Chang, W. S., Wang, Y. X.	'An investigation on Reliability and optimum buffer stocks of amchine tool, transferlines and their simulation on digital computers'.	Proc. Matador Conference 1983 571-576
Yeoh, O.B.	'Integrated Transfer Line Manufacturing Systems - The Application of Information Theory and Microprocessor Technology'.	Phd Thesis 1982, University of Birmingham, England

Appendices

Appendix I

Input Parameters and Performance Measures of Transfer Lines

Some of the major parameters are as follows:-

Inputs	**Outputs**
No. of Machines	Efficiency
Level of Balance	Time in System
No. Cutters per Machine	Cost per Unit Volume
Tool Change Frequency	Scrap Levels
Tool Change Time	Queue Lengths
Breakdown Frequency	Operator Utilisation
Repair Time	Tool Costs
Cycle Time	- Utilisation
Queue Spaces	- % of life used
Operator Availability	
Repair Labour	
Materials Availability	
Conveyor Speeds	
Conveyor Breakdowns	
Reject Levels	

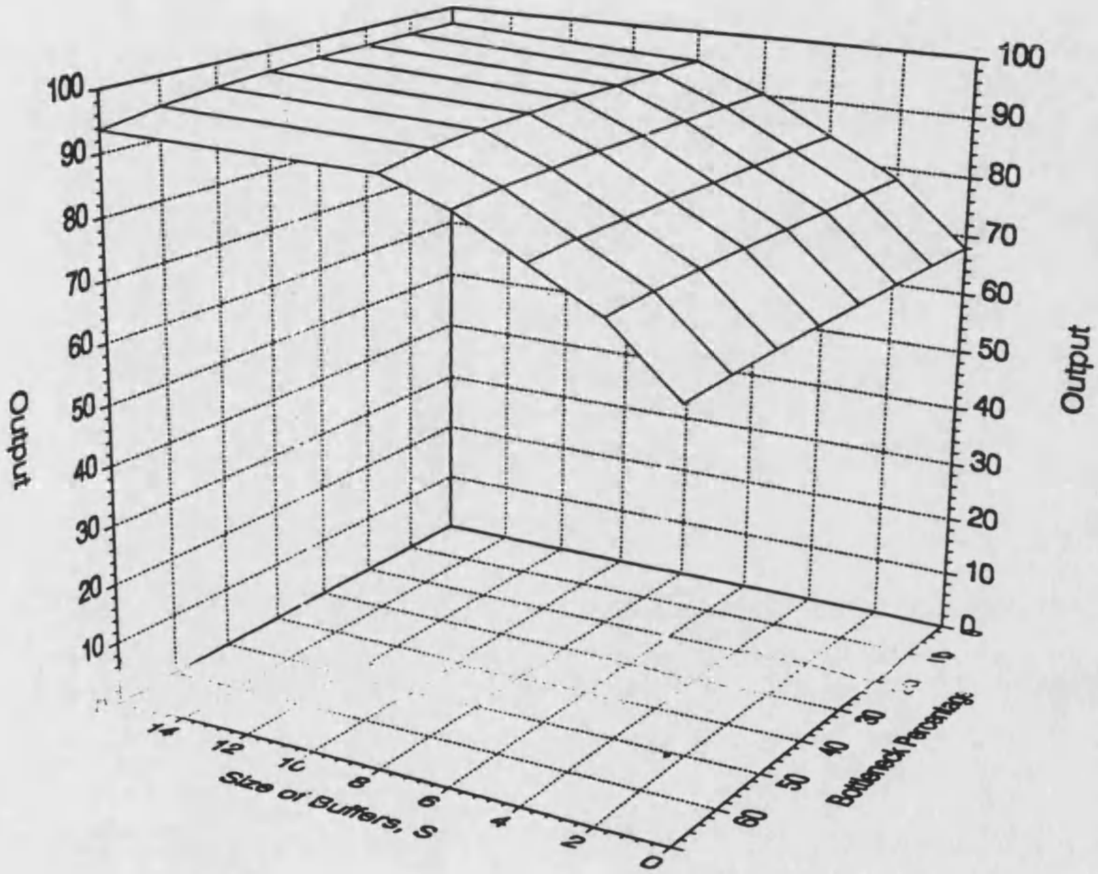
Note : Some of these measures are dependant on others (e.g Level of balance is dependant on the machine cycle time but both are inputs).

Appendix II

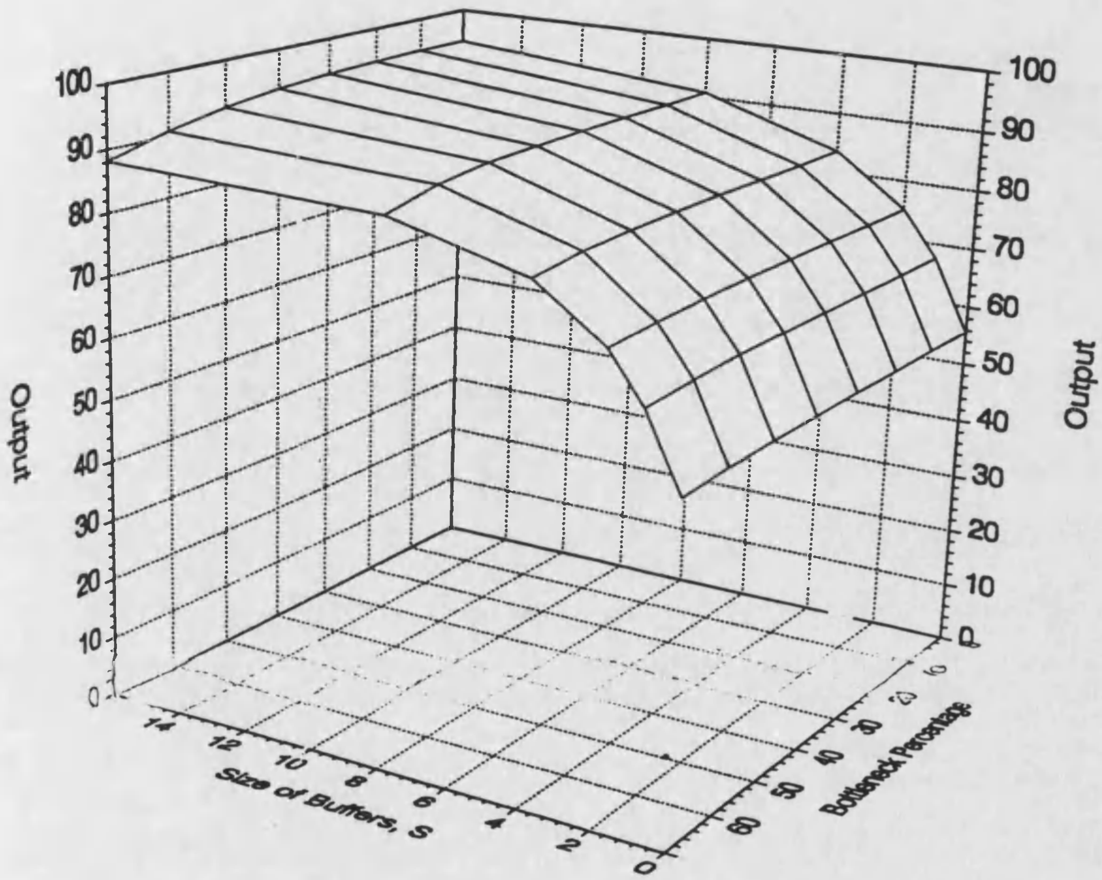
Graphs showing the effect of different Buffer Sizes and Bottlenecks on Line Output

The following graphs are results from simulations run on a 30 machine long line. Each graph is for a different mean percentage downtime (from 2% to 16% at 2% intervals) for all machines except the bottleneck at M15. Buffers have been evenly spaced along the line.

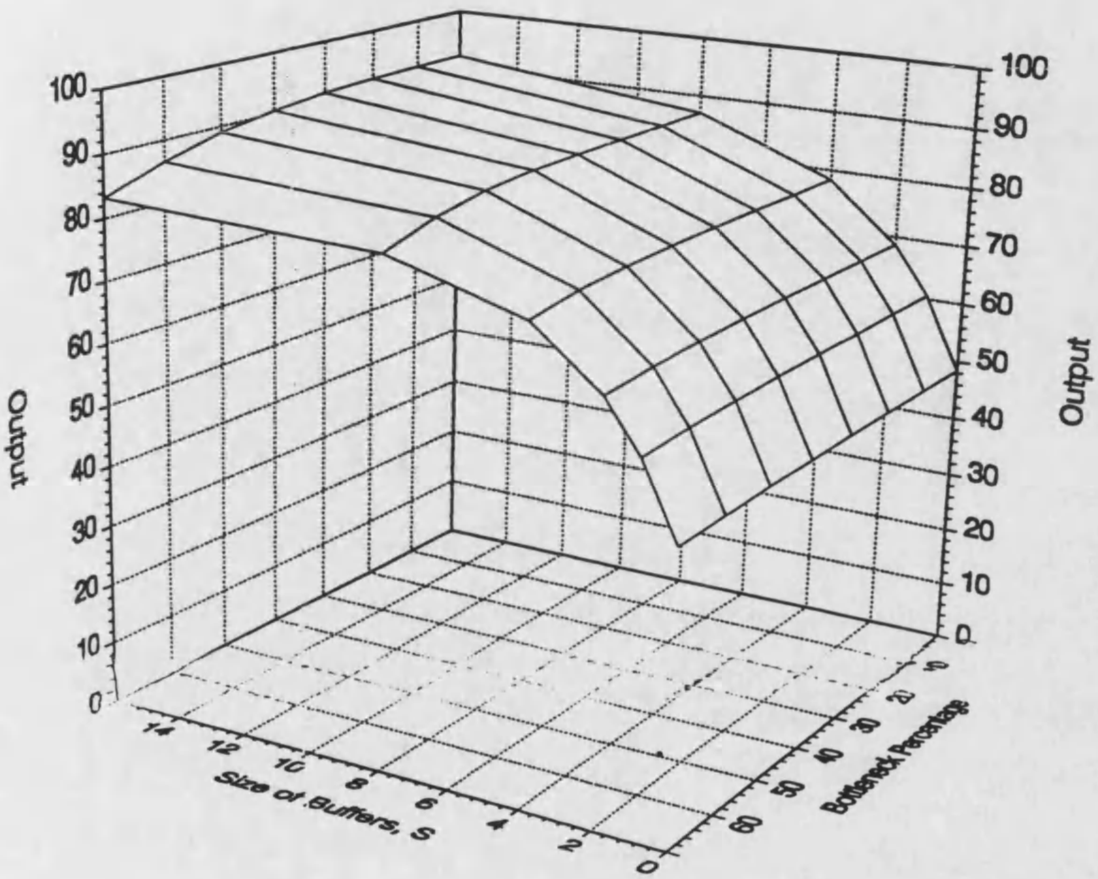
a) Machine downtime 2%



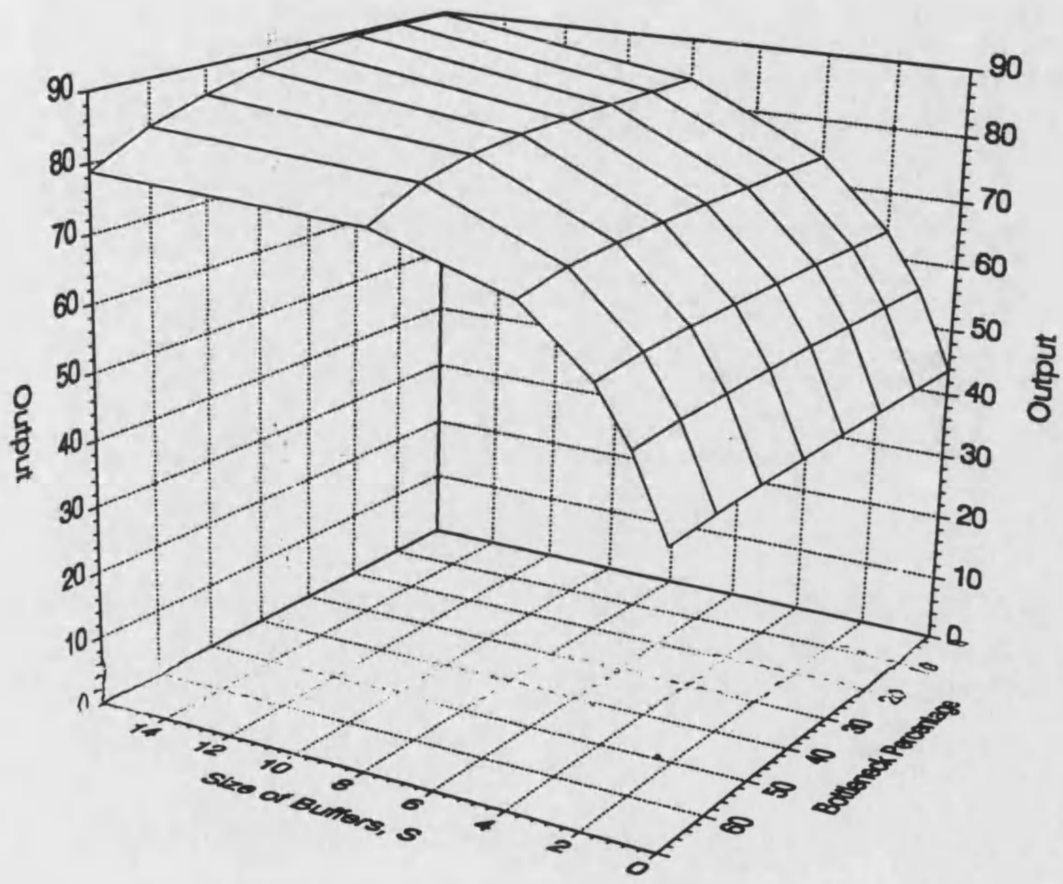
b) Machine downtime 4%



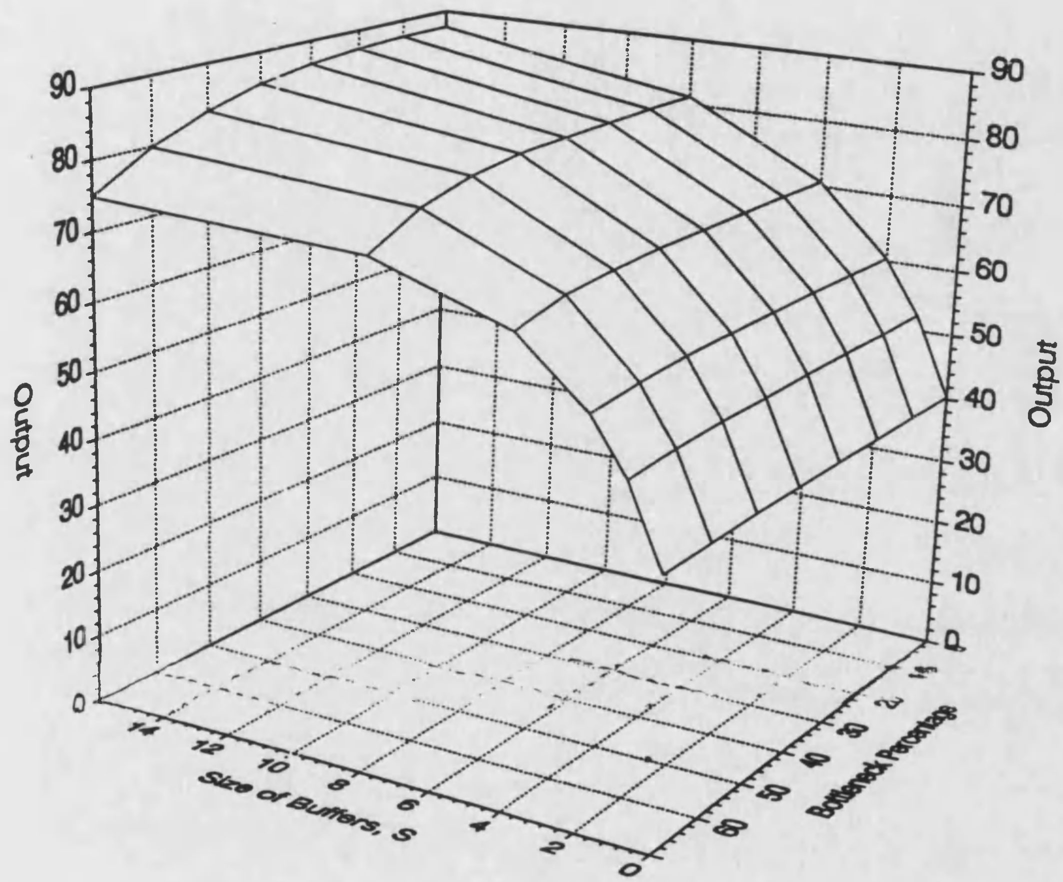
c) Machine Downtime 6%



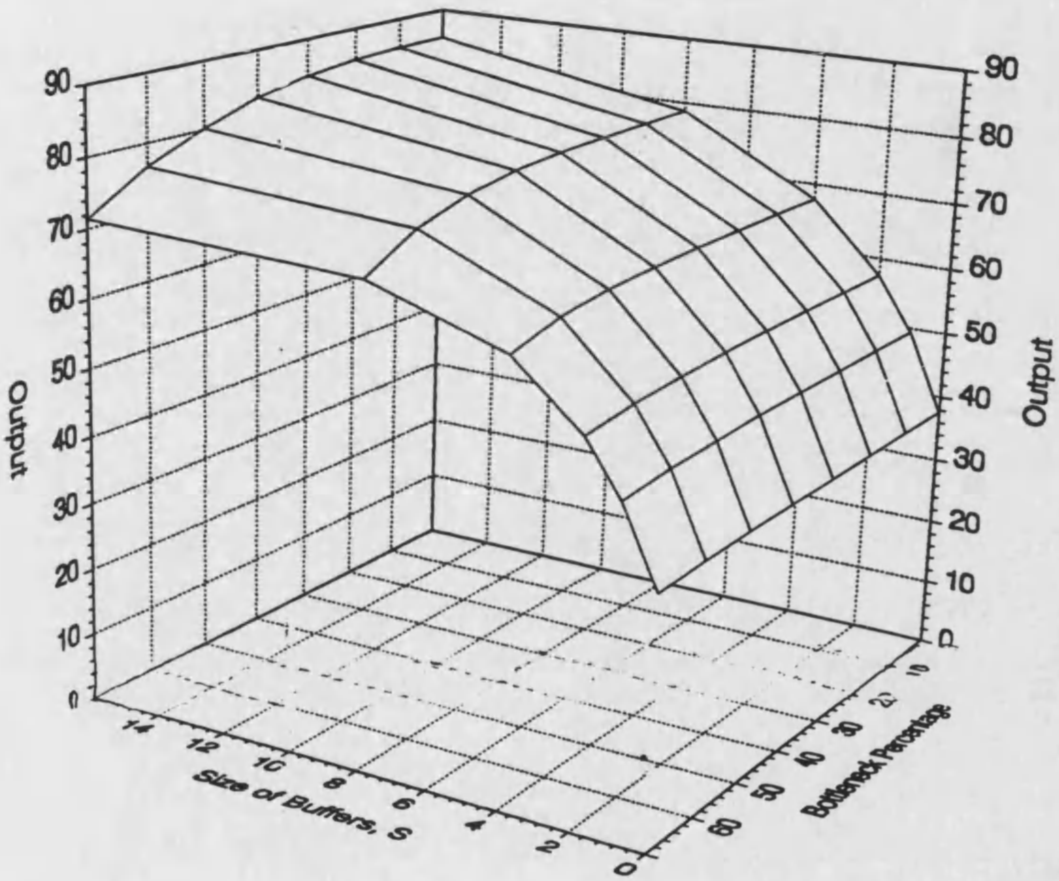
d) Machine downtime 8%



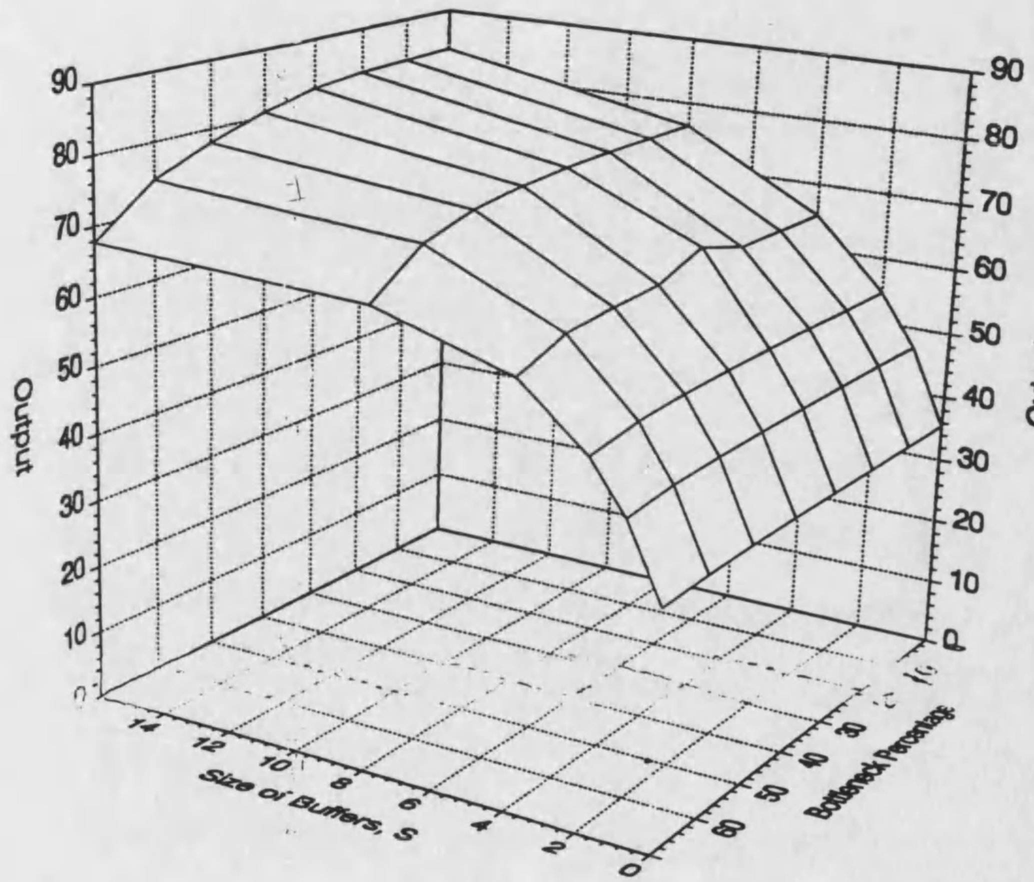
e) Machine downtime 10%



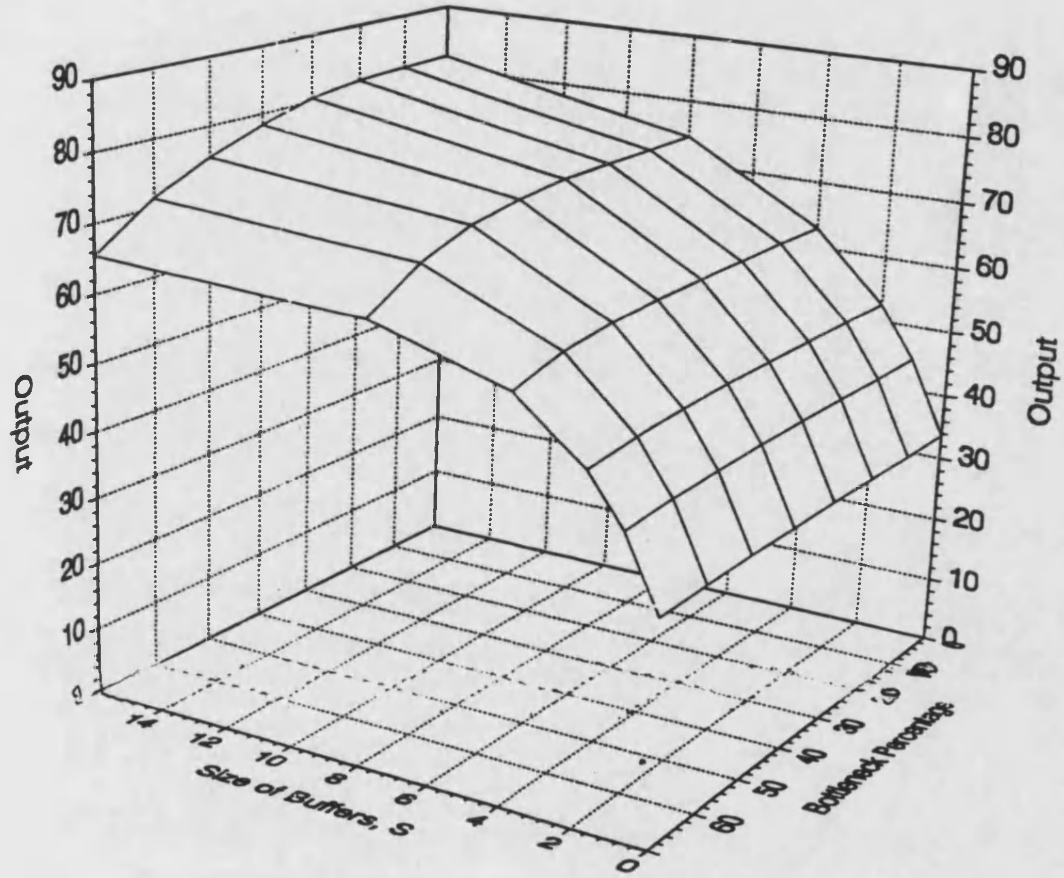
f) Machine downtime 12%



g) Machine downtime 14%



h) Machine downtime 16%



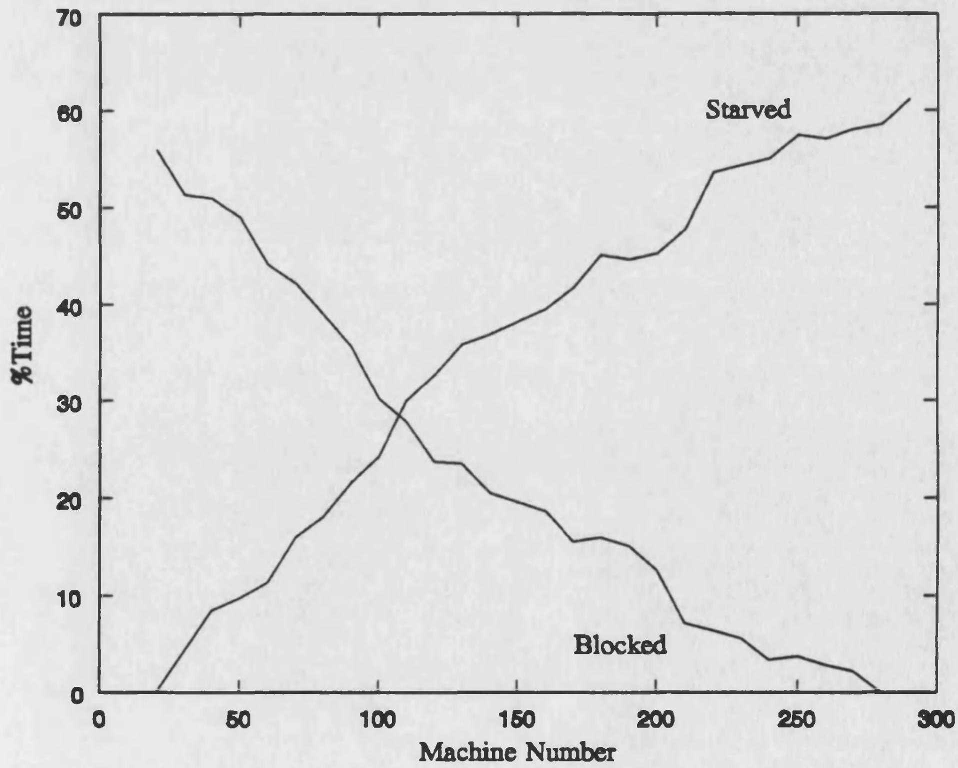
Appendix III

Case study simulation results

The data and graphs on the following pages refer to the different stages of buffering the modified Ford Zeta engine block machining line described in Chapter 9.

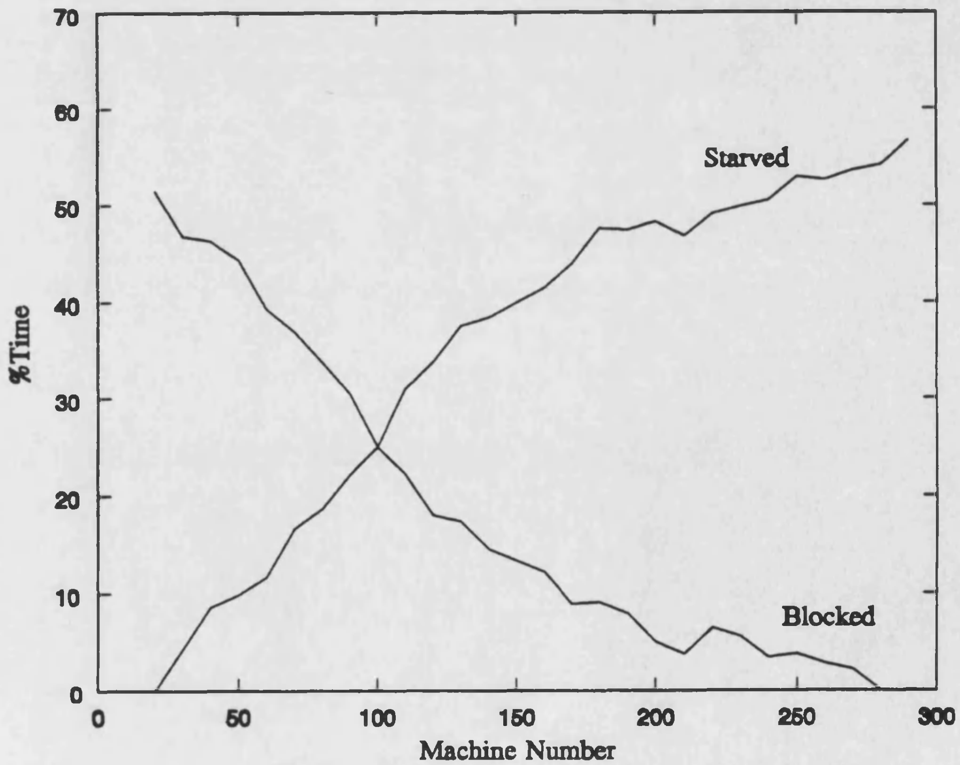
Stage 1

	%TIME				GRADIENTS	
MACHINE	WAITING	BUSY	BLOCKED	DOWN	WAITING	BLOCKED
20	0.00	38.87	55.87	5.26	4.24	
30	4.24	38.87	51.25	5.63	4.23	4.62
40	8.47	38.87	50.89	1.77	1.24	0.36
50	9.71	38.87	48.94	2.48	1.59	1.95
60	11.30	38.87	44.13	5.70	4.65	4.81
70	15.95	38.87	42.15	3.03	2.14	1.99
80	18.09	38.87	38.92	4.11	3.44	3.22
90	21.53	38.87	35.67	3.93	2.69	3.25
100	24.22	38.87	30.25	6.66	5.70	5.42
110	29.92	38.87	27.85	3.36	2.72	2.40
120	32.64	38.87	23.73	4.76	3.24	4.12
130	35.88	38.87	23.53	1.71	0.99	0.19
140	36.88	38.87	20.56	3.69	2.66	2.97
160	39.53	38.87	18.71	2.89	2.20	1.85
170	41.74	38.87	15.55	3.85	3.38	3.16
180	45.11	38.87	16.02	0.00	-0.49	-0.47
190	44.62	38.87	15.10	1.41	0.59	0.92
200	45.21	38.87	12.57	3.35	2.56	2.53
210	47.77	38.87	7.14	6.22	5.73	5.43
220	53.50	38.87	6.43	1.19	0.80	0.71
230	54.30	38.87	5.59	1.24	0.72	0.84
240	55.02	38.87	3.33	2.77	2.40	2.26
250	57.42	38.87	3.71	0.00	-0.39	-0.38
260	57.03	38.87	2.83	1.27	0.94	0.88
270	57.97	38.87	2.16	0.99	0.58	0.66
280	58.55	38.87	0.00	2.57	2.57	2.16
290	61.13	38.87	0.00	0.00		0.00



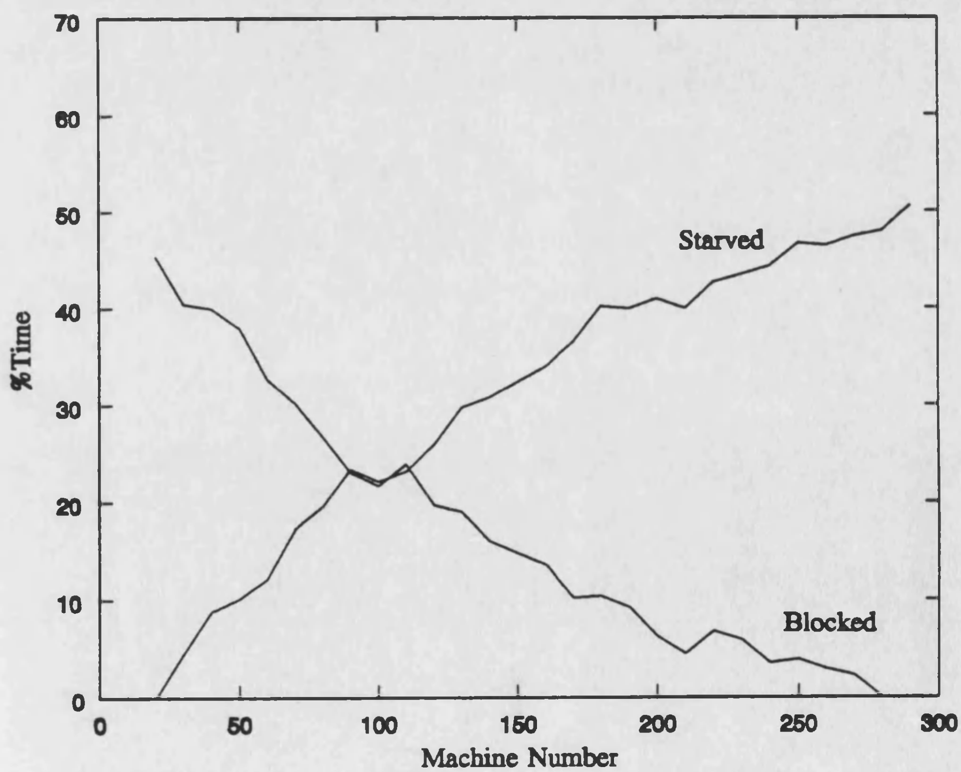
Stage 2

	%TIME				GRADIENTS	
MACHINE	WAITING	BUSY	BLOCKED	DOWN	WAITING	BLOCKED
20	0.00	43.33	51.41	5.26	4.29	
30	4.29	43.33	46.75	5.63	4.29	4.66
40	8.57	43.33	46.32	1.77	1.25	0.42
50	9.83	43.33	44.36	2.48	1.79	1.96
60	11.62	43.33	39.36	5.70	5.01	5.01
70	16.63	43.33	37.01	3.03	2.08	2.35
80	18.70	43.33	33.85	4.11	3.48	3.16
90	22.18	43.33	30.56	3.93	2.82	3.29
100	25.00	43.33	25.01	6.66	5.99	5.55
110	30.99	43.33	22.32	3.36	2.89	2.69
120	33.88	43.33	18.03	4.76	3.67	4.29
130	37.55	43.33	17.41	1.71	0.91	0.62
140	38.46	43.33	14.52	3.69	3.09	2.89
160	41.55	43.33	12.23	2.89	2.40	2.28
170	43.94	43.33	8.88	3.85	3.65	3.35
180	47.59	43.33	9.09	0.00	-0.21	-0.20
190	47.37	43.33	7.89	1.41	0.90	1.19
200	48.28	43.33	5.04	3.35	-1.52	2.85
210	46.75	43.32	3.71	6.22	2.27	1.33
220	49.03	43.33	6.45	1.19	0.79	-2.74
230	49.82	43.33	5.61	1.24	0.66	0.84
240	50.48	43.33	3.42	2.77	2.42	2.20
250	52.90	43.33	3.77	0.00	-0.35	-0.35
260	52.55	43.33	2.84	1.28	0.96	0.93
270	53.51	43.33	2.16	0.99	0.58	0.68
280	54.09	43.33	0.00	2.57	2.57	2.16
290	56.67	43.33	0.00	0.00		0.00



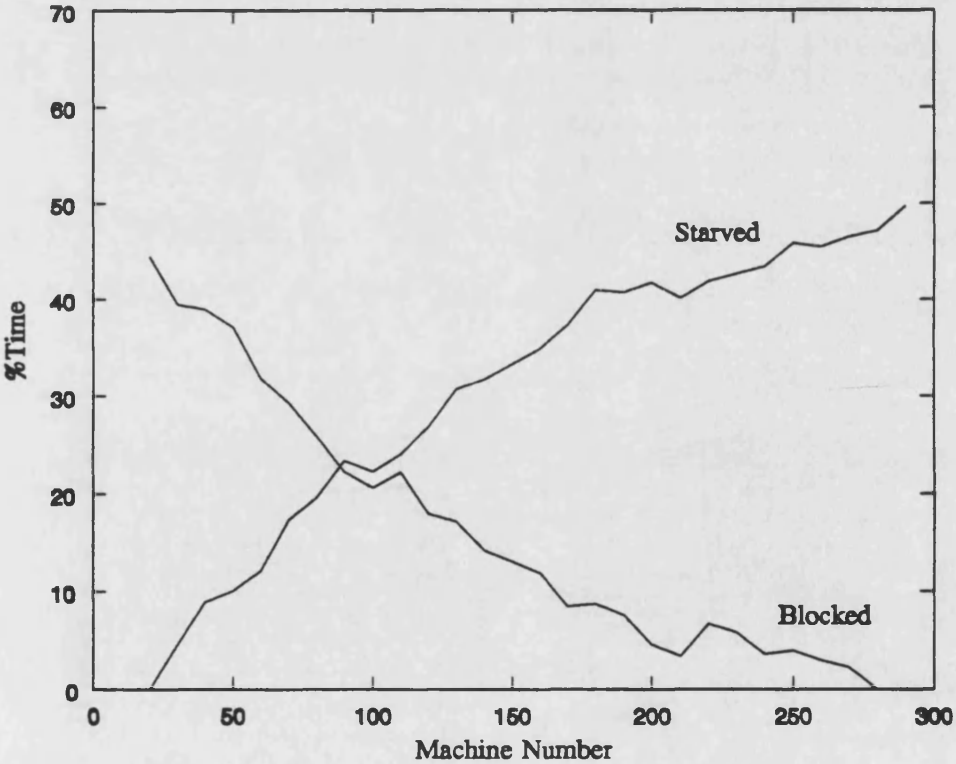
Stage 3

MACHINE	%TIME WAITING	BUSY	BLOCKED	DOWN	GRADIENTS WAITING	BLOCKED
20	0.00	49.47	45.28	5.26	4.45	
30	4.45	49.47	40.45	5.63	4.33	4.83
40	8.78	49.47	39.98	1.77	1.32	0.47
50	10.11	49.47	37.95	2.48	2.02	2.03
60	12.13	49.47	32.71	5.70	5.16	5.24
70	17.29	49.47	30.21	3.03	2.34	2.50
80	19.62	49.47	26.80	4.11	3.83	3.42
90	23.45	49.46	23.16	3.93	-1.28	3.64
100	22.17	49.46	21.71	6.66	1.02	1.45
110	23.19	49.46	23.98	3.36	2.84	-2.28
120	26.03	49.46	19.74	4.76	3.75	4.24
130	29.78	49.46	19.04	1.71	1.11	0.70
140	30.89	49.46	15.95	3.69	3.17	3.09
160	34.06	49.46	13.58	2.89	2.48	2.37
170	36.54	49.46	10.15	3.85	3.66	3.44
180	40.20	49.46	10.34	0.00	-0.27	-0.19
190	39.93	49.46	9.20	1.41	1.03	1.14
200	40.97	49.46	6.22	3.35	-1.05	2.98
210	39.91	49.45	4.41	6.22	2.72	1.81
220	42.64	49.46	6.71	1.19	0.82	-2.29
230	43.46	49.46	5.84	1.24	0.81	0.86
240	44.26	49.46	3.50	2.77	2.41	2.35
250	46.67	49.47	3.86	0.00	-0.34	-0.36
260	46.33	49.47	2.92	1.28	0.99	0.94
270	47.32	49.47	2.22	0.99	0.63	0.71
280	47.96	49.47	0.00	2.57	2.57	2.22
290	50.53	49.47	0.00	0.00		0.00



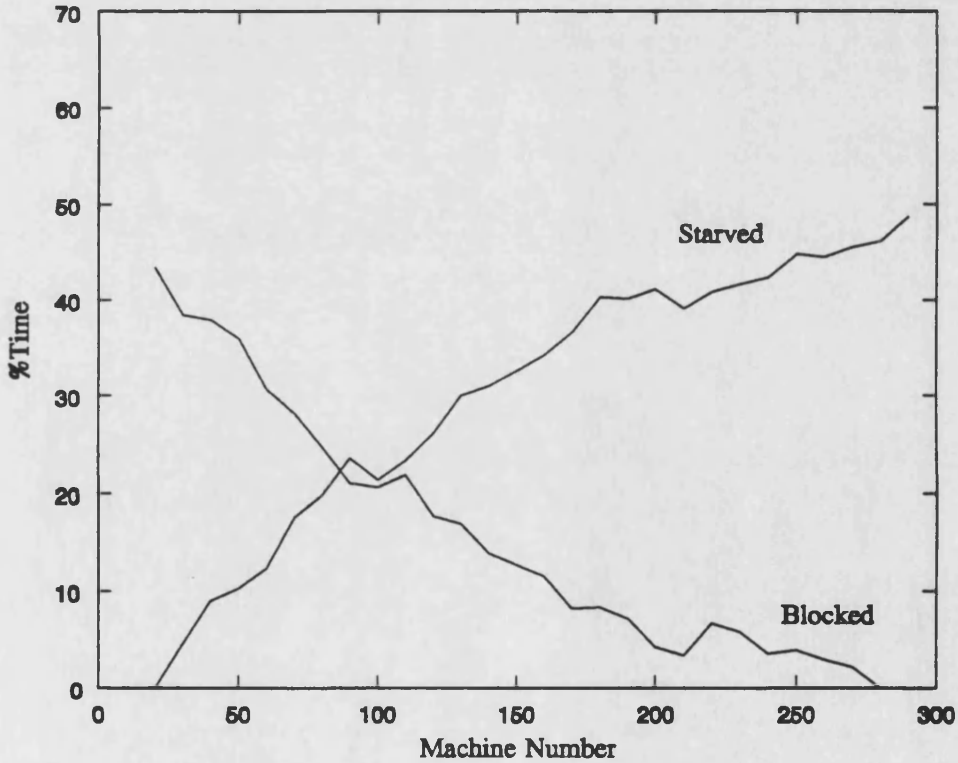
Stage 4

	%TIME				GRADIENTS	
MACHINE	WAITING	BUSY	BLOCKED	DOWN	WAITING	BLOCKED
20	0.00	50.35	44.39	5.26	4.53	
30	4.53	50.35	39.48	5.63	4.36	4.91
40	8.89	50.35	38.99	1.77	1.18	0.49
50	10.07	50.35	37.10	2.48	2.04	1.89
60	12.11	50.35	31.84	5.70	5.16	5.26
70	17.27	50.35	29.34	3.03	2.33	2.50
80	19.60	50.35	25.93	4.11	3.82	3.41
90	23.43	50.35	22.30	3.93	-1.11	3.64
100	22.31	50.35	20.68	6.66	1.76	1.62
110	24.07	50.35	22.22	3.36	2.84	-1.54
120	26.90	50.35	17.98	4.76	3.85	4.24
130	30.75	50.35	17.18	1.71	1.00	0.80
140	31.76	50.35	14.20	3.69	3.17	2.98
160	34.93	50.35	11.83	2.89	2.40	2.37
170	37.33	50.35	8.48	3.85	3.60	3.36
180	40.93	50.35	8.72	0.00	-0.26	-0.24
190	40.67	50.35	7.57	1.41	1.04	1.14
200	41.71	50.35	4.59	3.35	-1.62	2.99
210	40.09	50.33	3.36	6.22	1.71	1.23
220	41.80	50.34	6.67	1.19	0.82	-3.31
230	42.62	50.34	5.80	1.24	0.75	0.86
240	43.37	50.34	3.52	2.77	2.41	2.29
250	45.78	50.34	3.88	0.00	-0.33	-0.37
260	45.45	50.34	2.93	1.28	1.00	0.95
270	46.45	50.34	2.22	0.99	0.63	0.72
280	47.08	50.34	0.00	2.57	2.57	2.22
290	49.66	50.34	0.00	0.00		0.00



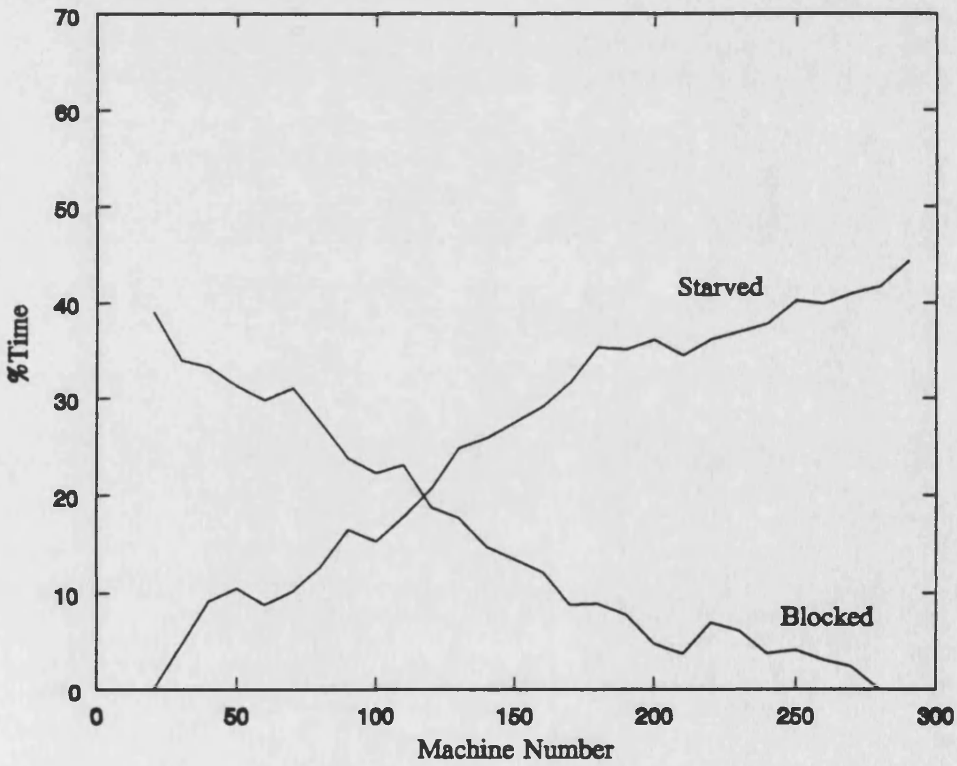
Stage 5

	%TIME				GRADIENTS	
MACHINE	WAITING	BUSY	BLOCKED	DOWN	WAITING	BLOCKED
20	0.00	51.33	43.42	5.26	4.55	
30	4.55	51.33	38.49	5.63	4.39	4.92
40	8.94	51.33	37.96	1.77	1.30	0.53
50	10.23	51.33	35.96	2.48	2.04	2.00
60	12.27	51.33	30.70	5.70	5.19	5.26
70	17.46	51.33	28.18	3.03	2.34	2.53
80	19.81	51.33	24.75	4.11	3.85	3.43
90	23.66	51.33	21.09	3.93	-2.29	3.66
100	21.37	51.33	20.64	6.66	2.01	0.45
110	23.38	51.32	21.93	3.36	2.85	-1.29
120	26.23	51.32	17.68	4.76	3.83	4.25
130	30.06	51.32	16.90	1.71	1.02	0.78
140	31.08	51.32	13.90	3.70	3.22	3.01
160	34.29	51.32	11.49	2.89	2.38	2.41
170	36.67	51.32	8.16	3.85	3.67	3.33
180	40.34	51.32	8.33	0.00	-0.25	-0.18
190	40.09	51.32	7.18	1.41	1.02	1.16
200	41.12	51.32	4.21	3.35	-1.98	2.97
210	39.14	51.29	3.35	6.22	1.71	0.86
220	40.85	51.30	6.66	1.19	0.81	-3.31
230	41.65	51.30	5.81	1.24	0.71	0.85
240	42.36	51.30	3.56	2.77	2.42	2.25
250	44.78	51.31	3.91	0.00	-0.31	-0.35
260	44.48	51.31	2.94	1.28	1.01	0.97
270	45.49	51.31	2.21	0.99	0.63	0.73
280	46.12	51.31	0.00	2.57	2.57	2.21
290	48.69	51.31	0.00	0.00		0.00



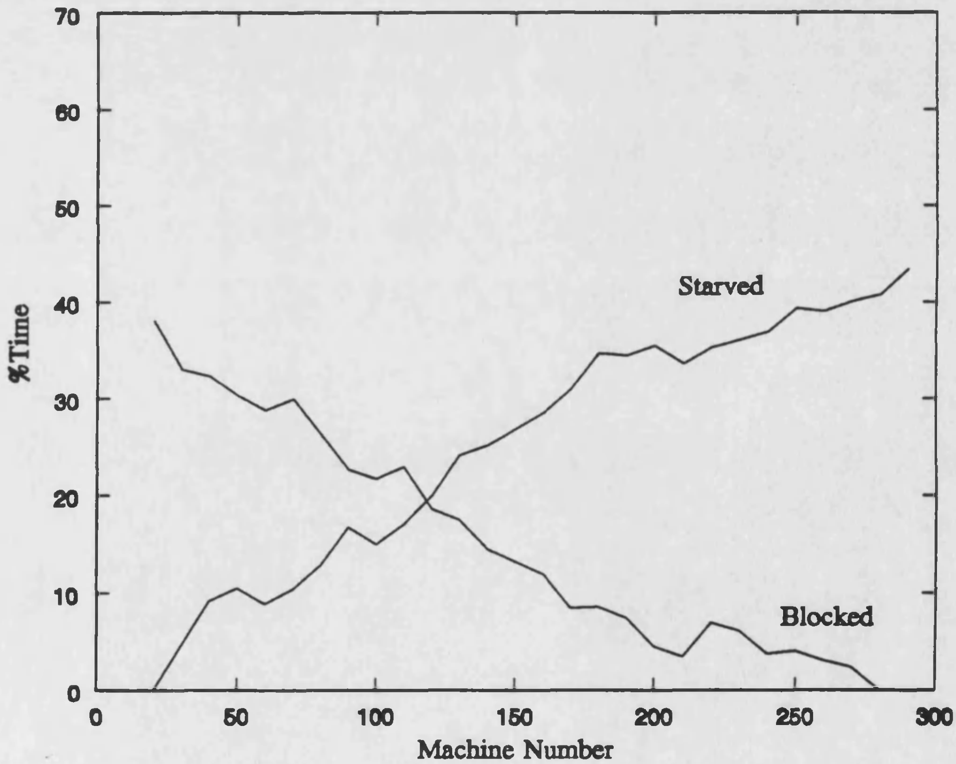
Stage 6

	%TIME				GRADIENTS	
MACHINE	WAITING	BUSY	BLOCKED	DOWN	WAITING	BLOCKED
20	0.00	55.73	39.01	5.26	4.59	
30	4.59	55.73	34.04	5.63	4.55	4.97
40	9.14	55.73	33.36	1.77	1.34	0.68
50	10.48	55.73	31.31	2.48	-1.73	2.04
60	8.75	55.73	29.82	5.70	1.39	1.49
70	10.14	55.73	31.10	3.03	2.53	-1.27
80	12.67	55.73	27.49	4.11	3.85	3.61
90	16.51	55.73	23.83	3.93	-1.22	3.66
100	15.29	55.73	22.32	6.66	2.51	1.51
110	17.80	55.73	23.11	3.36	2.90	-0.79
120	20.70	55.73	18.81	4.76	4.16	4.30
130	24.86	55.73	17.69	1.71	1.05	1.11
140	25.91	55.73	14.66	3.70	3.32	3.03
160	29.23	55.73	12.15	2.89	2.46	2.51
170	31.69	55.73	8.74	3.85	3.70	3.42
180	35.38	55.73	8.89	0.00	-0.26	-0.15
190	35.13	55.73	7.74	1.41	1.03	1.15
200	36.16	55.73	4.76	3.35	-1.70	2.97
210	34.46	55.70	3.63	6.22	1.72	1.14
220	36.18	55.70	6.92	1.19	0.81	-3.30
230	36.99	55.70	6.08	1.24	0.81	0.85
240	37.79	55.70	3.73	2.77	2.43	2.35
250	40.22	55.70	4.08	0.00	-0.29	-0.35
260	39.93	55.70	3.09	1.28	1.01	0.98
270	40.94	55.71	2.37	0.99	0.78	0.73
280	41.72	55.71	0.00	2.57	2.57	2.37
290	44.29	55.71	0.00	0.00		0.00



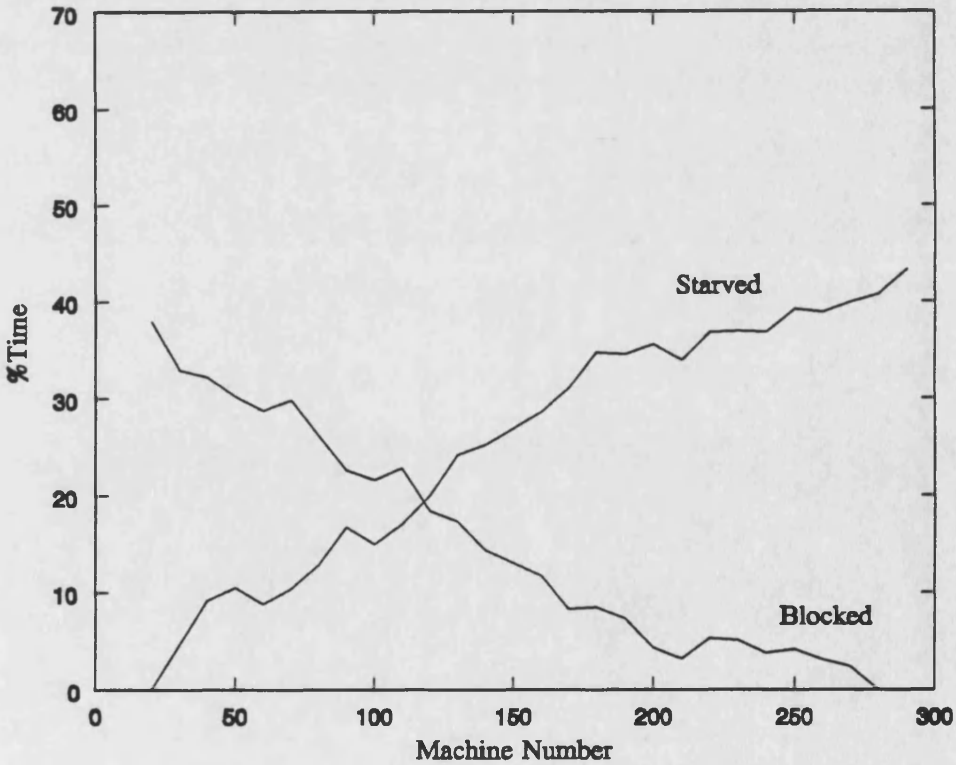
Stage 7

MACHINE	%TIME WAITING	BUSY	BLOCKED	DOWN	GRADIENTS WAITING	BLOCKED
20	0.00	56.66	38.08	5.26	4.60	
30	4.60	56.66	33.11	5.63	4.57	4.98
40	9.17	56.66	32.40	1.77	1.35	0.71
50	10.52	56.66	30.34	2.48	-1.71	2.06
60	8.81	56.66	28.84	5.70	1.52	1.51
70	10.32	56.66	29.99	3.03	2.56	-1.15
80	12.88	56.66	26.35	4.11	3.84	3.64
90	16.71	56.66	22.70	3.93	-1.77	3.65
100	14.94	56.66	21.74	6.66	2.05	0.96
110	17.00	56.66	22.99	3.36	3.00	-1.25
120	19.99	56.66	18.59	4.76	4.16	4.40
130	24.15	56.66	17.48	1.71	1.05	1.11
140	25.20	56.66	14.44	3.70	3.38	3.04
160	28.58	56.66	11.87	2.89	2.46	2.57
170	31.04	56.66	8.46	3.85	3.70	3.41
180	34.74	56.66	8.61	0.00	-0.23	-0.15
190	34.50	56.66	7.43	1.41	1.02	1.17
200	35.53	56.66	4.46	3.35	-1.86	2.97
210	33.66	56.62	3.50	6.22	1.64	0.96
220	35.30	56.62	6.88	1.19	0.80	-3.38
230	36.10	56.62	6.04	1.24	0.81	0.84
240	36.91	56.62	3.70	2.77	2.43	2.35
250	39.33	56.63	4.04	0.00	-0.30	-0.35
260	39.04	56.63	3.06	1.28	1.01	0.98
270	40.05	56.63	2.33	0.99	0.75	0.73
280	40.80	56.63	0.00	2.57	2.57	2.33
290	43.37	56.63	0.00	0.00		0.00



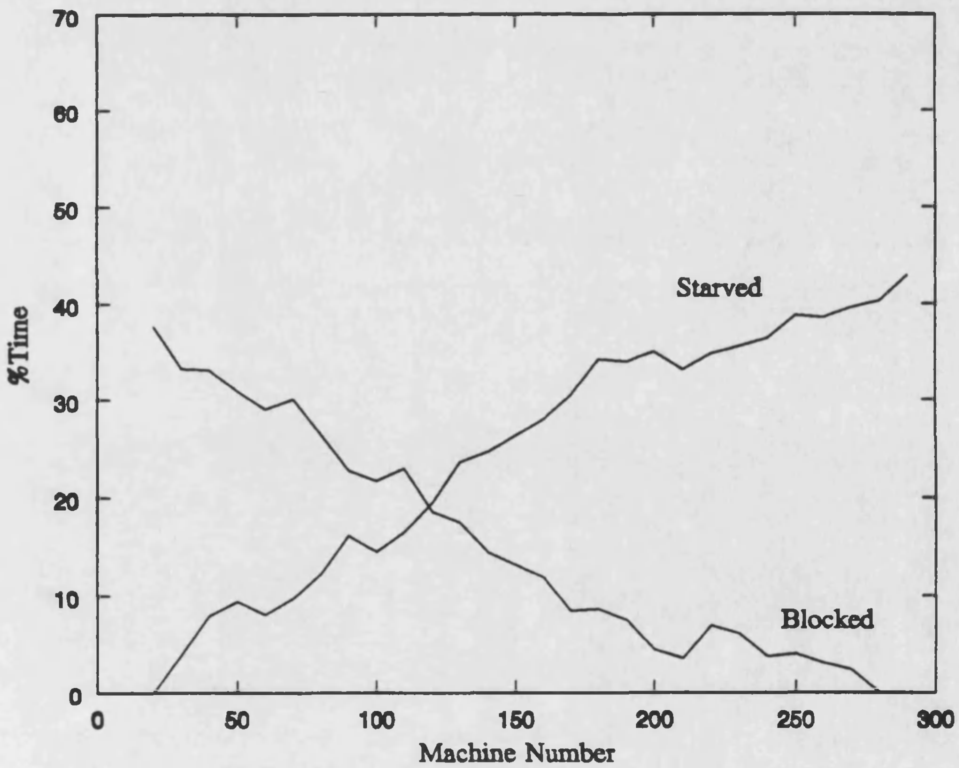
Stage 8a

MACHINE	%TIME WAITING	BUSY	BLOCKED	DOWN	GRADIENTS WAITING	BLOCKED
20	0.00	56.78	37.96	5.26	4.60	
30	4.60	56.78	32.98	5.63	4.56	4.98
40	9.17	56.78	32.28	1.77	1.35	0.70
50	10.52	56.78	30.22	2.48	-1.76	2.06
60	8.76	56.78	28.77	5.70	1.57	1.45
70	10.33	56.78	29.86	3.03	2.56	-1.09
80	12.88	56.78	26.22	4.11	3.84	3.64
90	16.72	56.78	22.57	3.93	-1.75	3.65
100	14.96	56.78	21.60	6.66	2.06	0.98
110	17.03	56.78	22.83	3.36	3.00	-1.23
120	20.03	56.78	18.43	4.76	4.16	4.40
130	24.19	56.78	17.32	1.71	1.05	1.10
140	25.23	56.78	14.29	3.70	3.38	3.03
160	28.61	56.78	11.72	2.89	2.47	2.57
170	31.08	56.78	8.30	3.85	3.70	3.42
180	34.77	56.78	8.45	0.00	-0.24	-0.15
190	34.54	56.78	7.28	1.41	1.04	1.17
200	35.57	56.78	4.29	3.35	-1.66	2.98
210	33.92	56.74	3.12	6.22	2.87	1.17
220	36.78	56.75	5.28	1.19	0.15	-2.15
230	36.93	56.75	5.08	1.24	-0.16	0.19
240	36.77	56.75	3.71	2.77	2.40	1.38
250	39.16	56.75	4.09	0.00	-0.29	-0.38
260	38.87	56.75	3.10	1.28	1.02	0.99
270	39.90	56.75	2.36	0.99	0.78	0.74
280	40.67	56.75	0.00	2.57	2.57	2.36
290	43.25	56.75	0.00	0.00		0.00



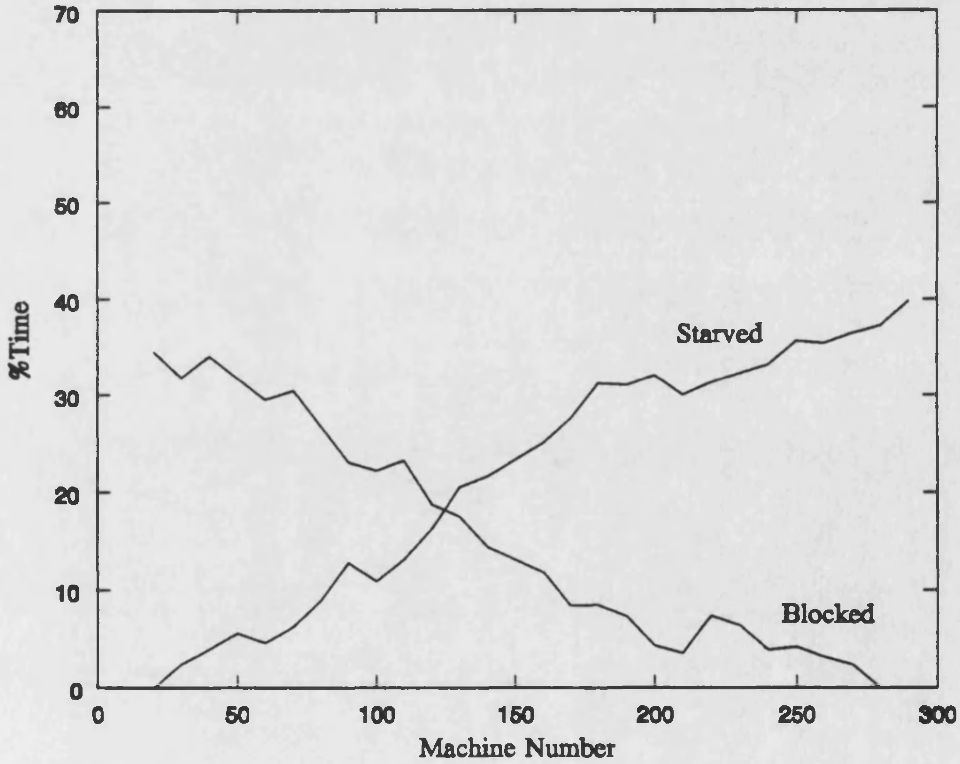
Stage 8b

MACHINE	%TIME WAITING	BUSY	BLOCKED	DOWN	GRADIENTS WAITING	BLOCKED
20	0.00	57.13	37.61	5.26	3.93	
30	3.93	57.13	33.30	5.63	3.97	4.31
40	7.91	57.13	33.19	1.77	1.53	0.11
50	9.44	57.13	30.95	2.48	-1.35	2.24
60	8.09	57.13	29.08	5.70	1.65	1.87
70	9.74	57.13	30.09	3.03	2.57	-1.01
80	12.31	57.13	26.45	4.11	3.84	3.65
90	16.15	57.13	22.80	3.93	-1.64	3.65
100	14.51	57.13	21.70	6.66	2.00	1.09
110	16.51	57.13	23.00	3.36	2.99	-1.30
120	19.50	57.13	18.61	4.76	4.15	4.39
130	23.65	57.13	17.51	1.71	1.06	1.10
140	24.70	57.13	14.47	3.70	3.38	3.04
160	28.08	57.13	11.90	2.89	2.45	2.57
170	30.54	57.13	8.49	3.85	3.70	3.41
180	34.24	57.13	8.63	0.00	-0.24	-0.15
190	34.00	57.13	7.46	1.41	1.02	1.17
200	35.02	57.13	4.50	3.35	-1.84	2.97
210	33.18	57.09	3.50	6.22	1.62	0.99
220	34.80	57.10	6.90	1.19	0.80	-3.40
230	35.61	57.10	6.06	1.24	0.82	0.85
240	36.43	57.10	3.70	2.77	2.43	2.36
250	38.86	57.10	4.04	0.00	-0.29	-0.34
260	38.57	57.10	3.06	1.28	1.01	0.99
270	39.58	57.10	2.33	0.99	0.75	0.72
280	40.32	57.10	0.00	2.57	2.57	2.33
290	42.90	57.10	0.00	0.00		0.00



Stage 9

	%TIME				GRADIENTS	
MACHINE	WAITING	BUSY	BLOCKED	DOWN	WAITING	BLOCKED
20	0.00	60.24	34.50	5.26	2.33	
30	2.33	60.24	31.80	5.63	1.62	2.71
40	3.94	60.24	34.04	1.77	1.60	-2.25
50	5.55	60.24	31.73	2.48	-1.00	2.31
60	4.55	60.24	29.51	5.70	1.68	2.22
70	6.23	60.24	30.50	3.03	2.66	-0.98
80	8.89	60.24	26.76	4.11	3.89	3.74
90	12.78	60.24	23.05	3.93	-1.92	3.70
100	10.86	60.24	22.24	6.66	2.28	0.81
110	13.14	60.24	23.26	3.36	3.14	-1.01
120	16.28	60.24	18.72	4.76	4.27	4.54
130	20.55	60.24	17.50	1.71	1.14	1.22
140	21.69	60.24	14.37	3.70	3.38	3.13
160	25.07	60.24	11.80	2.89	2.50	2.58
170	27.57	60.24	8.35	3.85	3.70	3.45
180	31.27	60.24	8.49	0.00	-0.22	-0.15
190	31.05	60.24	7.31	1.41	1.04	1.18
200	32.09	60.24	4.32	3.35	-2.03	2.99
210	30.06	60.19	3.53	6.22	1.22	0.79
220	31.28	60.19	7.33	1.19	0.97	-3.80
230	32.25	60.20	6.32	1.24	0.90	1.02
240	33.15	60.20	3.88	2.77	2.49	2.44
250	35.64	60.20	4.16	0.00	-0.25	-0.28
260	35.39	60.20	3.13	1.28	1.03	1.03
270	36.43	60.20	2.38	0.99	0.80	0.75
280	37.23	60.20	0.00	2.57	2.57	2.38
290	39.80	60.20	0.00	0.00		0.00



Appendix IV - Papers published by the author during this research

i) - 'The Influence of Machine Failure on Transfer Line Performance'.
Owen & Mileham 1991

ii) - 'The Use of Buffers to Improve Transfer Line Performance'.
Owen & Mileham 1992

iii) - 'A Method of Buffering Transfer Lines to Maximise Output'.
Owen & Mileham 1993

Full references for these papers are given in the references chapter

The Influence of Machine Failure on Transfer Line Performance

Geraint Wyn Owen and Dr.A.R. Mileham, University of Bath

Seventh National Conference on Production Research

Abstract

Transfer lines are typically used to machine single prismatic type components with high demand. They are comprised of many automatically linked machines, the number of which reflects the component complexity and required cycle time. In practice the output is significantly reduced by machine stoppages. This paper concentrates on the effect to the line output of stoppages irrespective of how the stoppage is caused. In order to evaluate the effect of stoppages, a simulation of a representative transfer line has been developed. The effects of the time windows introduced by stoppages are examined, together with the trends in starving and blocking that occur under varying line conditions. The consequential effects on uptime and output are also presented.

1.0 Introduction

The main causes of transfer line stoppages are tool changes and equipment and machine breakdowns. Stoppages not only effect the machine that is stopped, machines following the one stopped (downstream) are no longer supplied (starvation) and those preceding the one stopped (upstream) cannot cycle as their output is blocked. These disruptions reduce the productivity and hence waste the high capital investment typically associated with line construction. On-line storage is frequently used in transfer line construction to improve output, but this can have a high cost in terms of both capital and work in progress. By examining the way performance is reduced it is hoped that transfer line design may be improved.

2.0 Description of model

In order to investigate the effects of breakdowns a simulation model has been built. To reduce the model's complexity, the line is assumed to be fully balanced and that there is no shortage of labour to repair a broken machine. Fully balanced indicates that each machine has exactly the same cycle time and that there are equal amounts of storage between each machine. Also, the supply to the first machine and removal from the last machine is always maintained. The main parameters that can be varied are the cycle time (C), number of machines (N), repair time (R), number of buffer spaces between machines (S) and the breakdown frequency (B). The output from the line in a given time is the uptime of the final machine (i.e. the percentage of total time that the line is producing).

3.0 Stoppages in the Non-buffered environment

An initial model was constructed with a maximum line length of 35 machines and no on-line storage (i.e. S=0). A breakdown pattern that gave a downtime of 10% was then used on each machine. The output from the line was then recorded for a selected runtime after a steady state situation had been reached. Fig.1a. shows how the output varies as the length of the line increases. For each length of line, the time spent busy, blocked, starved and down for the individual machines were then studied. Fig.1b. shows the variation of these times for a 30 machine line, the output corresponding to the 30 machine case in Fig.1a.

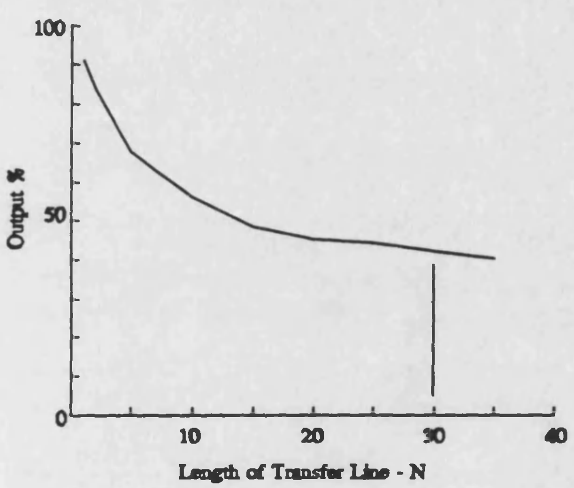


Fig.1a. Transfer Line Output

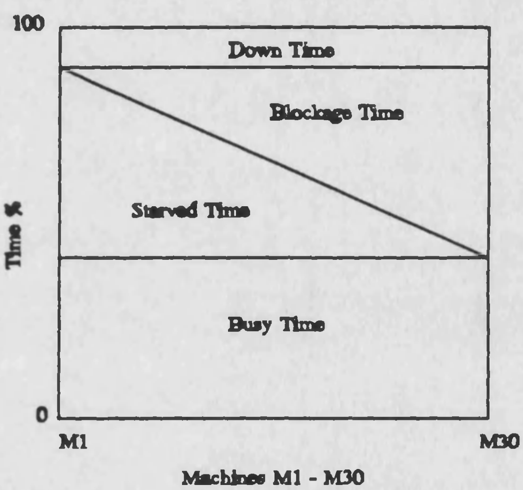


Fig.1b. Individual Machine Times

When a machine breaks down on this line, all machines upstream are unable to finish their cycles as their output is blocked. Downstream, a gap is produced as the machines are allowed to continue, but the supply route is stopped. This gap is defined as the Starvation Window, and when the broken down machine is repaired this window will flow down the line at the same rate as the components. The length of the window is equal to the number of machines that a component would have passed through in a time period equal to the repair time (= R/C).

The slope of the blockage and starvation lines can be explained as follows. When a machine stops, all upstream machines are blocked for a time equal to the repair time. If the last machine breaks down, all the machines will be blocked. If the second machine breaks down, only the first machine is blocked. Each machine will stop once in a period equal to the breakdown frequency, and so the blockage graph is built up as shown in Fig.2.

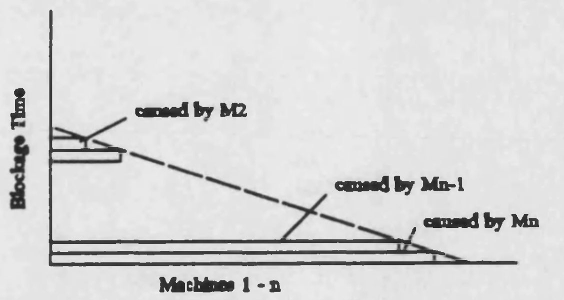


Fig.2. Effect of blockages on first machine.

The same argument goes for the starvation graph. However, instead of all the machines being starved together, each one is starved as the window produced by a breakdown passes over it for a time equal to the repair time.

By using Fig.1b. and Fig.2., it is therefore possible to calculate the theoretical output of the line. In a time period equal to the machines' breakdown frequency, all machines will breakdown once, and block all machines upstream. If we consider the first machine, this will not be starved, only blocked by the other breakdowns.

The total number of blockages will be N-1 since the first machine cannot block itself. The effect of each blockage is the same as the repair time. Thus;

Total Blockage Time on first machine = R x (N-1)

Since the first machine breaks down, it is stopped for a time R.
Therefore, Total Non-operating time = R x (N-1) + R
= R x N

As this is all in a time period equal to the breakdown frequency, B. The total non-uptime can be expressed as R x N / B x C

Thus the theoretical output efficiency = (1 - (R x N)/(B x C)) x 100%

The theoretical output gives a straight line graph, which does not match the results obtained in Fig.1a. The reason for the mis-match is the interference of blockages and starvation windows. Some of the blockage effects of machines breaking down do not reach the end of the line and effect the overall output. Starvation windows moving down the line absorb the blocking effects. Machines upstream are able to cycle as there are machines downstream that are empty and thus they are not blocked. Fig.3. shows the theoretical output for a line with a 10% downtime breakdown pattern. Also shown are the outputs from 3 different models each with a 10% downtime but with either frequent short stoppages or infrequent long stoppages.

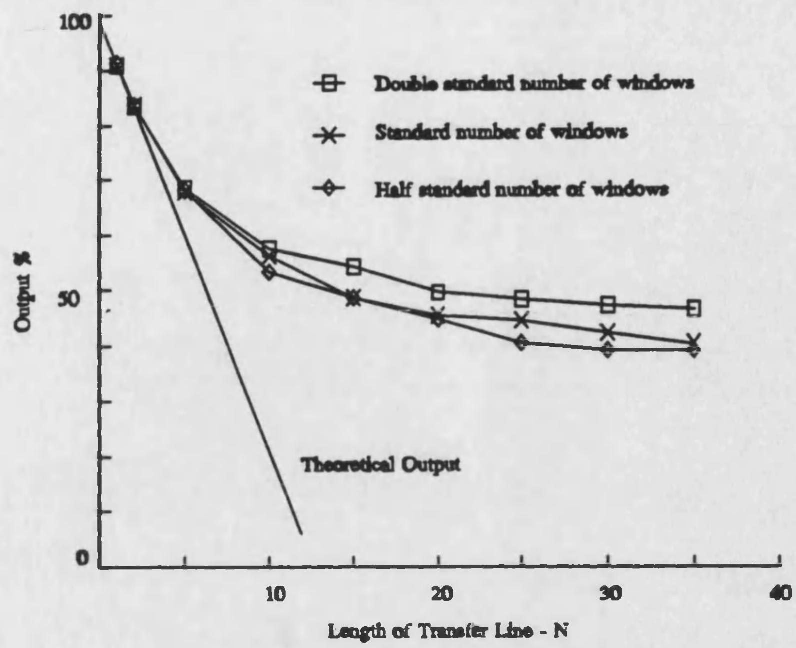


Fig.3. Comparison between Theoretical and Actual output.

The shorter more frequent stoppages can clearly be seen to give better line output. It is also worth noting how the actual output matches the theoretical output for short lines. This is because on shorter lines there is less interference as the starvation windows reach the end of the line quickly.

4.0 Stoppages in the Buffered environment

The mechanism of stoppages in the buffered environment is similar to that in the non-buffered environment. The main difference is that buffers are used to absorb blockages and starvation windows and thus localize their effects. The movement of starvation windows downstream is governed by the stock levels in the buffers through which they pass. Empty buffers will act as if the buffers were not present. Full buffers will be emptied thus halting the progress of the window.

Blockages passing upstream will fill empty and partially full buffers. If there is sufficient capacity, the blockage will not reach the end of the line. Once a buffer is full the next machine upstream will become blocked. Thus bigger buffers will tend to cause the effects of a breakdown to be restricted to fewer machines.

Buffers can be said not only to hold components but also hold room for blockages to fill and hold starvation windows. By holding windows there is an increase in the interference between blockages and the windows. Fig.4. shows how this increased interference effects the output. The bigger the buffers, the more localised the effect, and the higher the output from the line. Optimum buffer levels can thus be said to be half full, since they give an equal response to blockages and starvation windows.

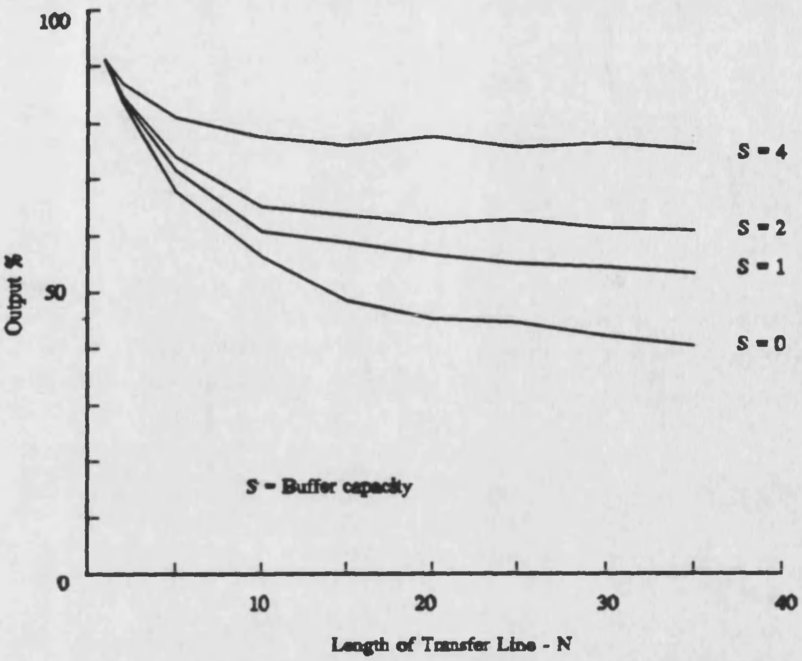


Fig.4. The effect of Buffer capacity on Line Output.

5.0 Conclusion

The output of transfer lines is reduced by the effects of machine breakdowns. Machines upstream are blocked and machines downstream are influenced by the passage of starvation windows. Breakdowns and other stoppages occur all the time, and there is an interference between the windows passing downstream and the blockages effecting machines upstream. This interference between breakdowns causes an improvement in productivity from the theoretical output of the line; the greater the level of interference the greater the output. Interference can be increased by either having more frequent, shorter breakdowns or by the use of on-line storage with capacity to contain both the effects of blockages and the effects of starvation windows. Further work will concentrate on generating windows through planned stoppages in order to compensate the effects of unplanned stoppages.

Acknowledgements

The research work carried out towards this paper has been funded by SERC.

The Use Of Buffers To Improve Transfer Line Performamce

G.W. Owen and A.R. Mileham

School Of Mech Eng, The University Of Bath.

Abstract

Transfer Lines typically consist of many automatically linked machines, and are used to machine components with a high demand. Machine unreliability due to breakdowns and tool changes greatly reduce the output levels of lines. Interstage storage or buffering is commonly used when operating lines in an attempt to isolate the effects of stoppages and irregularities in cycle time, in order to maximise output. This paper studies the optimum buffer distribution on balanced non-synchronous lines and examines the way buffers increase interference between blockages and starvation windows introduced by stoppages and hence output. Also studied is the effect that the presence of a bottleneck machine has on output, and how buffering distributions can be changed to minimise their effect and maximise output. The results of simulations employing various buffering distributions within a representative transfer line are presented.

1.0 Introduction

Machine stoppages caused by breakdowns and for tool changes greatly reduce the output of transfer lines. Not only is the stopped machine affected, but machines following the one stopped (downstream) are no longer supplied (starvation) and those preceding the one stopped (upstream) can no longer cycle (blocked). When building transfer lines, buffers are placed between consecutive machines in a belief that stores of components help isolate the effect of stoppages and hence increase output. Previous work by the authors [Owen & Mileham 1991] describes how interference between blockage patterns and starvation windows caused by stoppages on non-synchronous lines reduces the overall effect on output of each stoppage. Also discussed is the way that improvements in output made by adding buffers between machines are due to increased levels of interference caused by retarding the movement of starvation windows, as opposed to the traditional view of buffers as component stores.

In order to increase interference levels, the positioning of buffers may be more important than the actual size of buffers used [Buzzacott 1971]. It is hoped that by studying this area in detail, transfer line design and hence output can be improved. A large portion of previous work in this area has been analytical [Buzzacott 1967],[Hillier & So 1991]. These analytical studies, however, concentrate on short lines (3,4 and 5 stations) which does not relate to the use of transfer lines in industry. As with the previous work by the authors, the work on which this paper is based has been carried out using computer simulations, using the WITNESS simulation package. This has allowed concentration on long lines, 30 machines being typical, where buffering is more commonly used.

At this stage it is important to define exactly what buffers are. Buffering is the ability to store components between sequential operations. A buffer can take many physical forms, from a stack of parts on a pallet out in the yard to, more commonly in the case of a transfer line, a fully automated racking system feeding from and returning to a conveyor. The capacity of such a racking system is measured in terms of the number of components which it can store when full. This is known as the buffer size or the number of buffer spaces. A useful measure of the level of buffering on a given line is the average number of buffer spaces per machine, S. Where S = Total buffer spaces on the line / Number of machines

2.0 Buffers on a balanced non-synchronous lines

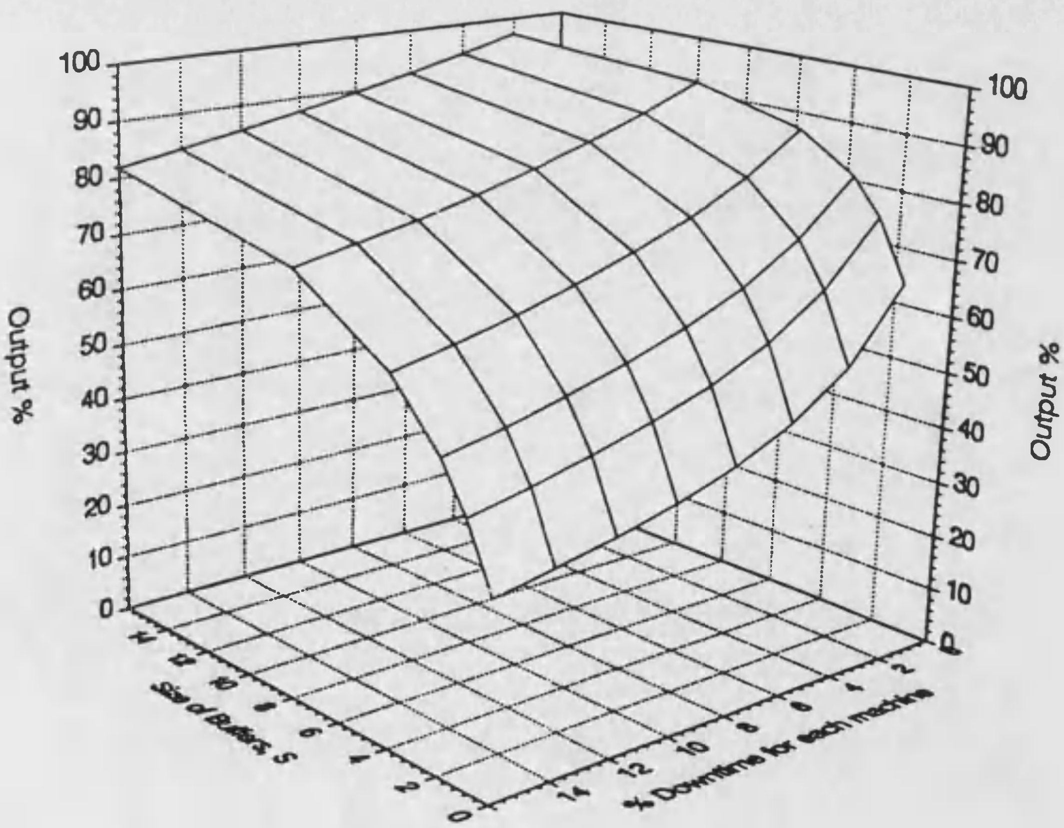


Figure 1 - The effect on output of different buffer levels on a Transfer Line for varying % Downtimes

The amount of buffering placed on the line greatly affects output. Figure 1 shows the results of a series of simulations carried out on a fully balanced line of 30 machines for different %downtimes. Equal sized buffers were placed between each machine (an even buffer distribution). The gains in output levels depending on the size of these buffers can clearly be seen. The addition of buffers between machines is, however, a law of diminishing returns. The improvement in output from the line caused by the addition of an extra buffer space is less the more buffering there is on the line. Since buffering has with it an associated cost, not only a significant capital equipment cost but also an increase in WIP (and hence throughput time), there is obviously a breakeven point between the cost of buffering and the gains made.

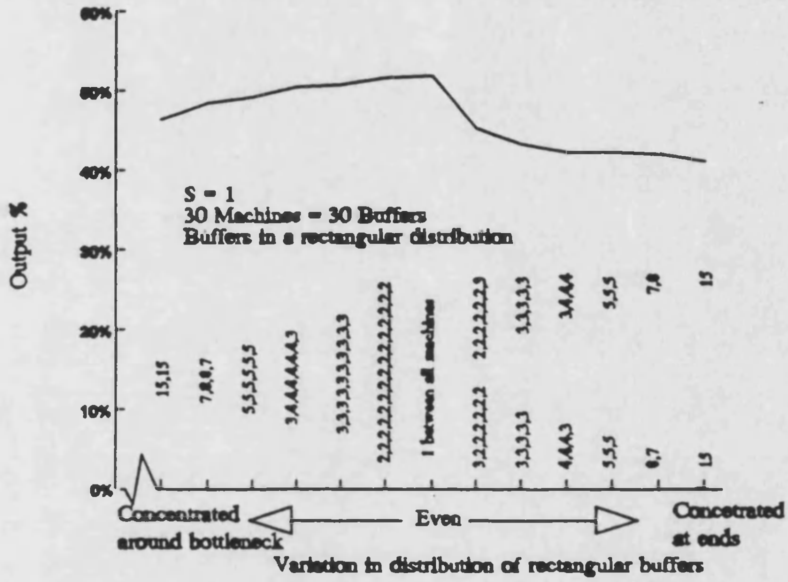


Figure 2 - Variation in output on a balanced line with different rectangular buffer distributions

Buffers, however, need not be spaced equally along the line. The effect of different buffer distributions, again on a balanced line 30 machine long, are shown in Figure 2. For each of the different distributions there are a total of 30 buffer spaces (i.e. S=1). Not all machines have buffers in between them. The buffers that are placed are said to be in a rectangular distribution, i.e. where there are buffers, there sizes are equal (e.g. 2 buffers x 15 spaces, or 6 x 5, or 10 x 3 etc). Where the buffers have been concentrated in the centre of the line, 15,15 represents 2 buffers of 15 spaces each between machines M14 & M15 and M15 & M16. Where buffers have been concentrated to the outside of the line 15---15 represents buffers of 15 between M1 & M2 and M29 & M30 and so on.

It can clearly be seen that an even spread of buffer spaces along the line gives higher output than either concentrating buffers at the ends or in the middle. This is to be expected since to maximise output, buffers are placed where they will maximise the retardation of the movement of starvation windows and blockages produced by the stoppages. As breakdowns are produced evenly along the line, even buffering will achieve this best.

3.0 Buffer distributions on un-balanced lines

Assuming a balanced line with even cycle time and machine unreliability, bottlenecks are caused by increased unreliability in terms of increased repair times, more frequent stoppages or a combination of both. During the work described in this paper the breakdown frequency has been used as the variable and the repair time kept constant. The effect of even buffering on a line with a single bottleneck is much the same as on the balanced line described above. The effects of rectangular buffer distributions are, however, different as can be seen from Figure 3. Concentrating buffering at the bottleneck which in this case is at the centre of the line gives a higher line output than either even buffering or concentrating the buffering at the outside of the line. In this case the highest output is achieved with 10 buffers each of 3 spaces between machines M10 to M20. Again this result is to be expected as it is concentrating buffers nearer the origins of the majority of the breakdown effects which in turn maximises the retardation of the starvation windows.

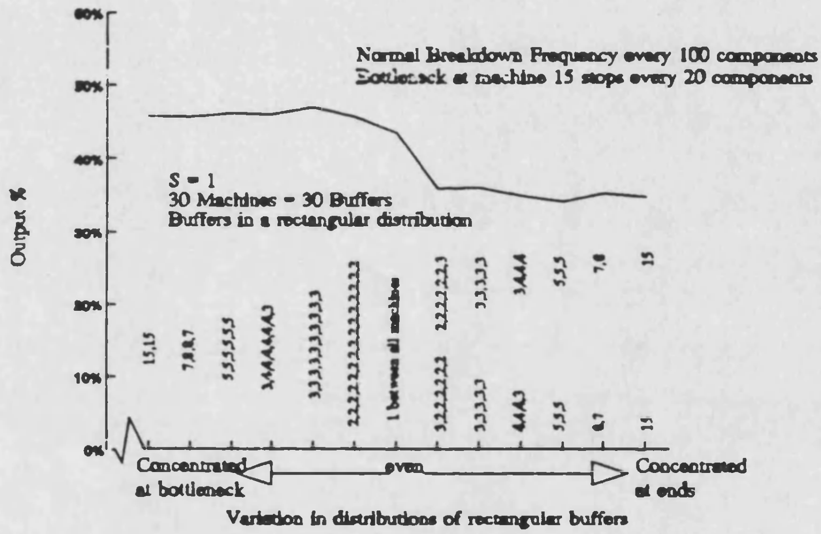


Figure 3 - Variation in output on a bottlenecked line with different rectangular buffer distributions

There is no reason why rectangular buffer distributions should give the maximum output for a given level of buffering. In order to determine the optimum buffering distribution a series of simulations have been carried out on transfer lines with different breakdown patterns. All the lines were 30 machines long with identical unreliability except for a bottleneck caused by higher unreliability at machine 15. Figure 4 shows the difference between the build up of the optimum buffering for two different lines.

The contrast between the way the optimum buffering builds up as the amount of buffering increases from S=0.07 to 0.60 is quite apparent. Clearly the more restrictive the bottleneck the more concentrated the buffering needs to be around it and the nearer the unreliability of the bottleneck is to the general level of unreliability the more even the spread of buffers required.

Line 1

Mean machine breakdown frequency - every 500 components except bottleneck machine - every 50 components. Stoppage time - 10 cycle times.

Total number of buffer spaces	Build up of buffering around the bottleneck
2	1 1
4	2 2
6	3 3
8	4 4
10	5 5
12	1 5 5 1
14	2 5 5 2
16	1 - - 2 5 5 2 - - 1

Line 2

Mean machine breakdown frequency - every 100 components except bottleneck machine - every 50 components. Stoppage time - 10 cycle times.

Total number of buffer spaces	Build up of buffering around the bottleneck
2	1 1
4	1 - - 1 1 - - 1
6	1 - 1 1 1 1 - 1
8	1 1 1 1 1 1 1 1
10	1 1 1 1 1 1 1 1 1 1

Figure 4 - Difference in buffer distribution build up on 2 bottlenecked lines

Results obtained from the study of other lines suggest that the choice of buffering pattern can be made using data on breakdown distributions, provided repair time is constant for all machines. The decision on whether to use concentrated buffering as in Line 1 in Figure 4, or even buffering as in Line 2 is dependant on the difference between the breakdown frequencies of the bottleneck machine and the machines around it. There appears to be no dependence on the overall downtime, nor the ratio between the breakdown frequencies. It is hoped that further study in this area will produce a set of rules which will allow the optimum buffer distribution for negating the effects of unreliability to be built into the line at the design stage.

4.0 Conclusion

Machine unreliability greatly reduces the output of transfer lines. Placing buffering between machines can reduce the effect and hence improve output. The positioning of the buffers does, however, greatly affect the gains made in output. On balanced lines, the optimum distribution is an even spread of buffers between each machine. Where lines have a bottleneck, buffers should be near the bottleneck. The exact distribution around the bottleneck is, however, dependant on the size of the bottleneck. The larger the bottleneck compared with the general level of unreliablity the more the buffering should be concentrated around it. Further work will concentrate on generating a set of rules on how transfer lines should be buffered both for breakdowns and fixed frequency tool changes.

References

1. G.W.Owen & Dr.A.R.Mileham, 'The influence of machine failure on transfer line performance', 7th Nat. Con. on Prod. Research, Hatfield Poly. Sept. 1991
2. Buzzacott, J.A., 'The role of inventary banks in flow line production systems.', International Journal of Production Reasearch(IJPR) 1971 Vol 9
3. Buzzacott, J.A., 'Autiomatic transfer lines with buffer stocks.', IJPR 1967 Vol 5
4. Hillier, F.S., So, K.C., 'The effect of machine breakdowns and interstage storrage on the performance of production lines.', International Journal of Production Research 1991 Vol 29

A Method for Buffering Transfer Lines to Maximise Output

G. W. Owen & Dr. A. R. Mileham,
Department of Mechanical Engineering,
University of Bath,
Bath, UK

ABSTRACT

Machine stoppages due to tool changes and breakdowns can greatly reduce the output of non-synchronous transfer Lines. In order to maximise output, the effects of these stoppages can be isolated from the rest of the line using interstage storage or buffering. This paper describes a method by which a near optimum buffering pattern can be quickly developed for a given line. The technique involves using computer simulation to study the time each machine spends blocked or starved as a result of other machine stoppages. Comparisons with other buffering strategies are also presented.

1. INTRODUCTION

Transfer lines typically consist of many automatically linked machines, and are used to machine components with a constant, high demand. Machine stoppages caused by breakdowns and for tool changes greatly reduce the output of transfer lines. When a machine stops, the rest of the line can be affected. The machines downstream (after the stopped machine) will also stop as they become starved of workpieces. Upstream (before the stopped machine) the output of the machines will become blocked and these machines are also forced to stop through no fault of their own.

In order to improve output, stores of components (buffers) are placed in between machines in a believe that this isolates the stoppage effects. Previous work by the authors (Owen & Mileham, 1991, 1992) describes how interference between the blocking and starving affects caused by stoppages on non-synchronous lines reduces the overall affect on output of stoppages. Also discussed is the way that additional buffers between machines improve output by increasing the levels of interference by retarding the movement of starvation windows, as opposed to the more traditional view of buffers as being component stores.

At this stage it is important to define 'a buffer'. Buffering is the ability to store components between sequential operations. Although the most common forms on transfer lines are automatic racking and queuing conveyors, a pallet out in the yard is an equally valid form. The capacity of any system is measured in terms of the number of components which it can store when full. This is known as the buffer size, the buffer capacity or the number of buffer spaces.

The positioning of any buffering placed on the line has been the subject of a limited amount of research (Buzzacott, 1967, 1971; Yamashina & Okamura, 1983; Conway, 1988). Most of this research has, however, been analytically based, and because of this has had to concentrate on short lines (up to 5 machines). The work has given a series of rules, some of which are conflicting, on how to place buffers on transfer lines. These include :-

- The size of any buffer must be in multiples of the mean repair time of the line (Buzzacott, 1967);

- The buffer capacity at any one point should never be greater than 5 times the mean repair time (Buzzacott, 1967);

- 5 stages (4 buffers) is the most a line should be split into (Buzzacott, 1967);

- The line can be separated at any point provided the 'Bowl Effect' of having the least reliable machines at the end of the line is maintained (Yamashina & Okamura, 1983);

- Allocate buffer capacity as nearly equally as possible. If, after equal allocation, some extra buffer capacity is available, spread it over the line at approximately equal intervals. The first and last buffer should get the lowest priority in this step (Conway, 1988);

- Central buffers are given the highest allocation and nearly equal but diminishing allocations are successively made to the outer buffers. This is an inverse bowl or triangular allocation pattern (Yamashina & Okamura, 1983);

It is clear that some rules seem far removed from typical line designs. The idea of designing a line with the least reliable machines at the end in particular. Surely lines must be designed by process planning constrains, rather than by machine reliability. Also Buzzacott (1967) claims that lines should not be split into more than 4 stages. Yet a whole production system is, in effect, one long line, but you would not attempt to make a car in as few as 4 stages. These are typical of the shortcomings of analytical work.

Since the improvements that can be made in line output by adding buffers can be significant, bottlenecks and how to buffer them appears to be a key area for further research. It is hoped that by studying this area in detail, transfer line design can be improved. As with the previous work by the authors, the work on which this paper is based has been carried out using computer simulations, using the WITNESS simulation package. This has allowed concentration on long lines, up to 40 machines being typical, on which buffering is more commonly used in industry.

2. THE EFFECT OF STOPPAGES AND BUFFERS ON LINE OUTPUT.

The effect of machine stoppages can be clearly seen from Figure 1. With no buffers (s=0) on a balanced line where each machine has a mean down time of 8%, the longer the line

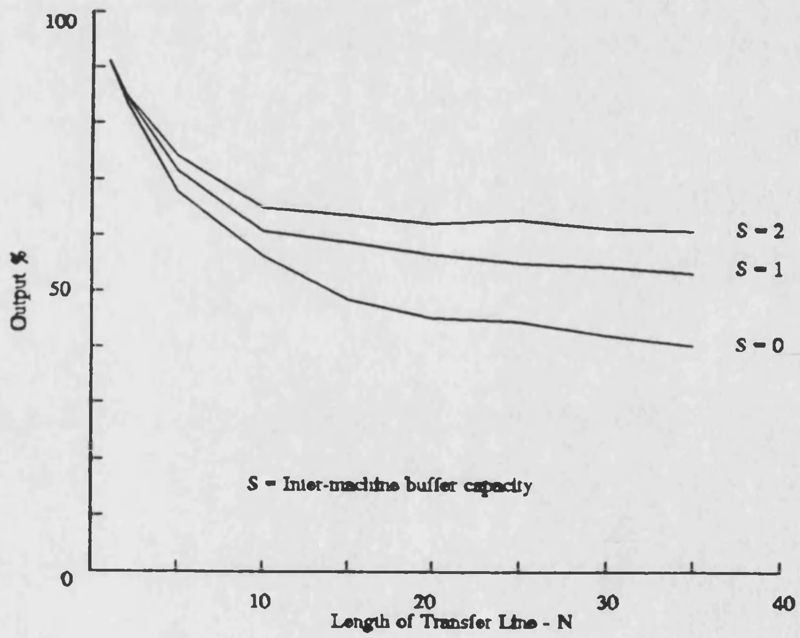


Figure 1. The effect of line length and buffer capacity on output.

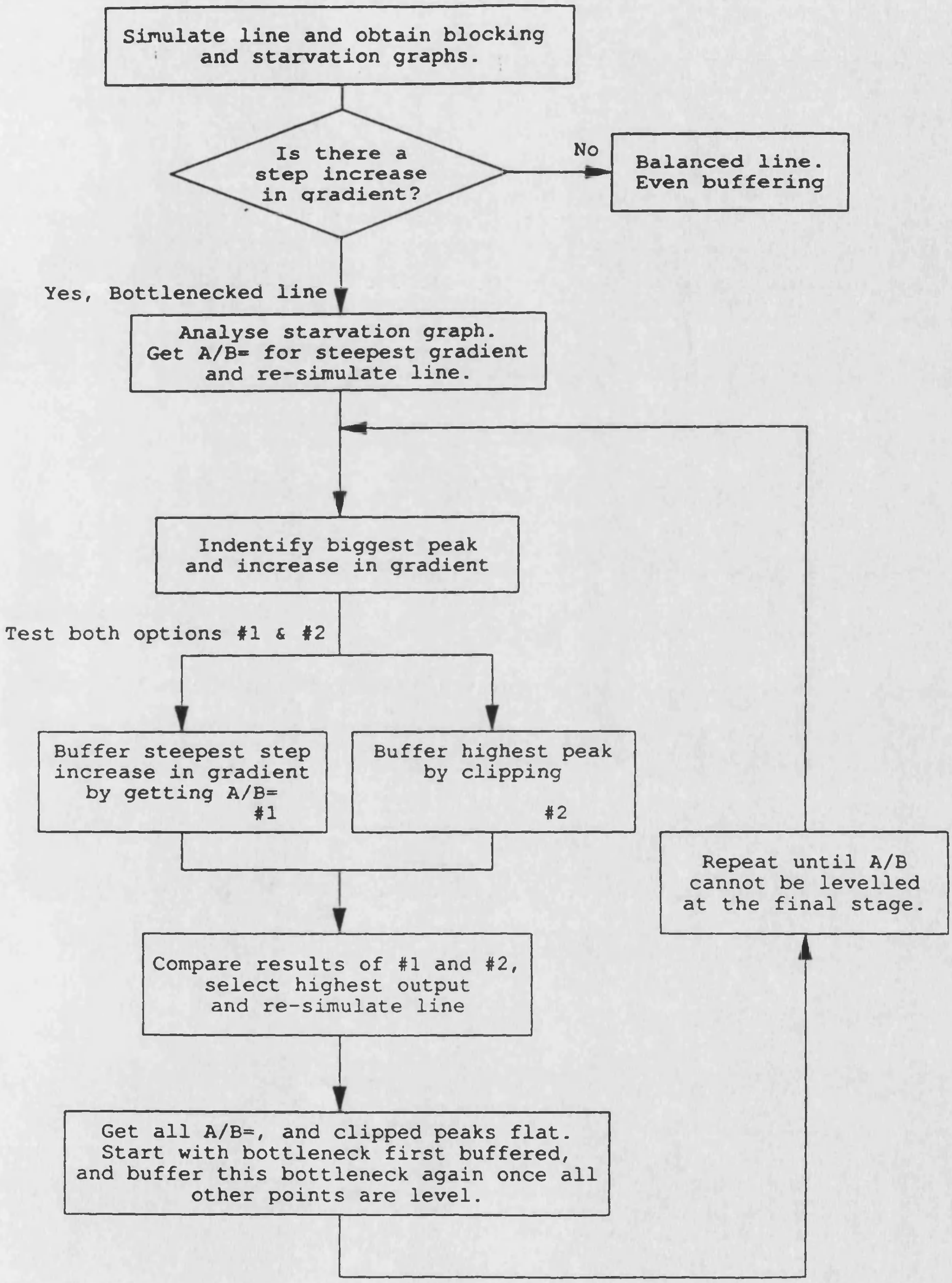


Figure 2. A flowchart of the methodology used to buffer a transfer line.

the lower the output, with a 29 machine long line having an uptime of less than 50%. If equal buffers are placed between the machines, the output increases, but adding the buffers is a law of diminishing returns (the gains made by adding each successive buffer get smaller as they are added). The placement of an even buffer pattern is, however, only an example of the gains that can be made through adding buffers to a line. In Section 4, it will be shown that for a bottlenecked line this pattern is far from ideal.

3. A BUFFERING METHODOLOGY

The following set of heuristics, with the aid of a flow chart (Figure 2), are used to place 'near optimum' buffers patterns on non-synchronous transfer lines. The lines output can never be greater than that of the least reliable machine, and the object of the buffering is to maximise output by preventing the bottleneck machine being blocked and starved. In order to carry out the analysis it is necessary to have a simulation model of the line, which when run will record the 4 main machine states (Busy, Broken Down, Blocked and Starved). The analysis is carried out by

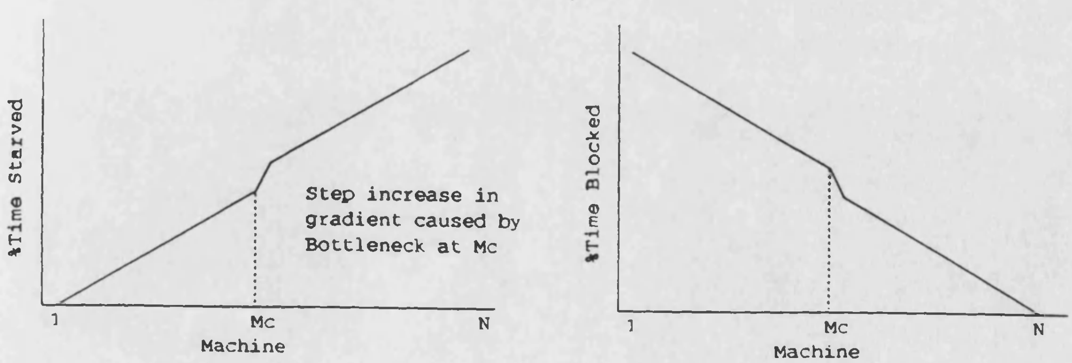


Figure 3. The starvation and blockage curves of a line with a bottleneck at Mc.

studying graphs of the amount of time each machine spends blocked and starved (for example Figure 3). The technique does not allow for any measure of cost effectiveness but the comparison between cost of buffering and value of the gains in output over time can be easily made.

Although many iterations through the methodology of adding single buffer spaces to the line are required, in practice, and with experience, it is possible to take 'short cuts' by adding buffering to several points at once to correspond to their proportional effects on the line. To buffer a line, the steps in the flow chart need to be followed and repeated until the bottleneck machine is neither blocked or starved or the required output is reached.

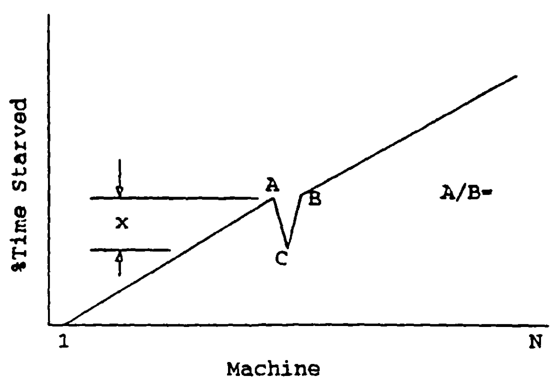


Figure 4. Starvation curve after buffering to get A/B=.

The following points expand the steps outlined in the flowchart:

- Bottlenecks can by identified by step changes in the gradient of the blocking and starvation curves (see Figure 3). The biggest bottleneck will be represented by the biggest step change. If there are no step changes, the line is balanced and should be buffered evenly.

- To buffer a bottleneck, buffers are placed either side of the bottleneck machine (M_c in Figure 3) in proportion to the amount of time spent blocked and starved. A bottleneck machine which is predominantly starved has buffers placed in front of it, a blocked machine has buffers placed after it. As buffering is added around the machine, point A tends to rise slowly, while point B falls more quickly. When the two points, A and B, are level, the bottleneck machine is not contributing to line disruption. In short hand this is written as A/B= (said A and B level). The resulting starvation curve is shown in Figure 4 and the peak produced is defined as having a height x.

- After A/B= is achieved by buffering around M_c, to gain further increases in output, equal buffers should be placed along the line away from M_c. This will remove the top of the peak as shown in Figure 5 and is referred to as clipping the peak. When two equal sized peaks are found on the line, the peak to be clipped is determined by whether the machine spends more time blocked or starved. In the case of a predominantly starved machine, buffers are placed upstream at M_{c-1}, M_{c-2} etc. The buffering should extend along the line until a level plateau is achieved as shown in Figure 5 with a resulting peak height z. In both cases A/B will no longer be level.

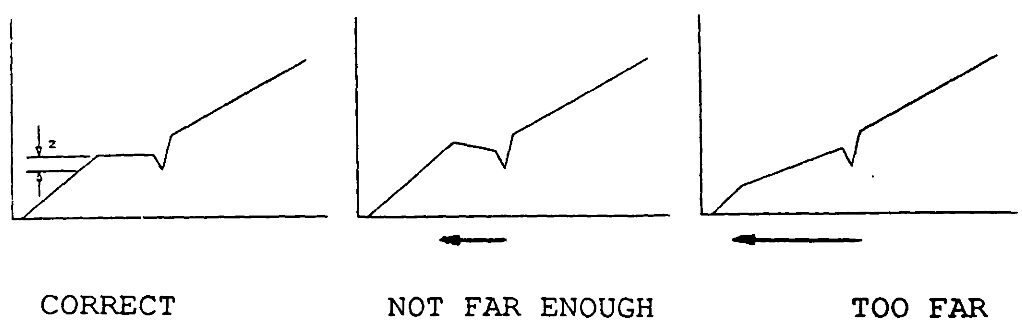


Figure 5. Starvation curves of a peak being 'clipped'.

- On complex lines it is often not clear whether to clip the next peak or buffer a step increase in gradient. In this case both options should be tested. In order to compare the results interpolation between the gains of different buffering levels has to be made.

- During the final stage of the flow chart, it may be necessary to keep adding more buffers many times over (particularly on complex lines), as completing the buffering at one point effects all the other points. When doing this the main bottleneck (which was buffered first to begin with and which has the biggest %downtime) must always be re-buffered first, as once it is no longer blocked and starved the output is maximised. When reaching the point of maximum output, the buffering will tend to become unstable and the final step of getting A/B= at the main bottleneck will not be possible.

4. AN EXAMPLE ON A SINGLE BOTTLENECK LINE

Consider a line of 29 machines. The cycle time is balanced, all machines except the middle machine (M15) have the same mean breakdown frequency and repair times. The middle machine is the bottleneck with a breakdown frequency 5 times as high as the other machines. The line is symmetrical and as such the buffering pattern is identical on both sides of the bottleneck machine. Figure 6 shows the change in the starvation curve as the line is buffered using the developed methodology. The blockage curve is a mirror image of this.

Step 1 - Identify the steepest point and buffer until A/B=, this is achieved in both the blockage and starvation curves with 6 buffers either side of M15, as shown in Figure 7. together with the output levels of the various stages. (Output 70.2%)

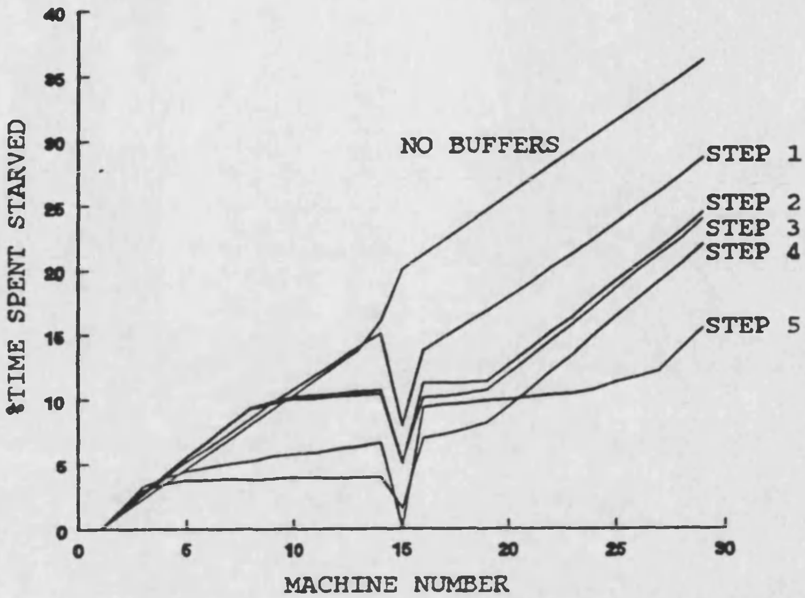


Fig. 6. The effect of the buffering methodology on the starvation curve.

Step 2 - Clip peaks. Both peaks are clipped using a row of 4 buffers up to M15. (74.0%)

Step 3 - A/B are levelled resulting in a buffer pattern around M15 of 1 1 1 1 8 8 1 1 1 1. (74.8%)

Step 4 - The peaks are again clipped, now using 2 buffers between each machine. Where 2 buffers extend from M4 to M14 and also from M17 to M27. (81.38%)

Step 5 - The final stage is to try and get A/B= again. Buffers are paced either side of M15, but instead of achieving this, the blockage and starvation of M15 falls to 0%. The maximum output of the line is, therefore, achieved (83.4%).

This final 'near optimum' buffering pattern can be seen in Figure 7 together with other buffering patterns using the same number of buffers. The gains to be made through having the correctly placed buffering can clearly be seen. At the maximum output (83.4%), the 'near optimum' buffer pattern produced requires 15-20% less buffer spaces than any other pattern, or alternatively, the output is 1.5 higher than for the same amount of buffering using any other published strategy.

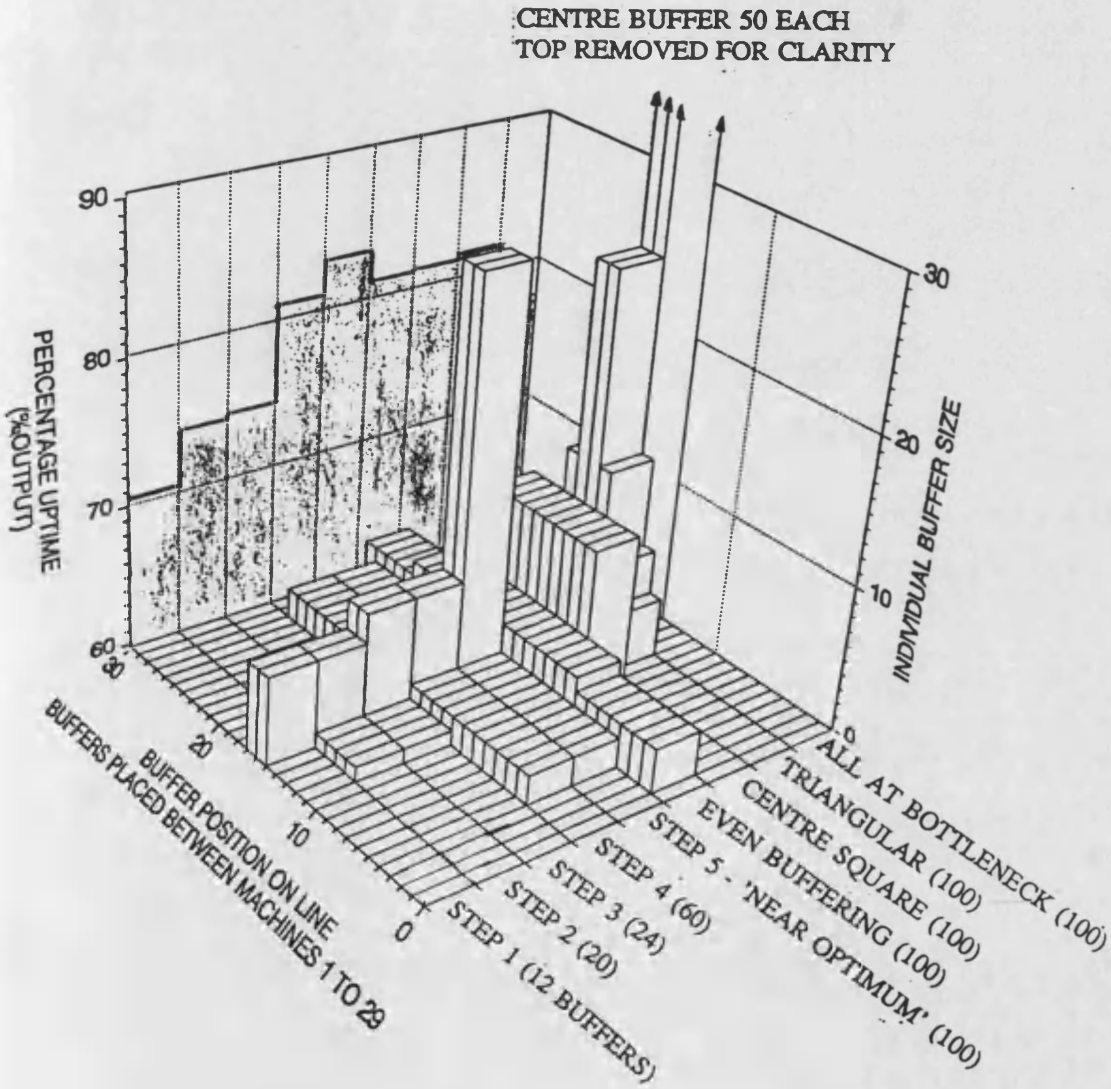


Fig. 7. Graph showing the build up of the 'near optimum' buffer pattern, together with a comparison in the output between this and other strategies.

5. CONCLUSION

Buffers can greatly increase the output of non-synchronous transfer lines. The positioning of the buffering on the line is important. Previous research has generated conflicting ideas on how this should be done. This paper details a set of heuristics to allow a 'near optimum' buffer pattern to be easily generated using the results of computer simulation. The resulting buffering compares favourably with other buffering patterns.

REFERENCES

Buzzacott, J.A. (1967). Automatic transfer lines with buffer stocks. Int. J. Prod. Eng. 5 183-200

Conway, R. (1988). The Role of Work In Progress Inventory In Serial Production Lines. Operations Research 36, 229-241

Owen, G. W., Mileham, Dr. A. R. (1991). The Influence of Machine Failure on Transfer Line Performance. 7th Nat. Con. on Production Research, Hatfield, UK

Owen, G. W., Mileham, Dr. A. R. (1992). The use of Buffers to Improve Transfer Line Performance. 8th Nat. Con. on Production Research, Birmingham, UK

Yamashina, H., Okamura, K. (1983). Analysing in-process buffer for multiple transfer line systems. Int. J. Prod. Eng. 21, 183-195