MASTER

A comparison of batch production on a flow line and job shop assembly in a semiconductor back-end

van Rhee, E.G.M.

Award date:
2011

Technische Universiteit
Eindhoven
University of Technology

Department of Mathematics and Computer Science
Den Dolech 2, 5612 AZ Eindhoven
P.O. Box 513, 5600 MB Eindhoven
The Netherlands
www.tue.nl

Supervisors
prof.dr.ir. I.J.B.F. Adan (TU/e)
dr.ir. M.A.A. Boon (TU/e)
ir. J.P.W. Stokkermans (NXP)

In cooperation with
NXP Semiconductors

Date
August 5, 2011

Version
3.4

A comparison of batch production on a flow line and job shop assembly in a semiconductor back-end

Master's thesis

E.G.M. van Rhee

Abstract

NXP Semiconductors is one of the world's largest suppliers of high performance semiconductor devices. At present, semiconductor devices are used in virtually any electronic equipment. Therefore, the demand for chips keeps on rising. However, due to the high costs of the equipment used in a semiconductor fabrication plant, it is not feasible to build new plants in order to increase production. Hence, every semiconductor company is challenged to find other ways to increase their production and reduce costs. ITEC, one of the departments of NXP Semiconductors Nijmegen, focuses on improving the equipment and processes used for the device assembly of discrete semiconductors in the back-end process of all NXP's fabrication plants. At present, this assembly usually takes place on a production line in which the machines are interconnected by a reel of leadframe containing the foundation for eventual products. Production can possibly be enhanced by changing the line configuration from a rigid flow line to a flexible job shop.

The main goal of this master's thesis is to compare two different production concepts (flow line and job shop) and get insight in their characteristics and performance based on several key performance indicators. Furthermore, NXP wants to get insight into the circumstances under which one production concept is to prefer over the other. Since the back-end process is too complicated and has many details, it does not allow an exact, or even good approximate, analytical evaluation. Hence, a production simulator has been developed in MATLAB to compare both production concepts. This simulator will, for NXP, serve as a reference framework for future decision making. Since a job shop is very flexible compared to a flow line, some straightforward algorithms have been implemented for the routing and splitting of jobs at the different machines in the network. If NXP Semiconductors decides to switch from a flow line to a job shop configuration, more complicated scheduling and routing algorithms can easily be implemented in the MATLAB simulation program. In the end, a gain of 50% in line throughput can be achieved for some products when switching from flow line to job shop due to poor machine utilization in a flow line. In contrast, in a fully balanced flow line with short and less frequent downtimes and no changeovers, the relative difference in throughput between a flow line and a job shop decreases to 3%, again in favor of the job shop. Due to the infinite buffers and the flexible routing, the line throughput in a job shop will, in general, not be smaller than the line throughput in a flow line. However, a job shop does require more employees and a better planning. It is further left to NXP's management to decide whether the possible gain in line throughput outweighs the increase of costs and research costs.

Contents

CHAPTER 1

Introduction

This chapter gives a short introduction to NXP Semiconductors and the processes involved in the creation of a semiconductor device. Furthermore, the problem description of this thesis is stated and at the end of this chapter the global outline of this thesis is given. Throughout this thesis several company specific names, definitions and abbreviations are used; these terms are be explained the first time and most of them are also briefly explained in Appendix A.

1.1 NXP Semiconductors

The research in this thesis was performed for NXP Semiconductors Nijmegen, the ITEC department. Since NXP is a relatively young company, this section will briefly describe the company NXP, the products they produce and the ITEC department.

1.1.1 History

The invention of the integrated circuit in mid-20th-century has had a big influence on modern society. These integrated circuits are now being used in many consumer products such as computers, mobile phones, FM-radios and navigation equipment. One of the companies that became very successful in the fabrication of these electronic products was Philips. Philips started fabricating semiconductor devices in 1953 and quickly became one of the big competitors in this industry. After more than 50 years of innovation, Philips announced the split off of its semiconductor division and in 2006 this division became a separate legal entity: NXP Semiconductors. Ever since, NXP has established large fabrication plants in more than 25 countries all over the world and remained one of the world's largest suppliers of high performance semiconductor devices.

1.1.2 Semiconductor devices

A semiconductor device is a component of an electronic circuit made from a material that is neither a conductor, nor an insulator, hence the name semiconductor. The fabrication of semiconductor devices is a combination of different hi-tech processes. These processes can roughly be divided in the following parts:

1. Wafer preparation

Figure 1.1: The NXP Semiconductors ISN-6 fabrication plant in Nijmegen [1].

2. Wafer processing
3. Semiconductor device assembly
4. IC testing

Whereas the first two steps are performed in clean rooms, the last two steps, also known as the back-end process, do not require an utterly clean environment and can be performed in regular factory halls. At present, semiconductor devices are used in virtually every electronic equipment. As a result, the demand for chips keeps on rising. However, due to the tremendously high costs of the equipment used in a semiconductor fabrication plant, it is not feasible to build new factories in order to increase production. Hence, every semiconductor company is challenged to find other ways to increase their production and reduce costs. ITEC, one of the departments of NXP Semiconductors Nijmegen, focuses on improving the equipment and processes used for the device assembly of discrete semiconductors in the semiconductor fabrication back-end for many fabrication plants. Line throughput can be increased in many different ways, e.g. by efficiency improvements such as error reduction, buffer enhancement or even an alternative production line configuration. ITEC's mission is to investigate all these possibilities and provide best in class assembly solutions.

1.1.3 Semiconductor device assembly

Once the wafer is fully processed in the clean room and the dies on the wafer have been cut (see Figure 1.2 for a picture of a cut wafer), different dies can be placed and connected to the product carrier called leadframe. This assembly process usually consists of the following three stages:

Die bonding Several dies are, one by one, picked up from the wafer and mounted on the product carrier. Either a eutectic solder, or an epoxy glue is used to bond the die to the leadframe. This process always takes place on so-called ADAT machines (abbreviated with A). In this thesis, the die bonding always is the first process.

Wire bonding Once all dies are attached to the product carrier, the bond-pads on the die have to be connected with the contact points on the leadframe. For this connection pure gold wire with a thickness of just 18, 23, 35 or 50μm is used. The wire bonding takes place on so-called PHICOM machines (abbreviated with P).

Moulding When all dies and wires have been placed, they are encapsulated with a plastic or epoxy cover in order to protect the semiconductor device against environmental influences. This encapsulation takes place on a Multi Plunger (abbreviated with MP). For this research project, the moulding always is the last step.

All these processes take place on a production line consisting of several ADATs, PHICOMs and one Multi Plunger. At present, the machines in a production line are interconnected by a large reel, consisting of approximately 500 meters, of double tracked leadframe; intermediate buffers are created by diverting the leadframe between two machines over a certain number of pulleys. A bare piece of leadframe and a magnification of one product position are depicted in Figures 1.3 and 1.4. Once a reel is depleted, or once it needs to be changed, a new reel is loaded at the first machine (A1) and manually fed through the subsequent machines, this makes the production line into a flow line. The leadframe itself has two purposes, not only is it used to transport the product from one machine to another, but parts of it are also used in the eventual product. When the dies, wires and part of the leadframe are encapsulated by the Multi Plunger, the leadframe will, in another process, be cut in such a way that parts of the leadframe still stick out of the cover. These parts of the leadframe are eventually used to connect the semiconductor device in an electrical circuit. The leadframe sticking out from the encapsulation can clearly be seen in Figure 1.5, depicting a final product from the SOT457 product family.

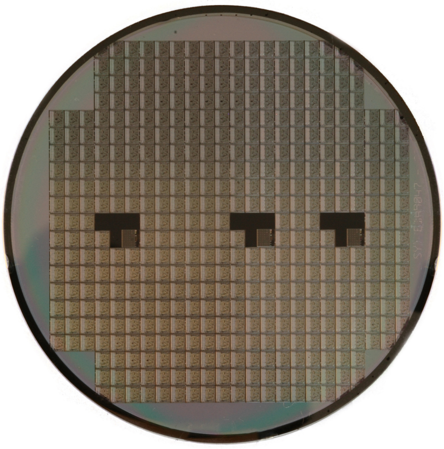


Figure 1.2: A wafer with dies on it, note that some dies were already removed [2].

1.1.4 Assembly line configuration

Figure 1.6 depicts a typical assembly line with 3 ADATs, 4 PHICOMs and 1 Multi Plunger, with intermediate buffers, for all the products in the SOT457 package. One of the big disadvantages of such a flow line is its lack in flexibility. For example, every single product will visit exactly 3 ADATs, 4 PHICOMs and 1 Multi Plunger, but for some products 2 ADATs, 3 PHICOMs, and 1 MP would be suffi-

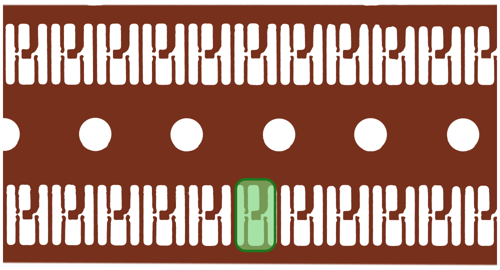


Figure 1.3: A small piece (3 cm) of double tracked leadframe, one product position is highlighted in green.

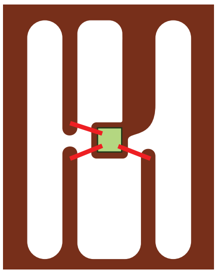


Figure 1.4: An example of one single product position containing one die (green) and 3 wires (red).

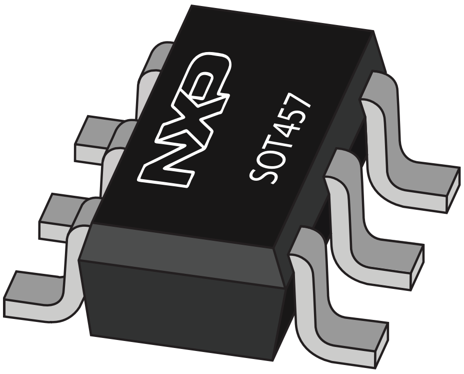


Figure 1.5: One of the products in the SOT457 package [3].

cient as well. In such a specific case, 1 ADAT and 1 PHICOM are in fact idling and the leadframe will pass through the machine without being processed. Hence, the utilization for these machines will be lower and due to financial depreciation the price of these specific products will rise or the profit margin will decrease. This unwanted effect may be prevented by changing the line configuration from a rigid flow line to a flexible job shop, as depicted in Figure 1.7. In a job shop the machines are no longer interconnected and the reel of double tracked leadframe is replaced by cassettes filled with 20 up to 40 small strips of leadframe, each containing 14 × 40 products. These cassettes will then be routed through the network and if a product only requires 2 ADATs, then those cassettes will only visit 2 ADATs and skip the third one. That third ADAT can, in a job shop, be used to process another order, hence, the machine utilization, and therefore also the throughput and profit margin, may increase. However, this is not the only advantage of a job shop compared to a flow line. In the flow line there are intermediate finite buffers and due to both the high cost and the lack of space in assembly halls these buffers can only store up to 2250 products if the leadframe is diverted once, and roughly 5000 products when the leadframe is diverted twice. Considering that theoretical throughput of an average production line is about 24000 products per hour, these buffer levels are equivalent to 6 respectively 12 minutes of unhampered production. However, in a job shop the cassettes can easily be stacked in a machine's auto feeder and since one cassette stores as many as 22560 products, the buffer levels greatly increase. In fact, one can regard the buffer capacity in a job shop as infinite since it is always possible to store a cassette near a machine. Hence, the total line throughput might again increase. The job shop also has some disadvantages. For example, one has to manually transport a cassette from one machine to another and although the total transportation time is negligible compared to the average service time, one still needs several operators taking care of this transportation process.

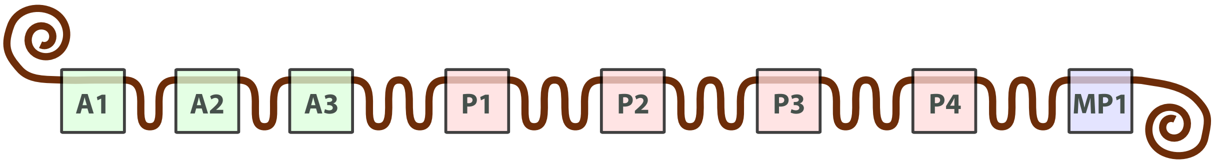


Figure 1.6: Machine layout in a flow line with an endless leadframe that connects the machines. The buffers between machines are created by diverting the leadframe over several pulleys.

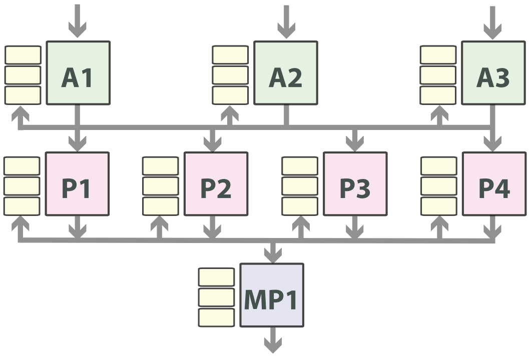


Figure 1.7: Machine layout in a job shop with cassettes that are routed through the network.

1.2 Thesis outline

The main goal of this master's thesis is to compare the two different production concepts (flow line and job shop) and get insight in their characteristics and performance measures in such as line throughput. Since the back-end process is too complicated and has many details, it does not allow an exact, or even good approximate, analytical evaluation. Hence, a production simulator has been developed to compare both production concepts. This simulator will, for NXP, serve as a reference framework for future decision making in whether or not to change the current flow lines into job shops. Due to the infinite buffers and the flexible routing, the line throughput in a job shop will, in general, not be smaller than the line throughput in a flow line. Hence, the intention of this research is not only to confirm (or reject) this idea, but, more important, to quantify the difference in certain specific scenarios.

In the next chapter an extended description for the assembly process is given. This process description also serves as a introduction for the mathematical model description in Chapter 3. In that chapter all the different aspects from the process description are translated to a mathematical model that forms the foundation of the developed simulator. The developed simulator is validated in Chapter 4 by comparing the simulator's predicted throughput or order sojourn time with estimates based on mathematical analysis, a simulator from a previous project and empirical production data. This not only validates the implementation of the simulator, but also validates the mathematical model from Chapter 3. Then, in Chapter 5 a flow line can finally be compared with a job shop by both letting them assemble a certain fixed set of orders. By varying some of the parameters, e.g. order size or machine speed, the influence of these parameters on a performance indicator such as total line throughput will become clear. Finally, in Chapter 6 the most important results will be highlighted and in Chapter 7 recommendations for future research projects will be given.

CHAPTER 2

Process description

This master's thesis focusses on two production concepts: the flow line and the job shop. This chapter provides the relevant background information for both production concepts. The first sections of this chapter provide a broad picture of both production networks, the last sections give more details for a specific elements in the production network. In every section, first the general assumptions are given, possibly followed by the network specific assumptions. At the end of every section the most important assumptions are summarized. Note that all the actual parameter values, e.g. number of machines or product characteristics, in this chapter are only indicative; any parameter can be adjusted in the mathematical model.

2.1 Network layout

The global layout of a flow line is depicted in Figure 2.1, the layout of a typical job shop is depicted in Figure 2.2. There are not a lot of differences between the job shop and the flow line in the broader picture; both networks consist of several machines and in a flow line an order needs to visit all these machines, whereas certain machines may be skipped by some orders in a job shop. It is furthermore assumed that the flow line always has as many machines as the job shop. The exact number of machines is determined by the products produced on the production line, for complicated products more machines will be required. For both production concepts it is assumed that there always is external demand and the last machine in the production network can always store its output somewhere. Hence, the first (set of) machine(s) in a network is (are) never starved, and the last machine is (are) never blocked.

Summary

- Both a job shop and flow line network consist of the same number of machines.
- The first (set of) machine(s) in a network can never be starved
- The last machine in a network can never be blocked.

2.2 Products

By changing the machine configuration, NXP can produce many different products. These products are categorized in so-called packages such as SOT457, and SOT23. For sake of simplicity, only five

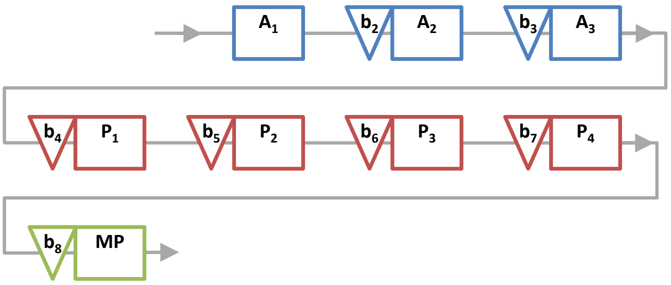


Figure 2.1: Machine layout in a flow line.

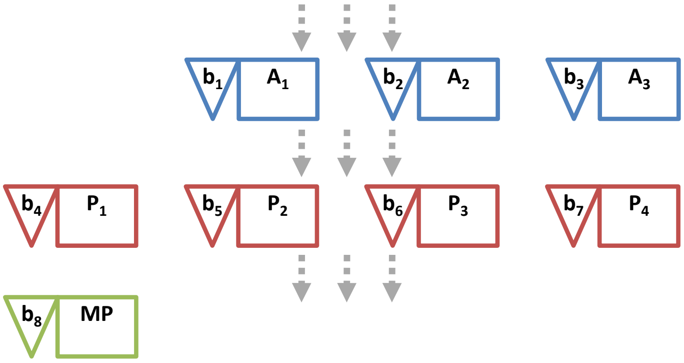


Figure 2.2: Machine layout in a job shop.

products from the SOT457 will be used in this project. The product's characteristics relevant for this project are:

- type of lead frame, in total there are 9 different types of leadframe
- number of dies (1, 2 or 3)
- number of 18µm wire bonds (0, 1, 2,...)
- number of 23µm wire bonds (0, 1, 2,...)
- number of 35µm wire bonds (0, 1, 2,...)
- number of 50µm wire bonds (0, 1, 2,...)
- the type of wire bonds (regular, single reverse or double reverse).

Table 2.1 lists these characteristics for the five products used in this project. Note that although product 2 and 3 have the same characteristics, the type of dies used in practice, and therefore also the products itself, are different. All these five products are schematically depicted in Figure 2.3.

Name	ID	lead frame	dies	18µm	23µm	35µm	50µm	wire type
BC807DS(/C)	1	1	2	0	4	0	0	double reverse
PBSS4140DPN(/B)	2	1	2	0	4	0	0	normal
PIRND2	3	1	2	0	4	0	0	normal
BAS21AVD(/C)	4	2	3	0	3	0	0	normal
BZA408B(/C)	5	3	1	0	4	0	0	normal

Table 2.1: Product information for simulator.

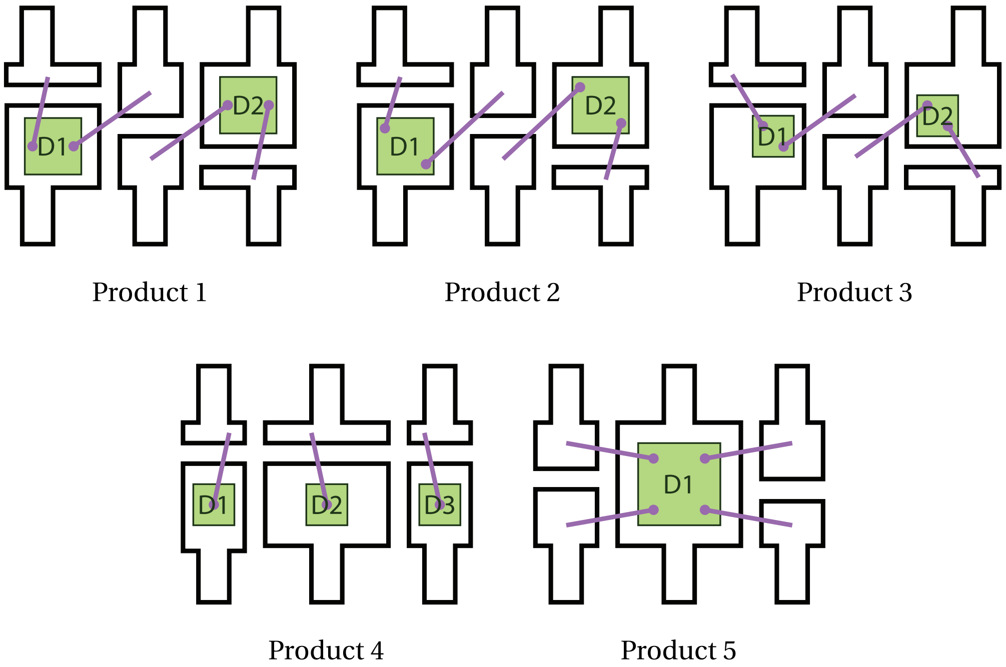

Product 1
Product 2
Product 3
Product 4
Product 5

Figure 2.3: Schematic overview of the layout of the five different product types.

2.2.1 Orders

Customers can place orders at NXP's Sales department, this raw order data is then processed by the planning department, resulting in a production planning for a certain period of time. Hence, at the start of any period it is known what orders, characterized by product type and order size, should be produced and in what sequence. Optimizing the order sequence will improve the line performance. However, doing so not only deviates from the main goal of this master's thesis, it also falls outside the scope of the ITEC department. However, if NXP Semiconductors decides to change from flow line to job shop assembly, further research on optimizing the order sequence is highly recommended. It is assumed that the orders enter the network in the same order as they are listed in the production planning. Usually a changeover is required for a machine between processing different products. The planning department tries to cluster orders in such a way that the eventual planning results in as little changeovers as possible. More information about this changeover process is given in Section 2.3.4.

Since at the beginning of a period it is known what products and in what order have to be produced, one needs to think about the order release process. In a flow line all orders could, in practice, be released at once. However, in a job shop doing so could result in a considerable growth of the work-in-progress levels. Hence, orders is only released at an ADAT, once that ADAT is actually able to start processing that order. In a flow line this means that a new order is released the moment the first ADAT has finished processing the previous order. In a job shop a new order is be released as soon as there is a free ADAT, i.e., when the buffer in front of the ADAT is empty, there are no cassettes being transported to that ADAT and the ADAT is not reserved for another assembly process. The reservation of ADATs for assembly processes is a strategy used to limit the number of changeovers and will further be explained in the Section 2.2.3.

2.2.2 Production stages

Especially in a job shop one needs to keep track of the progress of a cassette. Therefore ten production stages, see Table 2.2, have been defined. Although nine stages would have been sufficient for the SOT457 package, one additional die-stage has been added for products from the SOT23 package. Note that a product does not necessarily need to pass through all production stages. For example, if only one die needs to be placed for a certain product, then a cassette containing that product will skip stage 2, 3 and 4. It is also possible that a production stage does not change after a service completion. For example, if multiple PHICOMs are used to place several 18 μm wire bonds, then a product will several times be in stage 5. A more production oriented way of looking at production stages 0-4 is to see them as the position of a SOT23 product in a flow line with four ADATs, four PHICOMs and one Multi Plunger: if a product is in stage 3 then the third die on a column of leadframe will be placed and that column will be either in the buffer in front of the third ADAT or at the third ADAT itself.

stage	action	end of stage production progress
0	enter the network	at most 4 more dies, all wires and the mold still need to be placed on every product
1	place the first die	at most 3 more dies, all wires and the mold still need to be placed on every product
2	place the second die	at most 2 more die, all wires and the mold still need to be placed on every product
3	place the third die	at most 1 more die, all wires and the mold still need to be placed on every product
4	place the fourth die	all dies have been placed, all wires and the mold still need to be placed on every product
5	bond the 18 μm wires	all dies and 18 μm wires have been placed, all 23, 35, 50 μm wires and the mold still need to be placed on every product
6	bond the 23 μm wires	all dies, 18 and 23 μm wires have been placed, all 35, 50 μm wires and the mold still need to be placed on every product
7	bond the 35 μm wires	all dies, 18, 23 and 35 μm wires have been placed, all 50 μm wires and the mold still need to be placed on every product
8	bond the 50 μm wires	all dies and wires have been placed, the mold still need to be placed on every product
9	mold	the product (diode) is ready

Table 2.2: The different production stages.

2.2.3 Sub-batches

One of ITEC's main objectives is to improve the line throughput. The most obvious way to achieve this goal is to limit the number of machine changeovers in a production network. Not only will a machine not be able to assemble products while in changeover, changeovers might also result in starvation of a machine located downstream in the production network. In a flow line the changeovers are limited due to the clustering of orders in the production planning and since every product will visit the machine in exactly the same order, no intermediate changeovers will be required once the machines have correctly been configured. However, in a job shop the routing is very flexible and in the worst case a changeover might be required every time a new cassette is being processed at a machine. To prevent this from happening a principle called *wormhole routing* is used for the job shop setting. This principle simply means that the first cassette of an order determines the routing of all

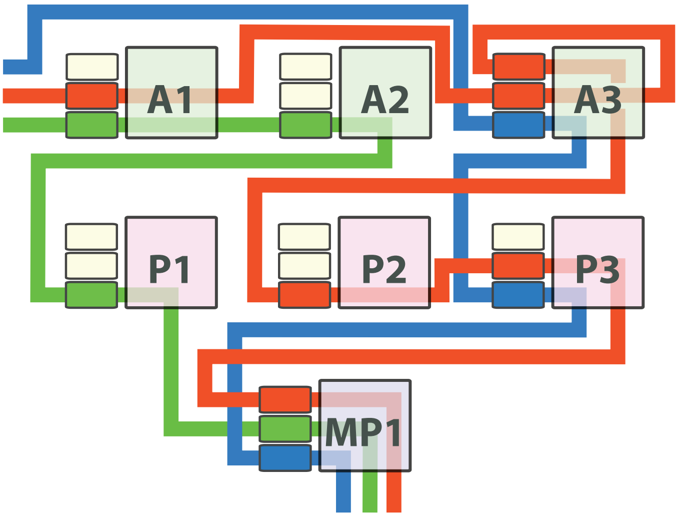


Figure 2.4: Schematic display of wormhole routing: every colored thick line depicts a worm, the undermost worm (in the buffer) will always be served first at a machine, the other worm has to wait until the tail from the undermost worm has left the machine.

other cassettes from that order and once a machine starts processing the first cassette, it is only allowed to process cassettes from that order and cassettes from a different order will have to wait until the last cassette from that order has been processed. The result is a production network in which worms of cassettes wriggle through a production network. The head of each worm determines the route of the entire worm and whenever the paths of two or more worms meet, one worm has to wait until the other worm has fully passed. This principle is depicted in Figure 2.4 and Table 2.3. Note that the depicted routing is arbitrary and not based on actual production data. Also note that the head of the red worm has to wait twice at machine A3: the first time it has to wait until the blue tail leaves machine A3 and the second time it has to wait until its own red tail has been processed once at machine A3. The wormhole routing principle does have a few disadvantages. The first one is related to the order size. Once an order is very big it might take over the entire network and since all machines will be processing that order, it might block the progress of other orders. This problem can be solved by dividing the orders into smaller parts called sub-batches. Every sub-batch can then be regarded as a separate order with its own routing and the previously described problem cannot occur. The downside is that one runs the risk of having a new changeover for every sub-batch. Hence the question becomes: How big should a sub-batch be? For this master's thesis two intuitive batch-split algorithms have been implemented, both are be introduced in Section 2.4.1. One other disadvantage

machine	1st reservation	2nd reservation	3rd reservation
A1	green, die 1	red, die 1	
A2	green, die 2		
A3	blue, die 1	red, die 2	red, die 3
P1	green, wire set 1		
P2	red wire set 1		
P3	blue, wire set 1	red, wire set 2	
MP1	blue, mold 1	green, mold 1	red, mold 1

Table 2.3: The sequential machine reservations for the wormhole routing depicted in Figure 2.4.

of the wormhole routing principle relates to undesirable starvation of machines. If, for example, the last cassette of a sub-batch is stuck somewhere, other machines in the production network have to wait for that specific cassette and starvation might occur. This is one of the trade-offs for limiting the number of changeovers, but by proper scheduling, e.g. performing jobs with high stuck-risks as late as possible, and possibly manual intervention of an operator the risk of starvation can be reduced.

2.2.4 Cassettes

In a job shop cassettes are used to transport the leadframe from one machine to another. A cassette can store a certain number of strips, denoted by h (usually $h = 40$). A strip is actually just a small piece leadframe containing $l \times w$ product positions (usually 14×40). Hence, one cassette can store a total of $h \times l \times w$ products. Right before an order of size n is released in a network, $\lceil \frac{n}{hlw} \rceil$ cassettes are filled with strips of leadframe. Because n will probably not be a multiple of $h \times l \times w$, one can choose to fully fill all cassettes except the last one, or one can evenly balance the number of strips in a cassette. The latter is assumed to be the default.

Summary

- A product is characterized by: numbers of dies, number of wires, wire- and leadframe type.
- At the beginning of a period it is know what products and in what order are going to be produced.
- Orders are releases as soon as the first ADAT (or any ADAT in a job shop) is free.
- Stages are used to keep track of the progress for a cassette in a job shop.
- Wormhole routing is used in a job shop to limit the number of changeovers.
- In a job shop big orders are split into smaller sub-batches.
- In a job shop cassettes filled with strips of leadframe are used to transport and store the products. Each order, and therefore also every sub-batch, consists of a certain number of cassettes.

2.3 Machines

In any network there are three type of machines: die bonders (ADATs), wire-bonders (PHICOMs) and a molding machine (Multi Plunger); in front of every machine there is a buffer to withstand small fluctuations in the production process that is running 24 hours a day, 7 days a week. The exact number of ADATs and PHICOMs is determined by NXP and will be treated as given input parameter. However, there will always be exactly one Multi Plunger in a production network. The speed at which a machine assembles items (Section 2.3.1) varies per machine type and setup. However, due to machine downtime (Section 2.3.2) and changeovers (Section 2.3.5) the actual throughput of a machine is significantly lower. Machine throughput is suppressed even more due to starvation which occurs when the buffer in front of a machine is empty, or due to blocking which occurs when a machine can not store its assembled products in a subsequent buffer because that buffer is already full.

2.3.1 Machine speed

The machine speed refers to the theoretic maximum number of dies, wires or molds that a machine can place per time unit and it is assumed that the speed of a machine in a job shop is the same as in a flow line. For an ADAT and a Multi Plunger the machine speed does not depend on the machine setup. However, for a PHICOM the machine speed does vary. For example, due to calibration processes it takes more time to wirebond four wires on two dies than it does to make four regular wire bonds for only one die. The default PHICOM speed refers to a the maximum number of wires a PHICOM can

setup	normal	single reverse	double reverse
1D1W	-9,8%	-16,9%	-22,9%
1D2W	0,0%	-100,0%	-15,8%
1D4W	5,0%	-4,2%	-12,5%
2D2W	-14,6%	-20,8%	-26,3%
3D3W	-16,3%	-22,5%	-27,5%

Table 2.4: PHICOM speed penalties for different configurations.

bond in a normal one die, two wire setting. Table 2.4 lists the relative speed penalties for different PHICOM configurations compared with the default speed. Furthermore it is assumed that a machine works at full speed, or does not work at all.

2.3.2 Machine downtime

Sometimes a machine is unexpectedly not able to assemble products, either because of a machine error or because an operator has manually stopped a machine (halted state). Whenever this situation occurs, the operator has to solve the problem and restart the machine. Once a machine is repaired, it will continue assembling items where it stopped. Although in practice an error might result in an incomplete final product, this effect will be ignored for this project. Besides the unplanned machine downtime, there also is scheduled downtime. The entire production line will, in general, be shut down during such a scheduled downtime and since the network layout usually does not influence the duration of the downtime, this planned downtime will have the same influence in a flow line and job shop. Therefore, the planned downtime will be ignored in this project.

The only exception is the planned periodic maintenance occurring at the Multi Plunger. In order to place a proper encapsulation, the big mold inside the Multi Plunger needs to be cleaned once every 8 hours. The cleaning process takes, on average, 30 minutes. In the model used for this project, fixed values are used for the inter cleaning times and the duration of the maintenance. Furthermore it is assumed that the inter cleaning times are time-, and not production, dependent.

2.3.3 Buffer

As has been mentioned before, there is a buffer in front of every machine. Strictly speaking, there is no actual buffer in front of the first ADAT in a flow line. However, since the production process actually is an assembly process, the big reel of leadframe could be regarded as a big finite buffer as well. In a job shop there is an output buffer at every machine as well. However, in this project it is assumed that there always are employees available to immediately start the transportation process once a cassette has been processed at a machine. Hence, this output buffer can never be full and can therefore be ignored. In a flow line the buffers are created by diverting the leadframe over a set of pulleys, therefore the buffer capacity is limited. However, in a job shop one can easily stack the cassettes in front of a machine. Hence, the buffers are assumed to be infinite. It is assumed that one can add any number of products to a buffer that is currently not full, even if this temporarily results in a buffer overflow. When a buffer is full, i.e., the current buffer level exceeds the buffer capacity, no more products can be added to that buffer. If another machine does try to add products to the saturated buffer, that machine will be blocked until the buffer is not full any more. In a job shop a blocked machine will completely stop processing cassettes. In a flow line a blocked machine will slow down to the speed of the next machine, thereby preventing a buffer overflow.

2.3.4 Workload balancing

Especially in a flow line one has to use the limited capacity as much as possible. However, it makes no sense to have a die-bond section assembling 48k products per hour, if the wire-bond section can only assemble 30k products per hour. NXP has some guidelines for balancing the workload on the machines in a flow line. Although it is unlikely that these guidelines result in the optimal configuration, it is beyond the scope of this master's thesis to improve these ways of working. However, some obvious improvements will be stated, it is further left to ITEC and NXP to further explore these possibilities. In this section first the balancing in a flow line and then in job shop will be discussed.

2.3.4.1 Flow line

As has been mentioned before, in a flow line a double tracked leadframe is used to transport the products from one machine to another. ITEC has developed and improved the machines for a flow line in such a way that a machine is able to process zero, one, or both of the tracks; processing two tracks takes twice the time of processing one track. The easiest and most intuitive way to describe the relation between tracks and machines is to divide an order size by the number of tracks, multiply the processing time at a machine with the same number and to keep in mind that a machine can now place 0, 1, or 2 dies/molds or any number of wires. If 0 dies/wires/molds are placed, a machine is said to be indexing and the machine will only pass through the leadframe. If a machine is indexing, it tries to keep the next buffer as full as possible. The exact regulations for the flow line and machine configuration are not known. However, the line configuration for the products used in this master's thesis are known. Since the model needs to be widely applicable, this information has been generalized. Because the machine configuration differs per machine type, the workload balancing will be explained for the three machine types separately. Note that this specific balancing only holds for a flow line configuration, the machine configuration for a job shop is discussed in Section 2.3.4.2.

ADAT

If there are m dies that need to be placed per product, then in total $2m$ dies will have to be bonded if a double-tracked leadframe is used. Now if there are n ADATs in the network, then the right most ADATs will, in practice, all place $\lceil \frac{2m}{n} \rceil$ dies each and the remaining ADATs at the beginning of the line will be indexing. Since it is assumed that the indexing itself takes no time at all, an indexing machine will be practically invisible from a product point of view and the buffer after the indexing machine will, in essence, be twice as big. The main disadvantage of the current strategy is that the buffer capacity of the indexing machines is unused since the first few ADATs can actually be ignored. One obvious improvement would be to configure the first ADATs to place $\lceil \frac{2m}{n} \rceil$ dies each and set the right most ADATs to indexing. Doing so will, in practice, increase the buffer level between the die- and wirebond section since the buffer in front of the first PHICOM is extended with the buffer in front of an indexing ADAT. However, this topic will not be further investigated in this master's thesis.

PHICOM

The workload at the n PHICOMs in the network is spread in a different way. If one assumes that m wires of the same diameter have to be placed per product, then in total $2m$ wires have to be placed per product column for a double-tracked leadframe. The odd (1st, 3rd, ...) PHICOMs will then place $\lfloor \frac{2m}{n} \rfloor$ wires and the even (2nd, 4th, ...) PHICOMs will place $\lceil \frac{2m}{n} \rceil$ wires. All products mentioned in this master's thesis require only one type of wire, for multi-type wire products a different strategy is required since a PHICOM can only bond one type of wire. For these products the PHICOMs will first be evenly spread for the different wire types, and then the same strategy as above will be applied.

Multi Plunger

Since there will only be one Multi Plunger in the network, the workload does not have to be balanced for this type of machine.

Example

The flow line configuration for the products in Table 2.1 and a network with a double-tracked leadframe, 3 ADATs, 4 PHICOMs and 1 Multi Plunger are listed in Table 2.5.

Product	A_1	A_2	A_3	P_1	P_2	P_3	P_4	MP_1
1	0 (0T)	2 (2T)	2 (2T	1D2W (1T)	1D2W (1T)	1D2W (1T)	1D2W (1T)	2 (2T)
2	0 (0T)	2 (2T)	2 (2T)	1D2W (1T)	1D2W (1T)	1D2W (1T)	1D2W (1T)	2 (2T)
3	0 (0T)	2 (2T)	2 (2T)	1D2W (1T)	1D2W (1T)	1D2W (1T)	1D2W (1T)	2 (2T)
4	2 (2T)	2 (2T)	2 (2T)	1D1W (1T)	2D2W (1T)	1D1W (1T)	2D2W (1T)	2 (2T)
5	0 (0T)	1 (1T)	1 (1T)	1D2W (1T)	1D2W (1T)	1D2W (1T)	1D2W (1T)	2 (2T)

Table 2.5: The total number of objects (dies, wires or molds) placed per product column and number of tracks (listed between brackets) processed by a machine in a default flow line; for the PHICOMs the number of dies involved in the wire bonding process are also listed (see Section 2.3.1).

2.3.4.2 Job shop

The machines in a job shop handle the input in a fundamentally different way. Whereas one could choose to process zero, one or two tracks in a flow line, in a job shop all the products in a cassette will receive exactly the same treatment. Hence a machine in a job shop will, strictly speaking, always process all 14 tracks. Due to the flexibility of a job shop, there are many different ways to balance the workload. In this project the cassettes from one order are split into several sub-batches, each with their own routing through the network. Any ADAT or Multi Plunger will place exactly one die or mold per product position and every PHICOM will bond all the wires of a certain diameter for every product in a cassette. As a result, one specific cassette will always visit exactly as many ADATs as there are dies on a product, exactly as many PHICOMs as the number of different diameter goldwires required for that product and always exactly one Multi Plunger. Due to the splitting in sub-batches, several machines can, simultaneously, process one order and the workload will therefore be balanced. The principle of splitting orders into sub-batches has already been discussed in Section 2.2.3. The different algorithms for splitting orders in sub-batches will be given in Section 2.4.1. Since the buffers in a job shop are infinite, one runs the risk of facing high work-in-progress (WIP) levels. Therefore it is possible to limit the total number of cassettes in a job shop. If there are more cassettes in a job shop, then no new sub-batches will be released in the network.

2.3.5 Machine setups

The setup of an ADAT or PHICOM is product specific. As soon as the number, type or position of the dies or wires that are placed by machine changes, a changeover will be required. A Multi Plunger always needs the same setup. Many factors influence the duration of this changeover. However, for this project they are classified in four categories per machine type. These four categories are listed below. For NXP it is worth mentioning that the term changeover process is used to describe many different actions possibly including the welding of one reel of leadframe to another reel, realigning and other processes.

Same product, same setup The same die/wire will be placed at the same position for the same product.

Different setup, same leadframe There is only a difference in the die/wire type or position, but the leadframe used for the current and the new product will be the same.

Different leadframe The leadframe for the current and new product are different; note that the die/wire type or position still may be the same.

Indexing The machine is set to indexing mode and leadframe will pass through the machines without being processed (only used in flow line).

In a job shop the changeover process is rather straightforward, as soon as a changeover for a machine is required the assembly process at that machine will be halted and continues once the changeover has been performed. Since a wormhole routing principle is used, only the start of processing a first cassette of a certain sub-batch can trigger a changeover at a machine and a new changeover and a new changeover is only allowed when the last cassette from that sub-batch has been processed. The changeover process at one machine has, due to the infinite buffers, little influence on other machines in the network. The changeover process in a flow line is similar to the one in a job shop: only the first piece of leadframe of a new order can trigger the start of a changeover and a machine's setup will not change until the last piece of leadframe from that order has fully been processed; a flow diagram for the changeover process of a simplified flow line is given in Figure 3.5. From this diagram one can conclude that no two machines in a flow line can simultaneously be in setup. Furthermore, the limited buffer capacities might cause starvation of machines down stream in the production line. In fact, a quick calculation yields that a buffer capacity of 4000 products and an average throughput of 20000 Uph (units per hour) can withstand a period of at most 12 minutes without input. Hence, if the changeover of one machine takes longer than 12 minutes, then an almost inevitable result will be the starvation of a machine further in the line.

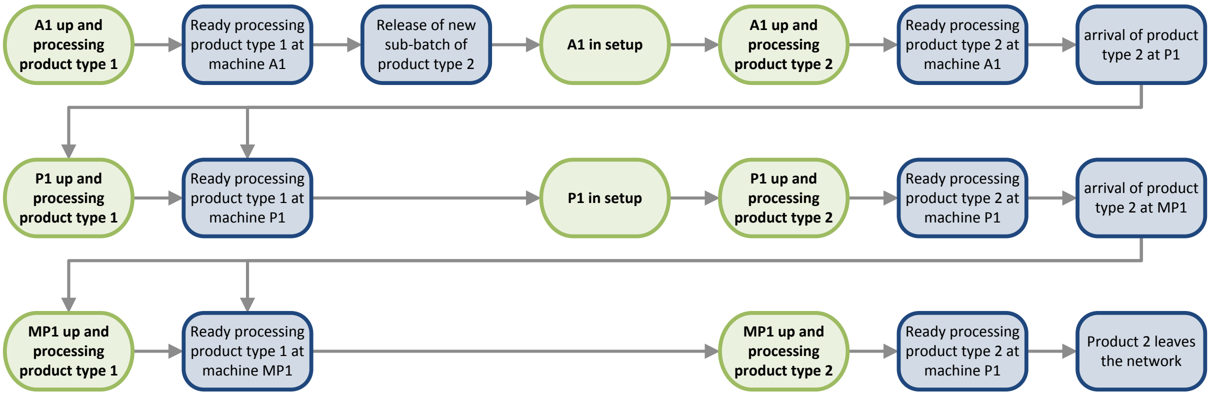

A1 up and processing product type 1
Ready processing product type 1 at machine A1
Release of new sub-batch of product type 2
A1 in setup
A1 up and processing product type 2
Ready processing product type 2 at machine A1
arrival of product type 2 at P1
P1 up and processing product type 1
Ready processing product type 1 at machine P1
P1 in setup
P1 up and processing product type 2
Ready processing product type 2 at machine P1
arrival of product type 2 at MP1
MP1 up and processing product type 1
Ready processing product type 1 at machine MP1
MP1 up and processing product type 2
Ready processing product type 2 at machine P1
Product 2 leaves the network

Figure 2.5: Process flow for changing the setup of an entire flow line with 1 ADAT, 1 PHICOM and 1 Multi Plunger.

2.3.6 Service policy

In a flow line products and orders can, due to the reel of leadframe connecting the different products and machines, only be processed in order of arrival, in a job shop there is much more freedom. In a job shop it is assumed that sub-batches are processed in order of arrival of their first cassette. Note that due to the wormhole routing principle, the cassettes located in a machine's buffer are not necessarily processed FCFS (first come, first served).

Summary

- A production network consists of three type of machines: die-bonders (ADAT), wire-bonders (PHICOMs), and a molder (Multi Plunger).
- There is a buffer in front of every machine.
- In a flow line the WIP-levels are controlled by the finite buffers. In a job shop the release of new sub-batches can be blocked if there are too many cassettes in the network.
- A machine is either up, down (error and halted), in setup, starved or blocked.
- Every machine assembles products at a certain rate. This rate is influenced by factors such as machine type, and machine setup. This rate is, in a job shop, used to determine the processing time of a certain cassette.
- The workload in a flow line can be spread by letting a machine process 0, 1 or 2 tracks. In a job shop this can be achieved by splitting an order into several sub-batches.
- For every production stage a different machine setup is required for the ADATs and the PHICOMs, the setup of a Multi Plunger does not have to be changed. The time required for a changeover differs per machine type and per changeover category.
- Products in a flow line and sub-batches in a job shop are processed FCFS.

2.4 Algorithms

There is a lot of flexibility in a job shop. However, this flexibility should be used wisely. This section focuses on the algorithms used to determine the size of a sub-batch (Section 2.4.1) and the routing of a sub-batch (Section 2.4.2). Since the main goal of this project is to get a global insight in the characteristics of a job shop, only straightforward algorithms have been implemented. Implementing more complex algorithms would only deviate too much from this project's main goal: getting a global insight in the characteristics of a job shop. Once it is certain that NXP is going to switch from flow line to job shop assembly, more complex algorithms can easily be added to the developed simulator. The same holds for many other assumptions made in this chapter (e.g. the FCFS sub-batch service policy), once the results for a straightforward job shop assembly network have been analyzed and seem promising, more complex strategies can easily be added and analyzed.

2.4.1 Splitting orders in sub-batches

Two algorithms for determining the number of sub-batches per order, for a job shop, have been developed. These two algorithms are called fixedSplit and smartSplit and are described below.

2.4.1.1 FixedSplit

The result of the *fixedSplit* algorithm is that every sub-batch consists of at most q cassettes, where q is an input parameter for the algorithm. This idea is formalized in Algorithm 1. For determining the number of cassettes in a sub-batch, the average number of cassettes in a sub-batch is either rounded up or down. The main advantage of the fixedSplit method is that every sub-batch will consist of at most q (input parameter) cassettes and if the orders are big enough, every sub-batch will consist of approximately q sub-batches as well, hence the name fixedSplit.

2.4.1.2 SmartSplit

One disadvantage of the fixedSplit algorithm is that it is not influenced by the average duration of a changeover. On one hand, if a changeover would take extremely long then one does not want to split an order into sub-batches since more sub-batches might imply more changeovers. On the other

Algorithm 1 fixedSplit

```
Input:
    • maximum sub-batch cassette size $q \in \mathbb{N}$
    • order size $c$ (measured in cassettes)
Output: the number of sub-batches $m$ to split an order in.
    1. Output $\left\lceil \frac{c}{q} \right\rceil$
```

hand, if a changeover would take no time at all, then it is advised to split an order in as many sub-batches as possible since, due to the machine reservation in combination with the wormhole routing, large sub-batches increase the risk of starvation. Therefore another algorithm has been implemented as well, because this algorithm does use some order information, this algorithm is called *smartSplit*.

The underlying principle of the smartSplit algorithm is a weighted tradeoff between the decrease of order sojourn time versus the increase of changeover time. Although this underlying principle is fairly intuitive, the formal description in Algorithm 2 is not that clear cut. Hence, the smartSplit algorithm will be explained using an example. Say there is an order consisting of 6 cassettes and each cassette needs to visit 1 out of 3 ADATs, 1 out of 4 PHICOMs and 1 out of 1 Multi Plungers. Now assume that processing 1 full cassette at an ADAT, PHICOM or Multi Plunger takes 45, 30, respectively 20 minutes and a changeover takes, on average, 25 minutes for an ADAT and 15 minutes for a PHICOM. Since there are no changeovers required for a Multi Plunger, and since there is only one Multi Plunger, this machine will be ignored by the smartSplit algorithm. If the order is split into only one sub-batch, then one changeover at the ADAT and one changeover for the PHICOM will be required, taking an average of $25+15=40$ minutes; processing the 6 cassettes on an ADAT respectively PHICOM takes $6 \cdot 45 = 270$ respectively $6 \cdot 30 = 180$ minutes. Hence, it takes on average $40+270+180 = 490$ minutes to process this order (including the changeover time). If the order is split into 2 sub-batches, then twice as many changeovers may be required ($40 \cdot 2 = 80$ minutes in total). However, the two sub-batches can then be processed simultaneously (assuming that enough machines are available). Hence, the order processing time will be $\frac{270+180}{2} = 225$ minutes, yielding a total of $80+225 = 305$ minutes. If the order is split into 3 sub-batches, then the total changeover times would be 120 minutes and the overall processing time would only be $\frac{270+180}{3} = 150$ minutes, yielding a total of $120+150 = 270$ minutes. Since there only are 3 ADATs, at most three sub-batches can simultaneously be assembled. Hence, for the smartSplit algorithm it makes no sense to split an order in more than 3 sub-batches.

The conclusion from the previous example is clear cut: one should split the order in 3 sub-batches each containing 2 full cassettes if one wants to keep the order sojourn time as small as possible. However, since more changeovers are required, the decrease in order sojourn time will be at the expense of machine utilization. In the smartSplit algorithm one can give extra weight to every changeover with input parameter $q \geq 1$. For example, $q = 1$ results in the comparison above, whereas $q = 2$ practically multiplies the duration of a changeover with a factor 2. Hence, the weighted total processing time (including changeover time) would be 530, 385 and 390 minutes for splitting in 1, 2 respectively 3 sub-batches. In that case the outcome for the algorithm would be to split the order in 2 sub-batches. In general a q close to 1 results in a fair comparison, whereas a large q will split the order very few sub-batches.

The main advantage of the smartSplit algorithm is that it actually uses the average changeover duration. The biggest disadvantage is that the algorithm ignores waiting times at machine and only compares the gain in processing time towards the loss due to an additional changeover. Therefore, an order will never be split into more sub-batches then there are ADATs or PHICOMs in a network since one simply cannot simultaneously process al these sub-batches. Hence, if a huge order is split in, say 3, sub-batches, then these 3 sub-batches will still be big.

Algorithm 2 smartSplit

Input:

- changeover penalty parameters $q \geq 1$
- order size p (measured in products)
- total number of dies (d), 18µm wires (w_{18}), 23µm wires (w_{23}), 35µm wires (w_{35}), 50µm wires (w_{50}) required per product
- average ADAT and PHICOM speed v_{A} and v_{P} (measured in die/wire placements per hour)
- average ADAT and PHICOM changeover C_{A} and C_{P} (measured in hours)
- number of ADATs (N_{A}) and PHICOMs (N_{P}) in network.

Output: the number of sub-batches m to split an order in.

1. **For** $i = 1, 2, \dots \min(N_{\text{A}}, N_{\text{P}})$, **calculate**

$$\begin{aligned}\text{WPT}(i) = {} & \frac{p \cdot d}{i \cdot v_{\text{A}}} + \frac{p \cdot (w_{18} + w_{23} + w_{35} + w_{50})}{i \cdot v_{\text{P}}} + \\ & q \cdot i \Big(d \cdot C_{\text{A}} + \left[\mathbb{1}_{\geq 0}(w_{18}) + \mathbb{1}_{\geq 0}(w_{23}) + \mathbb{1}_{\geq 0}(w_{35}) + \mathbb{1}_{\geq 0}(w_{50}) \right] \cdot C_{\text{P}} \Big),\end{aligned}$$

2. **Output** $\underset{i}{\operatorname{argmin}} \, \text{WPT}(i)$

2.4.2 Routing

Whereas the routing in a flow line is fixed, the routing in a job shop is very flexible. The only constraints are the following:

1. First the leftmost die is placed, then the one in the middle and the rightmost die will be bonded last.
2. First the 18µm wires are placed, followed by the 23, 35, 50µm wires (in that order).
3. First all dies have to be placed, then all the wires and placing the mold always is the last procedure.
4. The first cassette of a sub-batch determines the routing for all cassettes in that sub-batch (wormhole routing).
5. Once a cassette is located in a machine's buffer, it will not be removed from that buffer until it has been processed.

Although the first two constraints are not necessary for NXP, they do provide more structure for the job shop assembly process. The effect of removing these two constraints will not be further investigated. The routing of the first cassette of a sub-batch will be determined dynamically, therefore three straightforward algorithms have been implemented: random (Section 2.4.2.1), shortest queue (Section 2.4.2.3), and smallest setup (Section 2.4.2.2). Before explaining the algorithms in more detail, two machine characteristics will first be defined: a machine is *feasible* for a cassette if the next assembly process for the products in a cassette can take place at that machine, e.g. for stage 0-4 (see Table 2.2 for stage information) only an ADAT is feasible. A machine is *free* if there are no cassettes in its buffer, there are no cassettes in transportation to that machine and the machine is not reserved for a sub-batch. All three algorithms give absolute priority to feasible free machines, i.e., whenever there is a feasible free machine, the first cassette of a sub-batch will go to that machine no matter what algorithm is selected. If there are multiple feasible free machines then the selected algorithm will be applied to the set of feasible free machines. If there are no feasible non-full machines at all, then the algorithm will return the empty set, in that case the machine where the cassette was allocated before will get blocked, note that this situation can only occur when the buffers are finite.

2.4.2.1 Random

The random algorithm picks, as the name suggests, a random feasible free machine. If there are no feasible free machines, then the algorithm picks a random feasible machine. Although NXP may never use this algorithm in practice, the random routing algorithm can be used in the validation process of the developed simulated. Furthermore, the performance of the other two algorithms can now be compared with the random routing algorithm. If applying another algorithm yields the same, or possibly even worse results than the random algorithm, then there probably is a lot of room for improvement. The random algorithm is formalized in Algorithm 3.

Algorithm 3 random

```
Input:
    • set of feasible machines $M_0$
Output: a machine $m$ to transport the first cassette of a sub-batch to.
    1. Determine $M_e \subseteq M_0$, the set of all feasible free machines.
    2. If $M_e \neq \emptyset$, then $M = M_e$, else $M = M_0 \setminus M_f$, where $M_f \subseteq M_0$ is the set of all machines with full
       buffers.
    3. Output a randomly selected element $m \in M$.
```

2.4.2.2 Smallest setup

Since the routing of a sub-batch is determined dynamically, every call to a routing algorithm must have been preceded by either the release of the first cassette c_1 of a sub-batch in the network, or by a service completion of a first cassette c_1 of a sub-batch. One of the consequences of the wormhole routing is that only a first cassette of a sub-batch can trigger a changeover of a machine and since sub-batches are processed in order of arrival at a machine, the last setup of a machine will be the machine setup required for the last sub-batch located in a machines buffer, i.e., the sub-batch belonging to the hindmost first cassette of a sub-batch. Hence, the type of changeover, and therefore also the duration, required for changing a machine to process cassette c_1 can be determined on forehand. The smallest setup algorithm uses this idea to route to the machine at which the eventual changeover duration will be as small as possible. If there are more feasible machines sharing the smallest changeover duration, then the shortest queue algorithm is used on this set of machines. If there still are more machines satisfying these conditions, then the first machine from this set is picked. The smallest setup algorithm has been formalized in Algorithm 4.

Algorithm 4 smallest setup

Input:

- set of feasible machines M_0

Output: a machine m to transport the first cassette of a sub-batch to

1. **Determine** $M_e \subseteq M_0$, the set of all feasible free machine.
2. **If** $M_e \neq \emptyset$, **then** $M = M_e$, **else** $M = M_0 \setminus M_f$, where $M_f \subseteq M_0$ is the set of all machines with full buffers.
3. **Determine** $M_l \subseteq M$, the set of machines for which the eventual changeover duration to processing the new cassette is as small as possible.
4. **If** $\|M_l\| = 1$, **then** $m = M_l$, **else** find the first machine $m \in M_l$ such that the buffer content of machine m is as small as possible.
5. **Output** m.

2.4.2.3 Shortest queue

The shortest queue algorithm picks, as the name suggests, a feasible, preferably free, machine with the least number of products in its buffer. If there are more feasible machines with equal number of products in the buffer, then the smallest setup algorithm is used on this set of machines. If there still are more machines satisfying these conditions, then the first machine from this set is picked. The shortest queue algorithm could, in the future, easily be enhanced by not only looking at the number of products in a buffer, but also include the expected number of changeovers or total changeover duration for the products currently in the buffer. The regular shortest queue algorithm is formalized in Algorithm 5.

Algorithm 5 shortest queue

Input:

- set of feasible machines M_0

Output: a machine m to transport the first cassette of a sub-batch to.

1. **Determine** $M_e \subseteq M_0$, the set of all feasible free machine.
2. **If** $M_e \neq \emptyset$, **then** $M = M_e$, **else** $M = M_0 \setminus M_f$, where $M_f \subseteq M_0$ is the set of all machines with full buffers.
3. **Determine** $M_l \subseteq M$, the set of machines with the least number of products in their buffer.
4. **If** $\|M_l\| = 1$, **then** $m = M_l$, **else** find the first machine $m \in M_l$ such that the eventual changeover time for the new cassette at machine m is as small as possible (use principle of Algorithm 4).
5. **Output** m.

Summary

- Two algorithms for splitting an order in a job shop into sub-batches have been implemented:
    - FixedSplit, make sure that every sub-batch consists of at most q cassettes.
    - SmartSplit, balance the decrease of (overall) processing time against the increase of total changeover duration.
- Three sub-batch routing algorithms have been implemented for a job shop network:
    - Random, pick a randomly feasible machine.
    - Shortest queue, go the a feasible machine with the smallest queue.
    - Smallest setup, go to a feasible machine where the expected changeover time is as small as possible.
- All three algorithms give absolute priority to feasible free machines.

CHAPTER 3

Model description

The process information mentioned in the previous chapter has been translated to a mathematical model. Since the detailed information has already been given, Sections 3.2 and 3.3 very briefly describe the most important elements from the mathematical model. However, first a very important model assumption will be motivated in Section 3.1.

3.1 Discretized flow line

Two previous research projects performed by Mathematics for Industry program for NXP showed that a fluid model is perfectly able to describe a typical NXP flow line. However, such a continuous fluid model will not be able to accurately describe the way cassettes are processed in a job shop. Therefore, a discrete event simulator (with service completions) has been developed for the job shop. Now either a new simulator had to be developed for simulating a flow line, or the job shop simulator had to be changed in such a way that the simulator is able to accurately describe a flow line. In order to have maximum flexibility, the same simulator will be used to simulate a flow line and a job shop. However, it is not possible to simulate a continuous flow line using the simulator for the discrete job shop. Therefore the flow line is discretized. The most intuitive way to see this is to cut the endless leadframe in pieces of, say 1000, products and then feed these chunks of leadframe through the network. Key assumption here is that a machine can only process one chunk of leadframe at a time and one chunk can only be processed at one machine at a time. By decreasing the length of the chunk of leadframe, one decreases the number of products in a production bulk. As a consequence the discretization error may get smaller, however, this will also (linearly) increase the running time of the simulator. Hence, there is a trade-off between running time and accuracy, in Sections 4.1.8 and 4.2 the effect of discretizing a flow line will be investigated. Note that, apart from the value of some parameters (e.g. buffer levels), the only practical difference between a flow line and a discretized job shop is the routing of orders. The routing of a chunk of leadframe is fixed since every chunk visits every machine in exactly the same order whereas in a job shop the routing is variable.

3.2 Flow line model

A regular SOT457 flow line with N_{A} ADATs, N_{P} PHICOMs and N_{MP} Multi Plungers can be modeled as a production network with $N_{\mathrm{A}} + N_{\mathrm{P}} + N_{\mathrm{MP}}$ machines (depicted in Figure 3.1), all with an input buffer of certain size. A certain known set of orders, characterized by product type and size, all have to be processed by the machines in the production network. Since the flow line is discretized, an order is split into several small parts called sub-orders and every sub-order will have to visit every machine in a fixed order: sub-orders enter the network at the first machine, after being processed at a machine the sub-orders moves on to the next machine in the line; a sub-order leaves the network once it has been processed at the Multi Plunger. If a machine tries to transport a recently processed sub-order to a full buffer, that machine will be blocked until the target buffer is not full any more. Between processing the sub-orders from two different orders a changeover time might be required for a machine, during this changeover a machine will be down. The processing time of one sub-order at a machine is determined by several parameters such as product type, machine type, machine speed (e.g. the processing time is 0 when a machine is indexing). Unfortunately, the machines in the production network do break down, the machine reliability is discussed in Section 3.2.1. Furthermore, the last machine in the network, the Multi Plunger, also requires a periodic maintenance: once every 8 hours the Multi Plunger will be down for 30 minutes and after this downtime it will continue with its production.

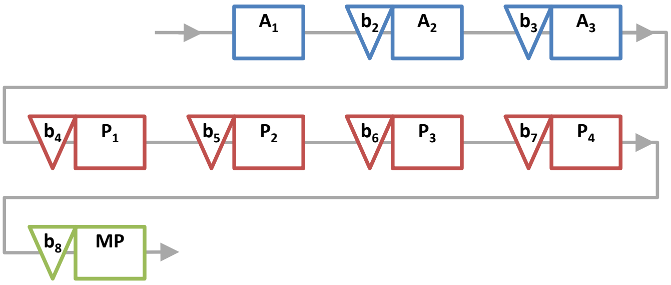


Figure 3.1: Machine layout in a flow line.

3.2.1 Machine downtime

In this project it is assumed that the mutual independent up- and downtimes are production-dependent, i.e., a machine can only break down when it is actually processing items. The underlying distribution for the up and down times is not known. However, NXP Semiconductors uses a system called AWACS to keep track of the machine performance and its up- and downtime behavior. The raw data in the AWACS database is, by ITEC, converted to a state file containing the time, duration and number of products produced for every state. A small segment of such a state file is listed in Table 3.1. From these state files one can extract the historical machine up- and downtimes, note that in this process certain uninteresting machine states are ignored ignored by ITEC. The data in Table 3.1 would result in an empirical up- and downtime sample of $30, 82, 17$ and $5, 10, 8$ seconds respectively. In the

State description	time (s)	duration (s)	products produced
wait_input (starved)	127	10	0
halted (down)	35	10	0
production (up)	0	30	28
	45	82	68
	137	17	13
error (down)	30	5	0
	154	8	0

Table 3.1: A small example of a state file, note that not all possible states are listed in this example.

developed simulator one can choose between sampling from the empirical data, either by selecting a random element or by maintaining the same order, or one can sample from a fitted phase type (mixed Erlang or hyper-exponential) distribution based on Tijms [4]. Whether a mixed Erlang or a hyper-exponential distribution is used depends on the sample's coefficient of variation. If the coefficient of variation $\mathrm{c}_{\mathrm{X}}^2 = \frac{\mathrm{Var}(X)}{\mathrm{E}(X)^2} \leq 1$ for a positive random variable X, one fits a mixed Erlang distribution $\mathrm{E}_{k,k-1}$ distribution with $k = 2, 3, \ldots$ such that

$$\frac{1}{k} \leq \mathrm{c}_{\mathrm{X}}^2 \leq \frac{1}{k-1}.$$

Consequently, the fitted distribution will with probability p and $1-p$ be the sum of $k-1$ respectively k independent exponential random variables of rate μ. By taking

$$\begin{aligned} p &= \frac{1}{1+\mathrm{c}_{\mathrm{X}}}\left(k\mathrm{c}_{\mathrm{X}}^2 - \sqrt{k(1+\mathrm{c}_{\mathrm{X}}^2) - k^2\mathrm{c}_{\mathrm{X}}^2}\right), \\ \mu &= \frac{k-p}{\mathrm{E}(X)}, \end{aligned}$$

the first two moments of the fitted distribution will correspond to the sample's first two moments.

If $\mathrm{c}_{\mathrm{X}}^2 > 1$, one fits a hyper-exponential H_2 distribution. Hence, the fitted distribution will with probability p and $1-p$ be an exponential distribution with rate parameter μ_1 respectively μ_2. If only the sample's first two moments are known, it is common practice to fit a hyper-exponential distribution with balanced means, i.e., $\frac{p}{\mu_1} = \frac{1-p}{\mu_2}$. By taking

$$\begin{aligned} p &= \frac{1}{2}\left(1 + \sqrt{\frac{\mathrm{c}_{\mathrm{X}}^2 - 1}{\mathrm{c}_{\mathrm{X}}^2 + 1}}\right), \\ \mu_1 &= \frac{2p}{\mathrm{E}(X)}, \\ \mu_2 &= \frac{2(1-p)}{\mathrm{E}(X)}, \end{aligned}$$

the first two moments of the H_2-distribution will correspond to the sample's first two moments. For most of the up- and downtime samples, a hyper-exponential distribution will be used, Figure 3.2 and 3.3 depict both the empirical and fitted cumulative distribution function for a sample of up and downtimes from a PHICOM machine. As can been seen from the figures, the fitted distribution actually is very close to the empirical distribution function. This justifies the usage of a fitted distribution.

The periodic maintenance at the Multi Plunger is also modeled as downtime. This downtime will occur every 8 hours and lasts for exactly 30 minutes. Right before the maintenance starts, the Multi Plunger stops placing molds and, as soon as the 30 minutes of maintenance are over, the machine resumes placing molds on products. The periodic maintenance can also interrupt the repair of a Multi Plunger. As soon as the Multi Plunger is cleaned and the periodic maintenance is over, the machine is repaired and will start placing molds again.

3.3 Job shop model

The mathematical job shop model is similar to the flow line model. The job shop network also consists of N_{A} ADATs, N_{P} PHICOMs and N_{MP} Multi Plungers (depicted in Figure 3.4), however, the buffers are now infinitely big. The input for the network once again is a set of orders, but the orders are no longer split into sub-orders. Instead, they are split into evenly big sub-batches and every separate

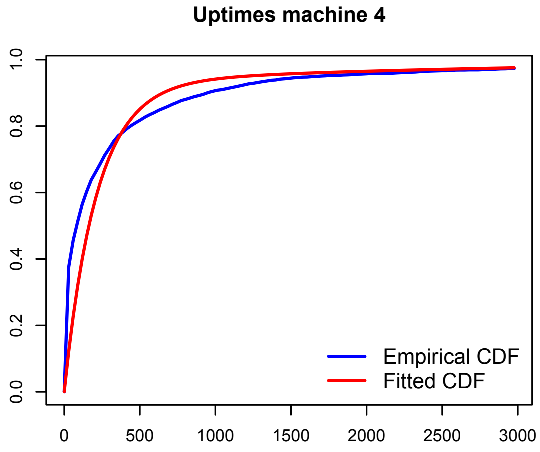


Figure 3.2: The empirical and the fitted phase type distribution function for a sample of up times.

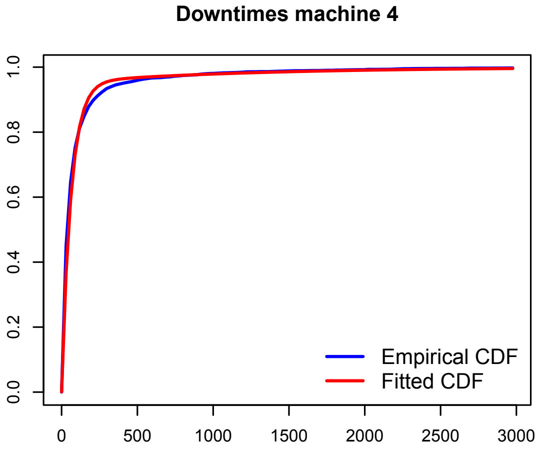


Figure 3.3: The empirical and the fitted phase type distribution function for a sample of down times.

sub-batch is then once more split into several cassettes of a certain fixed size. The number and order of machine visits for a cassette is determined by the cassette's product type. Furthermore, every first cassette of a sub-batch determines the routing for all cassettes in that sub-batch (wormhole routing, see Section 2.2.3). Once a machine starts processing the first cassette of a sub-batch, it will only process cassettes from that specific sub-batch until the last cassette from that sub-batch has been processed. Between processing two different sub-batches a changeover time might be required at a machine, during this changeover a machine will be down. Since the machines in a flow line are the same machines as in a job shop, Section 3.2.1 also describes the machine failures in a job shop setting. The periodic maintenance for the Multi Plunger in a job shop is also similar to the Multi Plunger's periodic maintenance process in a flow line.

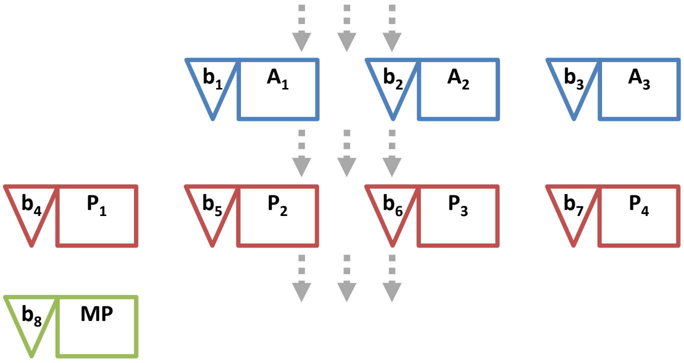


Figure 3.4: Machine layout in a job shop.

3.4 Performance measures

The performance measures that are used in this project are either related to a machine or to an order. One performance measure may be more interesting than another one, but in general, the most important performance measure is average line throughput, usually measured in (thousands of) units per hour. Whenever there is only one Multi Plunger in a production network, the line throughput is defined as the throughput of that Multi Plunger. Obviously, the average throughput per hour is strongly related to other performance measures. For example, an increase in utilization usually results in an increase of line throughput as well. Note that the performance measures referring to a fraction of time are calculated for the period that that specific machine is actually participating in the network. This period starts with the first arrival of a product at a machine and ends with the last service completion at the machine.

1. Machine based performance measures

   Throughput: the average number of products processed per hour.

   Up fraction: the fraction of time the machine is up.

   Utilization: the fraction of time the machine is both up and assembling products, note that this excludes indexing.

   Starved fraction: the fraction of time the machine is starved.

   Down fraction: the fraction of time the machine is down due to error occurrence.

   Setup fraction: the fraction of time the machine is in setup.

   Blocked fraction: the fraction of time the machine is blocked.

   Maximum buffer level: the highest buffer level at a machine. In a flow line this is measured in total number of products, whereas in a job shop the maximum buffer size is determined by looking at the number of cassettes. Note that the maximum buffer level statistic for the ADATs may be distorted since all cassettes of a sub-batch are released at once, disregarding buffer sizes.

2. Order based performance measures

   Sojourn time: the time between the release of the first sub-batch of an order until the last service completion for an order. The sojourn time for every order is stored and this data can be used to create an empirical distribution function, or just the average and variation, of the sojourn time for a certain product type.

3.5 Simulator description

Since the NXP assembly process for the SOT457 product package has many details, a discrete event simulator has been implemented in MATLAB to imitate the production environment. Although the simulator has been developed in MATLAB R2010b 7.11, the simulator can also be using in any MATLAB version higher than or equal to R2008b 7.7. Furthermore it is worth mentioning that the simulator has been developed in an object oriented way and the actual source code contains a lot of inline comment. Hence, anyone with some experience in object oriented programming should be able to understand the source code and adapt the developed simulator where needed. This makes the simulator ideally suitable for future research projects for NXP Semiconductors.

In this section the most important objects of the discrete event simulator will be highlighted. The state of a job shop or flow line network can be described by the following parameters:

1. for every machine:
   - the current machine state (up, down, in setup, in (periodic) maintenance, or blocked)
   - the residual up, down, maintenance, setup or service time (note that service time refers to the time it takes to process a cassette at one machine, do not confuse service with maintenance or repair)
2. for every cassette

  - the location in the network
  - the production stage
  - the sub-batch and order it belongs to
  - if applicable, the residual transportation time and the target machine

3. for every sub-batch
  - the routing determined so far
  - the cassettes belonging to this sub-batch

The cassette and sub-batch information required to describe the exact state of the network is rather evident and does not change very often. However, the machine information does change regularly. The change in residual times for a machine is rather clear and the moment a residual is 0, then something (usually the state of the machine) will change at a machine. Figure 3.5 depicts the state changes of one specific machine as the result of the occurrence of an event; these type of flow diagrams really contain the essential concept of a discrete event simulator. To keep the flow diagram transparent, some details are left out. For example, one can only start processing a cassette if the correct cassette is located at that machine. Also note that some flow-arrows are missing, for example, it is not possible to change the "in setup" state to the "in maintenance" state since preventive maintenance is only required at a MultiPlunger, but changeovers are not required at the MP, hence such a state-change can never occur. A production network now consists of several of these machine states that are connected, e.g. a cassette service completion at one machine will result in an order arrival at another machine.

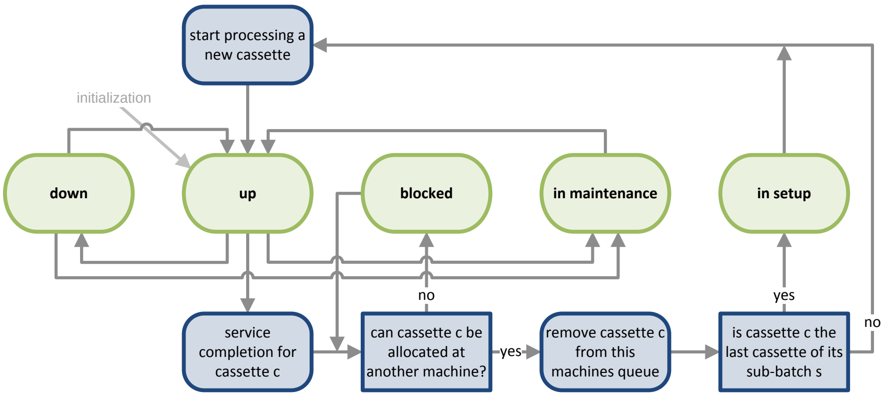

start processing a new cassette
initialization
down
up
blocked
in maintenance
in setup
no
yes
no
service completion for cassette c
can cassette c be allocated at another machine?
yes
remove cassette c from this machines queue
is cassette c the last cassette of its sub-batch s

Figure 3.5: State change diagram from machine perspective.

3.5.1 Simulator input

There are many ways to start a simulation, the usage of the GUI and other input data are explained in appendix B.

3.5.2 Simulator output

There are three different types of output. First of all, the machine based performance measures will be listed in a table in the GUI. Secondly, one can view the order sojourn time empirical distribution function along with the mean and variance of the order sojourn time. All these order-based statistics are split per product type; note that the order sojourn time is strongly related to the order size and once there are many orders for one product that vary in size, the variance in the order sojourn time will be high for that product. The third type of output is a time line visualizing the production and does not relate to the performance measures from Section 3.4. This time line is ideally suitable to get more insight in the way a production network reacts to disturbances such as machine failures, periodic maintenance and changeover times. An example of a production visualization for a small job shop is given in Figure 3.6. In this figure one can clearly see that the ADAT is blocked at the end due to the downtime at PHICOM 1 and the maintenance of the Multi Plunger causes a blockage of PHICOM 1 as well. Although the time line is a very powerful feature of the simulator, it will not be used in this report because it is less appropriate to compare a flow line and a job shop production concept.

3.5.3 Simulator performance

The fluid models that were used in the previous projects all resulted in a very efficient and fast simulator. Because of the discrete items (such as orders, sub-batches and cassettes) being simulated in this project, the real-time performance of the newly developed simulator is obviously worse when simulating a flow line. Many different strategies (e.g. incorporating downtime in uptime duration, using linked lists instead of object arrays) have been implemented in order to make the simulator a factor 10 faster. The built-in MATLAB performance analyzer revealed that a major part of the running time could be attributed to MATLAB's internal garbage collector. Once MATLAB is able to solve its problems with the garbage collector in a future release, the running time will very likely decrease significantly.

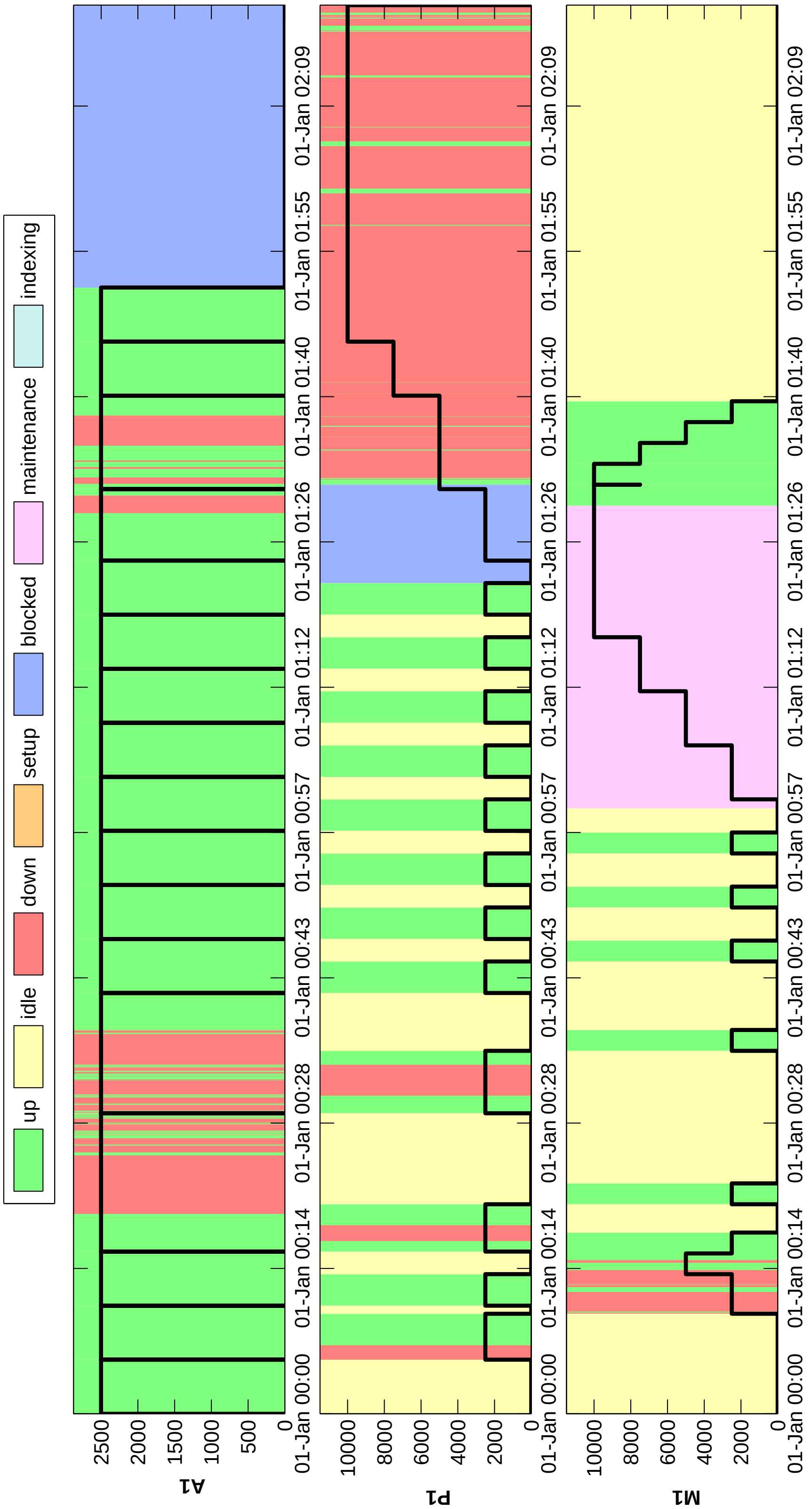


Figure 3.6: Visualization of the production in a small job shop; the color indicates the machine state at a certain time and the thick black line illustrates the number of products at a machine (both in process and in the buffer).

CHAPTER 4

Validation

One of the most important steps in the development of a simulator is the validation process. Therefore the following three validation subjects will be discussed:

1. Is the simulator correctly implemented.
2. Is the predicted output of the simulator correct.
3. Does the simulator's model represent reality.

For the first part, every single step the simulator made was monitored. However, stating all the different network states and steps for different scenarios will be beyond anyone's interest. Hence, that part of the validation process will not be discussed in this master's thesis; the other two parts will be discussed in this chapter. In Section 4.1 the results of running the simulator for some simplified production networks are compared with the results from a theoretical analysis. Then, in Section 4.2, the discretization error will be investigated by comparing the results from simulating a discretized flow line with the results of simulating a continuous flow line using the simulator that has been developed in a previous project. Once one is convinced that the simulator has been correctly implemented, the underlying model will be validated in Section 4.3 by comparing the simulator's results with results based on empirical production data from the AWACS database.

4.1 Theoretical results

Although the simulator has been developed to simulate an NXP production network, many other production networks can be simulated as well. For example, in practice there is an infinite supply of orders that are available from the start. However, the simulator does provide the option to delay the arrival of an order resulting in a Poisson order arrival process. In this section the results of the simulator will be compared with exact or good approximate results. Below the production "networks" that have been tested in this section are listed. For all the scenarios, only the simulator's prediction for the order sojourn time will be compared with the theoretical value.

Scenario 1: An M/M/1 queue (i.e., a network with 1 machine, exponentially distributed service and inter-arrival times.

Scenario 2: An M/D/1 queue (i.e., a network with 1 machine, deterministic service times and exponential inter-arrival times.

Scenario 3: A tandem queue with 2 machines, exponentially distributed service (machine 1 and 2) and inter-arrival times (machine 1 only).

Scenario 4: A job shop with 3 machines, exponentially distributed service and inter-arrival times, only 1 machine visit needed.

Scenario 5: A job shop with 3 ADATs, 3 PHICOMs, exponentially distributed service and inter-arrival times, exactly one ADAT and PHICOM visit are required per product.

Scenario 6: A job shop with 3 machines, deterministic service and exponential inter-arrival times, only 1 machine visit needed.

Scenario 7: A tandem queue with 2 machines, deterministic service times and exponential inter-arrival times (machine 1 only).

Scenario 8: A tandem queue with 2 machines, constant machine speeds, infinite order supply at machine 1, finite intermediate buffer and exponential up- and downtimes.

4.1.1 Scenario 1

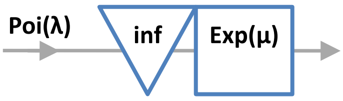


Figure 4.1: Schematic display of the network in scenario 1.

The most straightforward production network that allows an exact analysis is a network with one machine with an infinite buffer where both the time between the arrivals of two subsequent orders and the processing time of an order at a machine is exponentially distributed. Below the parameters for such an M/M/1 queue are given, the specific values have been chosen arbitrarily.

Number of machines	1
Buffer size	Infinite
Arrival rate (λ)	50 orders per hour
Service rate (μ)	$\frac{45000}{560}$
Occupation rate (ρ)	$\frac{\lambda}{\mu} = 0.622$

For the developed simulator this translates to the following input:

Number of machines	1 ADAT
Buffer size	Infinite
Arrival rate	50 orders per hour
Machine speed	45000 products per hour
Order input	700 x 560 products (requiring 1 die only)
Mean uptime	Infinite
Mean downtime	0
Replications	10

By solving the global balance equations for this system, the probability p_n that an arriving order finds n orders in a queue is given by (with occupation rate $\rho = \frac{\lambda}{\mu}$)

$$p_n = (1-\rho)\rho^n.$$

Hence, an arriving customer will, due to the PASTA (Poisson Arrivals See Time Averages) property and the memoryless property of the service time distribution, with probability p_n have to wait for n full service completions. The actual service times of an arriving customer will therefore, with probability p_n be the sum of $n+1$ exponentials. Using the Laplace Stieltjes transform of the sojourn time S, one can, by conditioning on the number of orders in a network, show that

$$\begin{aligned}\widetilde{S}(s) &= E(e^{-sS}) \\ &= \sum_{i=0}^{\infty} p_i E(e^{-s(B_1+B_2+\ldots+B_{i+1})}) \\ &= \sum_{i=0}^{\infty}(1-\rho)\rho^i E(e^{-sB_1})E(e^{-sB_2})\ldots E(e^{-sB_{i+1}}). \end{aligned} \tag{4.1}$$

Where B_i refers to the service time of the i^{th} customer. Since the service times are independent and exponentially distributed, one finds that

$$\widetilde{B}_i(s) = \frac{\mu}{\mu+s}. \tag{4.2}$$

Substitution of (4.2) in (4.1) yields that

$$\begin{aligned}\widetilde{S}(s) &= \sum_{i=0}^{\infty}(1-\rho)\rho^i E(e^{-sB_1})E(e^{-sB_2})\ldots E(e^{-sB_{n+1}}) \\ &= \sum_{i=0}^{\infty}(1-\rho)\rho^i \left(\frac{\mu}{\mu+s}\right)^i \\ &= \frac{\mu(1-\rho)}{\mu(1-\rho)+s}. \end{aligned}$$

Hence, the order sojourn time in an M/M/1 queue is exponentially distributed with rate parameter $\mu(1-\rho)$. Therefore, the expected sojourn time is given by

$$E(S) = \frac{1}{\mu(1-\rho)}. \tag{4.3}$$

For the previously described M/M/1 queue, the expected sojourn time is therefore equal to $\big(\frac{45000}{560}(1-0.622)\big)^{-1} = 0.0329h = 118.59s$. The estimated average sojourn time based on the simulator results is 118.30, whereas the estimated standard deviation was 114.99, the first 125 orders of every simulation run have been ignored in order to remove the warmup effect. Based on a one sample T-test, the $1-\frac{\alpha}{2}$ confidence interval for the estimated sample mean $\overline{\mu}$ of size n with unknown variance $\overline{\sigma}$ is given by $\overline{\mu} \pm t^{n-1}_{1-\alpha/2} \cdot \frac{\overline{\sigma}}{\sqrt{n}}$. The usage of a T-test is rectified by the classical Central Limit Theorem. Hence, for this simulated M/M/1 queue the 95% confidence interval for the mean is $118.30 \pm 1.96 \cdot \frac{114.99}{\sqrt{7000-1250-1}} = 118.30 \pm 2.97$. The theoretical value fits well within this confidence interval. Also note that the estimated sojourn time standard deviation is approximately equal to the mean value. This makes sense since the sojourn time in an M/M/1 queue is exponentially distributed. Hence, one can conclude that the developed simulator is able to simulate an M/M/1 queue.

4.1.2 Scenario 2

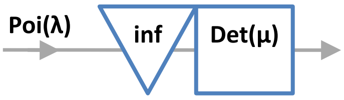


Figure 4.2: Schematic display of the network in scenario 2.

For the regular NXP production lines, the processing times are assumed to be deterministic, hence it seems obvious to change the exponential service times to deterministic service times. Although one can determine the Laplace-Stieltjes transform (LST) of the limiting distribution for the number of customers in the system, one has to invert this LST in order to determine the actual distribution of the sojourn time. Unfortunately, inverting this LST is very difficult, hence, one cannot give a closed form expression for the sojourn time distribution in a M/D/1 queue. However, one can use a mean value approach to calculate the exact mean sojourn time. The PASTA property once again implies that an arriving customer sees time averages. Since the network will be empty for a fraction of $1-\rho$ of the time (once again using occupation rate $\rho = \frac{\lambda}{\mu}$), an arriving customer will have to wait for other orders with probability ρ. When an arriving customer finds n customers in the queue (with $n \geq 1$), it has to wait for 1 residual and $n-1$ full service completions. With L_q the number of orders in a machines queue (excluding the order that is being processed), B the service time duration and R the residual service time duration upon arrival of an order, the expected waiting time W in an M/D/1 is given by

$$\mathrm{E}(W) = (1-\rho)\cdot 0 + \rho\mathrm{E}(R) + \mathrm{E}(L_q)\mathrm{E}(B). \tag{4.4}$$

Furthermore, Little's Law[5] applied to queues states that

$$E(L_q) = \lambda\mathrm{E}(W). \tag{4.5}$$

Combining (4.4) and (4.5) gives that , one finds that

$$\mathrm{E}(W) = \frac{\rho\mathrm{E}(R)}{1-\rho}.$$

For the M/D/1 queue the service time is constant, hence $\mathrm{E}(B) = \mu^{-1}$ and $\mathrm{Var}(B) = \mathrm{E}(B^2) - \mathrm{E}(B)^2 = 0$. Therefore, $\mathrm{E}(R) = \frac{\mathrm{E}(B^2)}{2\mathrm{E}(B)} = \frac{1}{2\mu}$. The expected sojourn time is now given by

$$\mathrm{E}(S) = \frac{2-\rho}{2\mu(1-\rho)} \tag{4.6}$$

Using the parameters from the list above, (4.6) yields an expected sojourn time of $0.0227h = 81.659s$. The estimated mean and standard deviation based on the simulator's results are 81.25 and 50.60 respectively. Note the first 50 orders of every simulation run have been ignored in order to remove the warmup effects. By using the same approach as in the M/M/1 case, this results in a 95% confidence interval of 81.25 ± 1.23. Again, the theoretical value fits well within this confidence interval.

4.1.3 Scenario 3

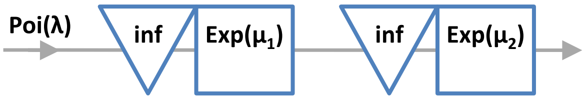


Figure 4.3: Schematic display of the network in scenario 3.

Another extension for the M/M/1 is adding another server after the first machine. This way a mini flow line is created where orders arrive at the first machine and leave the network when they have been processed at the second machine. For validating a straightforward flow line, the order sojourn time of a tandem queue with the parameters below will both be calculated and simulated.

Number of machines	2
Buffer size	Infinite
Routing	Arrival at machine 1, service at machine 1, service at machine 2
Arrival rate at machine 1 (λ)	50 orders per hour
Service rate machine 1 (μ_1)	$\frac{45000}{560}$
Service rate machine 2 (μ_2)	$\frac{60000}{560}$
Occupation rate machine 1 (ρ_1)	0.622
Occupation rate machine 2 (ρ_2)	0.467

For the developed simulator this translates to the following input:

Number of machines	1 ADAT, 1 PHICOM
Buffer sizes	Infinite
Arrival rate	50 orders per hour
ADAT speed	45000 products per hour
PHICOM speed	60000 products per hour
Order input	700 x 560 products (requiring 1 die only)
Mean uptime	Infinite
Mean downtime	0
Replications	10

Jackson [6] showed that in an open Jackson network of M/M/1 queues with machine utilization $\rho < 1$, the joint equilibrium state probability distribution function is the product of the equilibrium distributions function of the individual queues. Hence, the expected sojourn time is given by

$$\mathrm{E}(S) = \mathrm{E}(S_1) + \mathrm{E}(S_2) = \frac{1}{\mu_1(1-\rho_1)} + \frac{1}{\mu_2(1-\rho_2)}. \tag{4.7}$$

Substituting the parameters above in (4.7) yields an expected order sojourn time of $0.0504h = 181.59s$. The estimated mean and standard deviation based on the simulator's results are 179.95 and 129.90 respectively. Note that the first 160 orders of every simulation run have been ignored in order to remove the warmup effects. By using the same approach as in the M/M/1 case, this results in a 95% confidence interval of 179.95 ± 3.36. Again, the theoretical value fits well within this confidence interval.

4.1.4 Scenario 4

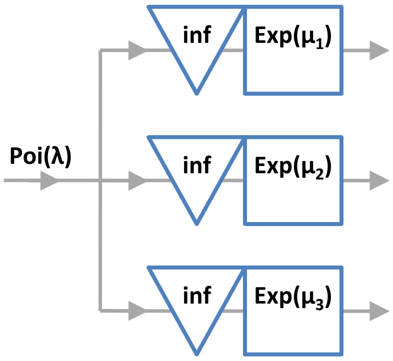


Figure 4.4: Schematic display of the network in scenario 4.

The fourth scenario is used to validate a job shop setting. However, not every job shop allows an exact analysis, in fact, many assumptions are required. For this scenario a network with only three machines will be used, the arrival process will again be Poissonian and the service times at the machines are exponentially distributed. When an order arrives it goes to one of the three machines, all with equal probability; after being processed at a machine the order immediately leaves the network. For the validation, the following parameters are used.

Number of machines	3
Buffer size	Infinite
Arrival rate (λ)	50 orders per hour
Service rate machine 1, 2, 3 (μ_1, μ_2, μ_3)	$\frac{15000}{560}$
Occupation rate machine 1, 2, 3 (ρ_1, ρ_2, ρ_3)	0.622

For the developed simulator this translates to the following input:

Number of machines	3 ADATs
Buffer sizes	Infinite
Arrival rate	50 orders per hour
ADAT speed	15000 products per hour
Order input	700 x 560 products (requiring 1 die only)
Mean uptime	Infinite
Mean downtime	0
Routing	Random
Replications	10

The result of the completely random routing is that the Poisson process can be split into three separate sub processes, one for the arrival process at each machine. These sub-processes are also Poissonian, each with rate $\frac{\lambda}{3} = 15$ orders per hour. The result is that the expected sojourn time at machine i can be calculated using M/M/1 formula (4.3). Hence, $\mathrm{E}(S_i) = (\mu_i(1-\rho_i))^{-1} = 0.099h = 355.76s$. The estimated mean and standard deviation based on the simulator's results are 357.15 and 346.64 respectively. Note that the first 75 orders of every simulation run have been ignored in order to remove the warmup effects, resulting in a 95% confidence interval of 357.15 ± 8.60. Again, the theoretical value fits well within this confidence interval. If another of the implemented routing algorithms would have been used, the Poisson process could not have been split into separate Poisson processes and this analysis could not have been performed.

4.1.5 Scenario 5

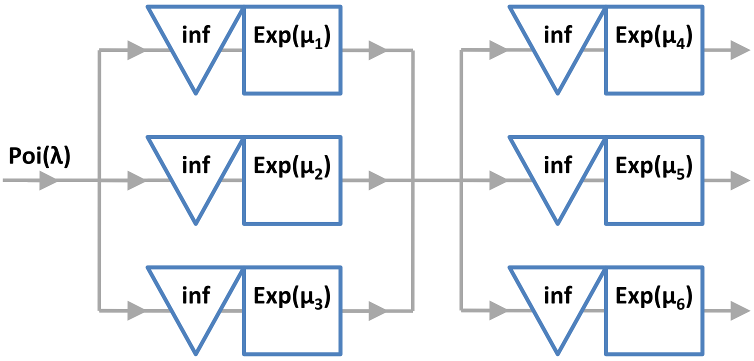


Figure 4.5: Schematic display of the network in scenario 5.

The network discussed the previous scenario can be extended by adding three PHICOMs and requiring one ADAT and one PHICOM visit per order, this results in the following parameters.

Number of machines	3 + 3
Buffer size	Infinite
Arrival rate	50 orders per hour
Service rate machine 1, 2, 3 (μ_1, μ_2, μ_3)	$\frac{15000}{560}$
Service rate machine 4, 5, 6 (μ_4, μ_5, μ_6)	$\frac{12500}{560}$
Occupation rate machine 1, 2, 3 (ρ_1, ρ_2, ρ_3)	0.622
Occupation rate machine 4, 5, 6 (ρ_4, ρ_5, ρ_6)	0.446

For the developed simulator this translates to the following input:

Number of machines	3 ADATs, 3 PHICOMs
Buffer sizes	Infinite
Arrival rate	50 orders per hour
ADAT speed	15000 products per hour
PHICOM speed	12500 products per hour
Order input	700 x 560 products (requiring 1 die and 1 wire)
Mean uptime	Infinite
Mean downtime	0
Routing	Random
Replications	10

Just like in scenario 3, the network once again is a feed forward Jackson network and the joint equilibrium state probability distribution is once again equal to the product of the equilibrium distributions functions of the individual queues. Since machines 1, 2, 3 and 4, 5, 6 are identical, the expected order sojourn time is given by

$$E(S) = E(S_{1,2,3}) + E(S_{3,4,5}) = \frac{1}{\mu_1(1-\rho_1)} + \frac{1}{\mu_4(1-\rho_4)}. \tag{4.8}$$

Substituting the parameters above in (4.8) yields an expected order sojourn time of $992.40s$. The estimated mean and standard deviation based on the simulator's results are 994.13 and 714.81 respectively. Note that the first 200 orders of every simulation run have been ignored in order to remove the warmup effects, resulting in a 95% confidence interval of 994.13 ± 19.82. Again, the theoretical value fits well within this confidence interval. If any of the other implemented routing algorithms would have been used, this analysis could not have been performed.

4.1.6 Scenario 6

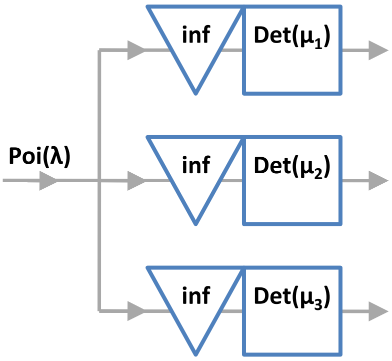


Figure 4.6: Schematic display of the network in scenario 6.

This scenario is the result of changing the exponential service times of scenario 4, to deterministic service times. Although all parameters remain the same, a different analysis is required. Just like in the analysis of scenario 4, the Poisson arrival process can be split into three parts, one arrival steam for every machine. The result now is that the network consists of three identical M/D/1 queues. Using equation (4.6) from scenario 2, the order sojourn time in this network is given by

$$\mathrm{E}(S) = \frac{2-\rho_{1,2,3}}{2\mu_{1,2,3}(1-\rho_{1,2,3})} = 0.0681h = 245.08s$$

The estimated mean and standard deviation based on the simulator's results are 245.52 and 158.43 respectively. Note that the first 250 orders of every simulation run have been ignored in order to remove the warmup effects, resulting in a 95% confidence interval of 245.52 ± 3.79. Again, the theoretical value fits within this confidence interval.

4.1.7 Scenario 7

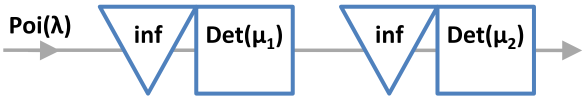


Figure 4.7: Schematic display of the network in scenario 7.

This scenario is, apart from the deterministic service times, similar to the situation in scenario 3. The exact parameters, listed below, slightly differ.

Number of machines	2
Buffer size	Infinite
Routing	Arrival at machine 1, service at machine 1, service at machine 2
Arrival rate at machine 1 (λ)	50 orders per hour
Service rate machine 1 (μ_1)	$\frac{45000}{560}$
Service rate machine 1 (μ_2)	$\frac{33500}{560}$
Occupation rate machine 1 (ρ_1)	0.622
Occupation rate machine 2 (ρ_2)	0.836

For the developed simulator this translates to the following input:

Number of machines	1 ADAT, 1 PHICOM
Buffer sizes	Infinite
Arrival rate	50 orders per hour
ADAT speed	45000 products per hour
PHICOM speed	33500 products per hour
Order input	700 x 560 products (requiring 1 die and 1 wire)
Mean uptime	Infinite
Mean downtime	0
Replications	10

Because the processing times are no longer exponential, the sojourn times at machine 1 and 2 are no longer independent. In fact, if the speed of machine 2 is smaller than the speed of machine 1, then a short waiting time at machine 1 implies an even shorter waiting time at machine 2. The opposite is also true, if the speed of machine 2 is higher than the speed of machine 1, then a long waiting time at machine 1 implies an even longer waiting time at machine 2. However, if one ignores this fact and just assumes that the sojourn times are independent, then an approximation for the mean sojourn time at machine i is given by

$$\frac{\rho_i}{1-\rho_i}\frac{c_{A_i}^2+c_{B_i}^2}{2}\mathrm{E}(B_i)+\mathrm{E}(B_i)$$

with

$$c_{A_i}^2 = (1-\rho_{i-1}^2)c_{A_{i-1}}^2 + \rho_{i-1}^2 c_{B_{i-1}}^2 \text{ for } i = 2,3,\ldots, \tag{4.9}$$

where $c_X^2 = \frac{\text{Var}(X)}{\text{E}(X)^2}$ denotes the squared coefficient of variation for process X (e.g. the inter-arrival or service times at machine 2). Using (4.9), an approximation for the mean order sojourn time is $235.75s$. The estimated mean based on the simulator's results is $247.16s$, which is significantly bigger than the theoretical value. However, this does make sense since the speed of machine 2 is lower than the speed of machine 1, hence, the waiting times will be positively correlated and the approximation will underestimate the actual value.

4.1.8 Scenario 8

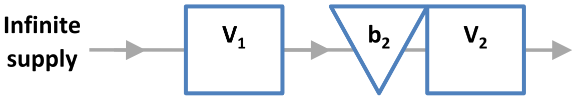


Figure 4.8: Schematic display of the network in scenario 8.

This continuous fluid model related scenario is completely different from the previous discrete scenarios. Whereas there were order arrivals in the previous scenarios, for this scenario it is assumed that there is an infinite demand. Machines are no longer processing orders, but they are assembling products at a certain fixed rate. The specific network regarded for this scenario is a flow line with 2 machines, every machine produces products at a certain fixed speed and there is a finite buffer between machine 1 and 2. If that buffer is full, then machine 1 is blocked and slows down to the same machine speed as machine 2. If the buffer is empty, then machine 2 is starved and stops working. Meanwhile, operation dependent errors can occur at any of the two machines, both the up- and the downtimes are exponentially distributed. The specific parameters for such a network with identical machines are given below.

Number of machines	2
Intermediate buffer size (N)	7500
Routing	Arrival at machine 1, service at machine 1, service at machine 2
Speed machine 1, 2 (v)	45000 products per hour
Uptime distribution machine 1, 2	Exp(λ)
Downtime distribution machine 1, 2	Exp(μ)

For the developed simulator this translates to the following input:

Number of machines	1 ADAT, 1 PHICOM
Buffer sizes	b_1 infinite, $b_2 = 7500$
ADAT speed	45000 products per hour
PHICOM speed	45000 products per hour
Order input	1000 x 100 products (requiring 1 die and 1 wire)
Mean uptime machine 1, 2	1033 s
Standard deviation uptime machine 1, 2	1033 s
Mean downtime machine 1, 2	243 s
Standard deviation downtime machine 1, 2	243 s
Replications	10

Li et al. [7] show that, by solving the parametric equations describing the steady-state behavior from

Gershwin [8], the line throughput for a fluid model with two identical machines is given by

$$\mathrm{TP} = \frac{v\Big(2v + N\mu\big(1+\frac{\lambda}{\mu}\big)\Big)}{2v\big(1+\frac{2\lambda}{\mu}\big) + N\mu\big(1+\frac{\lambda}{\mu}\big)^2}\,. \tag{4.10}$$

Substituting the specific model parameters in equation (4.10) yields that the mean throughput will be 33875 products per hour. The 95% confidence interval for the estimated throughput based on simulating 200x250 orders of various size (and therefore various discretization levels) are listed in Table 4.1. As can been seen, the line throughput seems to increase when the discretization gets finer. Unfortunately, this effect cannot fully be explained. One possible cause might be the duration of a blockage. In the continuous case machine 1 will slow down to the speed of machine 2 when the buffer is full, whereas in the discretized flow line machine 1 will only be blocked if a recently processed cassette cannot be stored at machine 2. However, as soon as there is enough buffer capacity available at machine 2, then the processed cassette from machine 1 immediately joins the queue of machine 2 and machine 1 is unblocked. Hence, the duration of the blockage of machine 1 in a discretized flow line is equal to the residual processing time at machine 2. These two procedures do differ quite a lot and will cause some discrepancies. However, for the network described in this scenario, the relative difference between a continuous and a discrete flow line is about 1% for discretization levels of $\frac{1}{4}$ up to 1 times the smallest buffer. In the next section the continuous and discretized flow line are compared more thoroughly.

Discretization	line throughput (kUph)	relative error
500	34871 ± 284	2.94%
1500	34409 ± 166	1.58%
2500	34127 ± 134	0.75%
3750	33954 ± 107	0.23%
7500	32184 ± 71	−4.99%

Table 4.1: Line throughput for different discretizations (i.e., products per chunk of leadframe) for scenario 8.

4.2 Comparison with fluid model

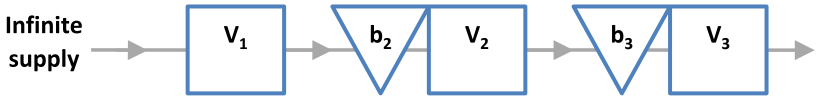


Figure 4.9: Schematic display of the network used for determining the discretization error.

If one wants to determine the size of the discretization error, one can only compare the results from the newly developed simulator with the results from the simulator developed in the previous projects, since most continuous flow lines do not allow an exact analysis. Therefore a production network as depicted in Figure 4.9 will be simulated using both simulators. The precise input is listed below.

Number of machines	1 ADAT, 1 PHICOM, 1Multi Plunger
Buffer sizes	b_1 infinite, $b_2 = 5750$, $b_2 = 15000$
ADAT speed (v_1)	28000 products per hour
PHICOM speed (v_2)	45000 products per hour
Multi Plunger speed (v_3)	33500 products per hour
Mean uptime machine 1, 2, 3	1033 s
Standard deviation uptime machine 1, 2, 3	1033 s
Mean downtime machine 1, 2, 3	243 s
Standard deviation downtime machine 1, 2, 3	243 s
Replications	10

The throughput, after warmup correction, for different discretizations is listed in Table 4.2, the relative error in this table is based on the difference between the continuous and discrete simulator. The (continuous) fluid model simulator yielded a throughput of 22417 ± 73 and for every discretization the confidence intervals overlap. More results, for different networks, are listed in Appendix C.

Discretization	line throughput (kUph)	relative error
15000	22583 ± 80	0.7%
7500	22601 ± 132	0.8%
5750	22630 ± 134	1.0%
2875	22662 ± 189	1.1%
1438	22513 ± 297	0.4%
719	22191 ± 465	−1%

Table 4.2: Line throughput for different discretizations (i.e., products per chunk of leadframe).

There are two conclusions that can be drawn from the tables in this section and the appendix. First of all, a general rule of thumb is to use a discretization of $\frac{1}{4}$ up to 1 times the smallest buffer capacity. A discretization of $\frac{1}{2}$ times the buffer size seems to be most appropriate. In some exceptional cases, for example when the bottleneck for the line throughput is the starvation and not the blocking of machines, a rougher discretization ($\frac{1}{2}$ up to 1 times the bottleneck's buffer capacity) can be used. The second conclusion, based on the results in the appendix, is that it is hard to give an estimate for the discretization error. In a small network the discretization error is less than 1% for the advised discretizations. However, in a bigger fully balanced network, the discretization error can increase up to 5%. A possible explanation for this effect has been given in the previous section. However, one should bear in mind that this difference does not imply that the model used in this master's thesis is unable to accurately describe a NXP production line. That topic is discussed in the next section.

4.3 Using production data

Now that the simulator has been validated using theoretical models, the underlying model for a flow line has to be validated, the validation of a job shop assembly network will be skipped since there is no production data available. However, for the flow line two type of data sets are available: one for a SOT23 and one for a SOT457 production line. The SOT23 production data is used to validate a flow line without changeovers, the SOT457 data is used to validate a flow line with changeover times.

4.3.1 SOT23 - flow line without changeovers

Table 4.3 lists the machine statistics for NXP's production line 48 between April, 1, 2011 up to May, 1, 2011. The fraction-rows indicate the fraction of time that a machine is used, down, blocked or starved. Since only the duration of a certain state and the number of products produced can be extracted from the AWACS database, the machine speed is calculated by dividing the total number of products produced by a machine by the total production time of a machine. The throughput of a machine is determined by once again adding the number of products produced in every production state and then divide by the total time that a machine is either in production, down, blocked or starved (periodic maintenance is included in the downtime). Note that the Multi Plunger throughput is almost 4 times bigger than the throughput of any other machine because the SOT23 production line uses a 4-tracked leadframe. Apart from the Multi Plunger that processes all 4 tracks, all other machines only process 1 track of leadframe in this production line. Table 4.3 lists statistics for the machines in line 49 between April, 1, 2011 and May, 1, 2011. Unfortunately, the data in Table 4.3 might not be very reliable. For example, in a flow line one expects that, due to the conservation of flow principle, the throughput is roughly the same for every machine. The 4-track corrected throughput for the Multi Plunger is 18571 Uph, therefore the throughput of line 48 varies between 18571 and 20062 units per hour. The line throughput for this flow line is defined as the average of all eight separate machine throughputs. Following this rule, the line 48 throughput is 19361 units per hour. An explanation for the difference in machine throughput might lie in the way that the data is collected. For example, not only has extremely long downtimes been excluded from the downtime sample, but some other machine specific states are also ignored. Nevertheless, one simply has to accept that there might be some errors in the production data and one should not focus too much on statistics for one machine, but focus more on performance measures based on the entire production line.

The machine speed information based on Table 4.3, along with the detailed up- and downtimes per machine can be used as input for the simulator. The exact input is listed below. The simulators detailed results for simulating 1000 runs of 100 orders (each for different order sizes and therefore also different discretizations) are listed in Table 4.5. Even though the variation in throughput between the machines in the flow line is fairly small, the total line throughput is once again defined as the average of all machines throughputs, this data is listed in Table 4.4. Although the data set contains production information for a period of one month, more days have in fact been simulated. As can been seen, the results for the different discretizations do not differ that much and especially the 1128 and 2252 product per chunk (referring to $\frac{1}{2}$ and 1 times the smallest buffer size) discretizations are very close. The discretization using 562 products per chunk shows greater contrast with the empirical values. This corresponds with the conclusion from the validation in Section 4.1.8: one should not use a discretization that is too small since that would overestimate the line throughput. If one ignores the specific fraction related performance measures and focuses on the most important performance measure, i.e., the line throughput, then the relative error is smaller than 1% for the advised discretization levels. This error is actually quite small, especially if one bears in mind that statistics such as line throughput are strongly related to input parameters such a machine speed, which had to be estimated. Obviously, if there are errors in the machine speed, then these errors also have an effect on the line throughput. All in all, one can conclude that the developed simulator is able to accurately predict key performance indicators such as line throughput. Hence, the developed simulator can perfectly be used to get insight in the differences between a flow line and a job shop assembly network.

Number of machines	4 ADATs, 4 PHICOMs, 1Multi Plunger
Buffer sizes	$b_1 = \text{inf}$, $b_2, \ldots, b_4 = 2250$, $b_5, \ldots, b_9 = 400$
ADAT speed (v_1)	22771 dies per hour
PHICOM speed (v_2)	23020 wires per hour
Multi Plunger speed (v_3)	89357 molds per hour
Order input	product requiring 1 die, 1 wire, 1 mold
Mean uptime ADAT1, 2, 3, 4	1327, 1817, 2408, 1978 s
Mean downtime ADAT1, 2, 3, 4	78, 104, 100, 78 s
Mean uptime PHICOM1, 2, 3, 4	1436, 1463, 1620, 1521 s
Mean downtime PHICOM1, 2, 3, 4	88, 116, 108, 100 s
Mean uptime MP1	585 s
Mean downtime MP1	74 s
Up- and downtime sampling	Fit distribution
Replications	10

Statistic	A1	A2	A3	A4	P1	P2	P3	P4	MP1
Utilization	0.86	0.86	0.85	0.86	0.84	0.84	0.86	0.83	0.83
Down fraction	0.05	0.05	0.03	0.03	0.05	0.07	0.06	0.05	0.10
Blocked fraction	0.07	0.08	0.08	0.07	0.06	0.05	0.00	0.04	0.00
Starved fraction	0.01	0.01	0.04	0.05	0.05	0.05	0.08	0.08	0.06
Machine speed (Uph)	22818	22784	22642	22838	22725	23160	23237	22954	89357
Throughput (Uph)	19716	19581	19164	19552	19148	19375	20062	19081	74282

Table 4.3: Empirical machine statistics for NXP SOT23 production line 48 between Apr-01-2011 and May-01-2011.

Discretization	line throughput (kUph)	relative error
562	19492	0.68%
1128	19300	−0.32%
2552	18694	−3.44%

Table 4.4: Line throughput for different discretizations (i.e., products per chunk of leadframe), the relative error is based on an empirical throughput of 19361 Uph.

4.3.2 SOT457 - flow line with changeovers

Unfortunately it was not possible to obtain all the required production data for a SOT457 production line, hence, this validation procedure had to be skipped. When the data is available, the end-user of the simulator can always try performing this validation.

	A1	A2	A3	A4	P1	P2	P3	P4	MP1
Discretization: 562 products per chunk of leadframe									
Utilization	0.85	0.85	0.85	0.85	0.84	0.84	0.84	0.84	0.87
Starved fraction	0.05	0.05	0.04	0.03	0.05	0.07	0.06	0.05	0.11
Down fraction	0.09	0.08	0.07	0.07	0.08	0.07	0.07	0.08	0.00
Blocked fraction	0.00	0.02	0.04	0.05	0.03	0.02	0.03	0.03	0.02
Throughput (Uph)	19464	19463	19489	19505	19497	19519	19510	19491	77944
	± 90	± 89	± 88	± 84	± 83	± 83	± 82	± 79	± 299
Discretization: 1128 products per chunk of leadframe									
Utilization	0.85	0.84	0.84	0.84	0.83	0.83	0.83	0.83	0.86
Down fraction	0.05	0.05	0.04	0.03	0.05	0.07	0.06	0.05	0.11
Blocked fraction	0.10	0.08	0.07	0.06	0.07	0.06	0.07	0.08	0.00
Starved fraction	0.00	0.03	0.05	0.06	0.04	0.03	0.04	0.04	0.03
Throughput (Uph)	19256	19263	19284	19302	19309	19328	19317	19314	77300
	± 58	± 57	± 56	± 55	± 53	± 52	± 51	± 51	± 207
Discretization: 2252 products per chunk of leadframe									
Utilization	0.82	0.81	0.81	0.80	0.80	0.81	0.81	0.80	0.83
Down fraction	0.05	0.05	0.03	0.03	0.05	0.06	0.05	0.05	0.11
Blocked fraction	0.13	0.10	0.08	0.07	0.07	0.06	0.06	0.07	0.00
Starved fraction	0.00	0.04	0.07	0.09	0.08	0.07	0.08	0.07	0.06
Throughput (Uph)	18647	18668	18680	18688	18692	18712	18719	18714	74909
	± 37	± 37	± 37	± 37	± 37	± 37	± 37	± 37	± 153

Table 4.5: Machine statistics based on the simulator's results for different discretizations.

CHAPTER 5

Comparison of job shop and flow line

One major challenge in this project was the collection of data. Since NXP has many flow lines and the machine settings of a flow line differ for every product, it was very hard to collect the right data. In order to overcome this problem, some rough estimates for the default values have been made. Table 5.1 lists the default values used in this chapter. Although some estimates might significantly differ from the actual values, this may not have a big impact on the comparison. After all, if a parameter value is inaccurately estimated, then it will have impact on both a job shop and a flow line. The only important value will be the Multi Plunger speed. If the speed of a Multi Plunger is decreased to a value lower than 35000 molds per hour, then the MP becomes the bottleneck in a production network. If that happens, the total line throughput is mainly determined by the Multi Plunger throughput and effects like blocking of an ADAT or starvation of a PHICOM will hardly have any influence on the total line throughput. In that case, the difference between a job shop and a flow line will be very small. If the Multi Plunger is not the bottleneck, it has little influence on the difference between an flow line and a job shop since there is only 1 Multi Plunger and it is located at the end of a production line. Obviously, if one doubts whether the results and conclusion still hold for more accurate input data, it is always possible to repeat an experiment.

For the actual comparison of a flow line with a job shop, several experiments have been designed. The main point of interest for these experiments are listed below. For all the flow lines that are simulated in this chapter, a discretization of $\frac{1}{2}$ times the smallest buffer capacity (1125 products) is used. The discussion of the results is quite extensive since not only the most important differences between a flow line and a job shop are highlighted, but this difference is also explained. Doing so will result in a better understanding of the main characteristics of both production concepts. At the end of every experiment the main findings are briefly summarized. For reasons of compactness, the detailed machines statistics are left out of the report. Since these detailed results are only interesting for NXP, these results will be handed over separately.

Experiment 1: Influence of product characteristics.
Experiment 2: Influence of order size.
Experiment 3: Influence of changeover time.
Experiment 4: Influence of number of machines.
Experiment 5: Influence of buffer size (flow line).
Experiment 6: Influence of routing algorithms (job shop).

Experiment 7: Influence of splitting algorithms (job shop).
Experiment 8: Using production data.

Parameter	standard value
ADAT	
Quantity	3
Buffer (in front of machine)	2250 products
Speed	28000 dies per hour
Product changeover time	10 minutes
Product and leadframe changeover time	20 minutes
Mean up- resp. downtime	249 resp. 78 seconds
Standard deviation up- resp. downtime	1097 resp. 202 seconds
PHICOM	
Quantity	4
Buffer (in front of machine)	5750 products
Speed (1 die, 2 wire setting)	48000 wires (resp 24000 products) per hour
Product changeover time	2 minutes
Product and leadframe changeover time	5 minutes
Mean up- resp. downtime	396 resp. 118 seconds
Standard deviation up- resp. downtime	1011 resp. 372 seconds
Multi Plunger	
Quantity	1
Buffer (in front of machine)	15000 products
Speed	72500 molds per hour
Periodic maintenance required every	8 hours
Periodic maintenance duration	30 minutes
Mean up- resp. downtime	572 resp. 86 seconds
Standard deviation up- resp. downtime	1201 resp. 376 seconds
Flow line specific	
Arbitrary machine set to indexing mode	5 minutes
Leadframe reel	double-tracked
Transportation time	0 seconds
Job shop specific	
Arbitrary machine set to indexing mode	5 minutes
Cassette dimensions	40x14x40 products
Leadframe reel	double-tracked
Transportation time	80 seconds

Table 5.1: Default parameter values for a SOT457 production network

5.1 Influence of product characteristics

The first experiment is related to the product characteristics. There are two reasons for this experiment being the most important and therefore first experiment. First of all, it is believed that the biggest difference between a flow line and a job shop can be attributed to the utilization of machines. Especially if an ADAT is indexing in a network where the die-bonding section is the bottleneck, valuable capacity is lost in a flow line whereas this would not occur in a job shop. The same reasoning holds for a PHICOM in a network where the wire-bond section is the bottleneck. Secondly, the results from this experiment may be useful for the following experiments as well (e.g. what products should be produced in order to have a balanced flow line and job shop). Furthermore one should know that changing the product characteristics is, in some cases (e.g. changing from 3 to 6 wires per product), similar to changing the machine speed. Hence, an experiment in which the machine speeds are altered is not included in this master's thesis.

Since the SOT457 package does not only contain the five products mentioned in Section 2.2, it may be interesting to investigate how well a production concept performs in producing only one type of product. There will be no changeovers, therefore product characteristics such as leadframe type do not matter. What does matter is the number of dies and wires per product. In Table 5.2 and Figure 5.1 the line throughput (defined as the Multi Plunger throughput) is given. One can immediately see that there is a huge difference in line throughput between the flow line and the job shop for the 1d1w, 1d2w, 2d2w and 2d4w products. The reason for that is that for these products the line throughput is mainly determined by the throughput of the die-bonders since the die-bond section is by far slower than the wire-bond section. Since only 2 ADATs are used in a flow line for products that require only one or two dies, one is actually comparing a flow line with 2 ADATs with a job shop with 3 ADATs. Therefore, the difference between line throughput can increase up to $\frac{3-2}{2} = 50\%$ (as can be seen in the 2d2w and 2d4w case). For the 1d4w products not the die-, but the wire-bond section is the bottleneck. Therefore the line throughput is mainly determined by the total PHICOM throughput. Since there will be no indexing PHICOMs in a flow line, the difference in throughput between a flow line and a job shop will not be as big as before. The difference of 13.7% can be attributed to less blocking (infinite buffers) and less starvation (better usage of ADAT capacity in job shop). For the 3d3w and 3d6w products the PHICOMs are the bottleneck as well and since every machine is used in a flow line for these products, there is no loss of capacity. Therefore the difference in line throughput is very small (roughly 3%). Since a flow line is well-balanced for a 3d6w product, this product will also be used in the other experiments in order to determine the difference between a balanced flow line and a job shop.

	1d1w	1d2w	1d4w	2d2w	2d4w	3d3w	3d6w
Flow line	42492 ±393	42681 ±412	34112 ±341	21712 ±113	21725 ±113	21328 ±135	19821 ±135
Job shop	60717 ±532	60622 ±538	38783 ±239	33409 ±386	33143 ±353	22020 ±250	20420 ±299
Difference	18226	17942	4671	11697	11419	692	599
Rel. diff.	42.9%	42.0%	13.7%	53.9%	52.6%	3.2%	3.0%

Table 5.2: Simulated line throughput for different product characteristics.

Summary

- Poor usage of machine (e.g., setting an ADAT to indexing) capacity results in a relative difference in line throughput of more than 50% (in favor of the job shop) for the 1d1w, 1d2w, 2d2w and 2d4w products.
- In a balanced flow line (3d6w products) without indexing machines a gain of only 3% can be achieved when one switches from flow line to job shop production.

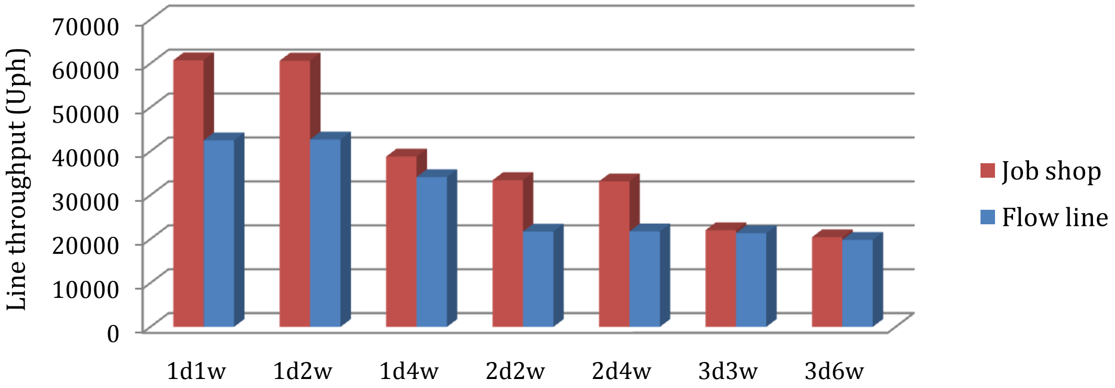


Figure 5.1: Simulated line throughput for different product characteristics.

5.2 Influence of order size

A typical difference between an obvious flow line package (SOT23) and a candidate job shop package (SOT457) is the product diversity. There are many changeovers for the SOT457 package and only few for the SOT23 package. Obviously, if an order is big, then the number of changeovers in a week is low. The relation between throughput and order size is investigated in this section.

For this experiment the simulator is given a fixed set of orders to assemble, this set of orders is a random permutation of product 1 up to 5 from Section 2.2 (see Appendix D). This order set is used the other experiments as well. Since it already became clear in the previous experiment that a job shop outperforms the flow line for product 1, 2, 3 (2d4w type) due to a better machine utilization, it may be interesting to investigate the relation between order size and line throughput for balanced networks. Therefore two separate comparisons have been made, one for comparing a flow line with a job shop in producing the regular products and one for comparing a flow line with a job shop when it is only producing 3d6w products. For the latter, the product characteristics of product 1 up to 5 have been changed such that every product requires 3 dies and 6 wires. Further characteristics such as leadframe or wire-bond type are left unchanged. Hence, the same changeovers are required for both cases. The results of varying the order size of every product from 20k up to 320k are given in Table 5.3 and Figures 5.2 and 5.3 (note that the scale of the x-axis is logarithmic). Especially for the small order sizes the job shop performs tremendously better than the flow line. Again, this is exactly what one expects. For instance, the total processing time of an order of size 20k and product type 1 is equal to $\frac{20000\cdot 2}{28000}+\frac{20000\cdot 4}{48000}+\frac{20000\cdot 1}{72500}=3.4h$. For that product the setup of 2 ADATs (the first ADAT is indexing) and 4 PHICOMs has to be changed which will take 28 up to 60 minutes (depending on whether a leadframe changeover is required). This changeover time may, in a flow line, also result in the blocking of machines upstream, or starvation of the machines downstream in the network. In a job shop this starvation will occur less frequently since the ADATs can be assembling different products. Hence, if ADAT 1 in a job shop is being setup to assemble product 1, then ADAT 2 may still be assembling a different product, thereby preventing a total starvation of the PHICOMs. Once the order gets bigger, the difference between a flow line and a job shop becomes smaller. The relative difference of 44.4% for an order size of 320k can mainly be attributed to the loss of ADAT capacity for products 1, 2 and 3 (as has been discussed in the previous section). The remaining part is the result of less starvation (infinite buffers, better routing) and less blocking (infinite buffers). For these big order sizes the blocking and starvation due to a changeover are very small. In the balanced flow line the relative difference in line throughput is only 6%. Although this difference is not as big as the differences before, it still is significant.

	20000	40000	80000	160000	320000
NXP products					
Flow line	14212 ±91.76	17646 ±66.94	20011 ±70.60	21511 ±378.90	22524 ±144.95
Job shop	27124 ±206.09	30410 ±283.06	31859 ±324.46	32032 ±285.64	32535 ±351.01
Difference	12912	12764	11848	10520	10011
Rel. diff	90.8%	72.3%	59.2%	48.9%	44.4%
3d6w products					
Flow line	11919 ±58.06	15105 ±46.15	17107 ±82.70	18335 ±156.17	18741 ±71.04
Job shop	18359 ±254.43	19828 ±188.39	19890 ±200.78	19821 ±172.15	19891 ±188.76
Difference	6440	4723	2783	1487	1150
Rel. diff	54.0%	31.3%	16.3%	8.1%	6.1%

Table 5.3: Simulated line throughput for different order sizes.

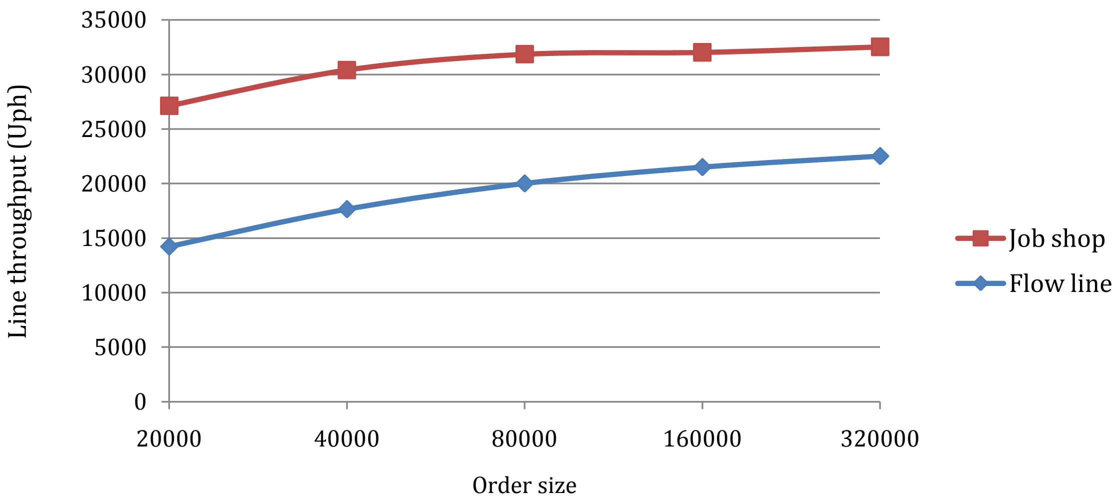


Figure 5.2: Simulated line throughput for different (logarithmical scaled) order sizes of NXP products.

Summary

- Small orders (< 50k) should always be produced on a job shop.
- A job shop is affected less by the order size than a flow line.

5.3 Influence of changeover time

A full buffer in a flow line can withstand a downtime of at most 5 minutes for an ADAT and 12 minutes for a PHICOM. Since a changeover on an ADAT takes 10 up to 20 minutes, such a changeover will often result in both a temporary starvation of a machine downstream and the blocking of a machine upstream in the network. Therefore, especially a flow line is, due to its finite buffers, very vulnerable to changeovers. As can be seen from the previous experiment, a job shop is less affected by capacity loss due to changeovers. In order to investigate the influence of the time lost due to machines

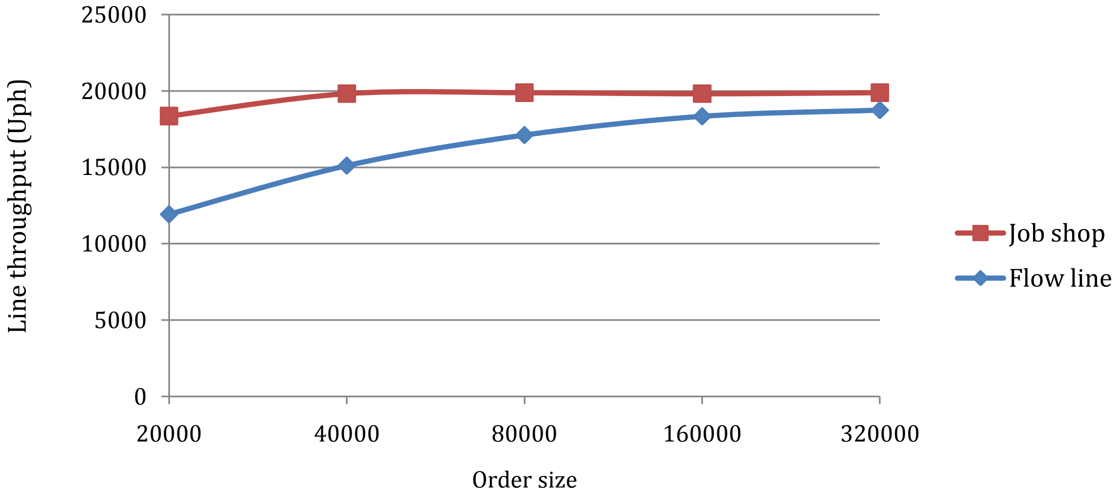


Figure 5.3: Simulated line throughput for different (logarithmical scaled) order sizes of 3d6w products only.

changeovers on machine throughput, the default changeover time will be scaled with a factor 0, $\frac{1}{2}$ and 2 in in this experiment.

The influence of the changeover time is investigated both in a network producing the regular NXP products and 3d6w products only (just like in the previous experiment). The same order set and product characteristics were used, however, the order size was set to 80k for all orders. The simulator's results are listed in Table 5.4. In this table one can once again see that the duration of a changeover is affecting the flow line more than the job shop. For the NXP products (resp. the 3d6w products) the relative difference between a flow line and a job shop increases from 44% to 77% (resp. from 3% to 39%) if one changes from no changeovers to double changeover times. The reason for this big difference is related to the blocking and starvation of a machine due to a changeover. When the duration of a changeover is scaled with a factor 2, a product plus leadframe changeover for an ADAT takes 40 minutes per machine. It can therefore take up to $3 \cdot 40 = 120$ minutes before the setup of all three ADATs is changed for a 3d6w product. In such a case it will take approximately 2 hours before the first new product arrives at the first PHICOM machine. The only products that can be processed by the first PHICOM in this period, are the products that are in the buffer in front of machines A2, A3 and P1, resulting in a total of at most $2250 \cdot 2 + 5750 = 10250$ products. Assuming that this PHICOM is only processing one track and placing 3 wires on a product, it will take $\frac{10250 \cdot 3}{48000} = 0.64h = 38$ minutes of uptime on a PHICOM before the combined buffer content is drained. Hence, starvation will definitely occur. To make it even worse, if the changeover of machine A2 takes 40 minutes, then machine A1 (placing 1 die per product per track) may already be blocked after $\frac{2250 \cdot 2}{28000} = 0.16h = 10$ minutes, resulting in a blocking period of 30 minutes. In a job shop the situation is completely different. Not only are the infinite buffers better able to withstand the fluctuations due to a changeover, the flexibility of the routing also results in less starvation of the machines. For example, if the setup of one ADAT is changed, then 2 other ADATs can provide the PHICOMs with enough work to prevent total line starvation.

Summary

- The line throughput in a job shop is less affected by the duration of a changeover.

		Scaling of the standard changeover time			
		0	0.5	1	2
NXP product					
Flow line	throughput	23545 ±392	21873 ±167	20011 ±71	16960 ±90
	setup fraction	0.05	0.03	0.02	0.00
Job shop	throughput	33847 ±575	32720 ±557	31894 ±576	30034 ±299
	setup fraction	0.05	0.03	0.01	0.00
Diff. in throughput		10302	10847	11884	13074
Rel. diff. in throughput		43.8%	49.6%	59.4%	77.1%
3d6w products					
Flow line	throughput	19276 ±309	18709 ±27	17107 ±83	14208 ±90
	setup fraction	0.05	0.03	0.02	0.00
Job shop	throughput	19784 ±280	19736 ±288	19888 ±305	19741 ±299
	setup fraction	0.05	0.02	0.01	0.00
Diff. in throughput		508	1028	2781	5533
Rel. diff. in throughput		2.6%	5.5%	16.3%	38.9%

Table 5.4: Simulated line throughput and fraction of the time that a machine is in changeover for different scaling factors of the default changeover times.

- When the changeover for a product takes a long time, that product should be produced on a job shop.
- The difference in line throughput between a balanced flow line (3d6w products) and a job shop without changeovers is small (< 3%).

5.4 Influence of number of machines

From the experiments in the previous sections it became clear that biggest difference (for the NXP products) between throughput in a flow line and a job shop can be attributed to a better usage of ADAT capacity. Four out of five products only require 1 or 2 dies, hence, the first ADAT will frequently be indexing in a flow line. This evidently brings up the question how well a production network performs when certain machines are removed, or machines are added to the network. Obviously, product 4 (3d3w) requires all three ADATs and prevents removing an ADAT. Therefore this product is excluded for the simulations in Section 5.4.1. In a job shop it is always possible to remove a machine as long as there is at least one ADAT, PHICOM and Multi Plunger. Therefore, product 4 is added again for the simulation in Section 5.4.2. In both sections the influence of the number of machines in a job shop network on the total line throughput is investigated.

5.4.1 NXP products except for product 4

Table 5.5 and Figure 5.4 display the simulated line throughput for different network configurations. The most apparent difference between a job shop and flow line is well visualized in the chart: changing from a 2A4P1MP network to a 3A4P1MP network makes a big difference in a job shop, whereas there is no difference at all between the two flow line networks. Again, this corresponds with the expectations. In a 2A4P1MP network the die-bond section is, by far, the bottleneck. Hence, if one goes from two to three ADATs in a job shop, one can expect an increase in line throughput up to 50% (simulation shows that the difference is only 40%). However, in a flow line the addition of the ADAT is useless since the additional ADAT will always be indexing. Hence, there will be no gain in line throughput at all. The opposite holds when one adds an ADAT to a 3A4P1MP network. In that case the throughput of a flow line increases with roughly 30% because now all four ADATs can be used for product 1, 2 and 3 (2d4w) (due to the double-tracked leadframe). On the other hand, adding an ADAT in a 3A4P1MP job shop does not increase the throughput that much. This is mainly because in such a job shop the total wire-bond capacity is approximately equal to the total die-bond capacity for the selected products. Hence, adding an ADAT without adding a PHICOM does not have a big impact in a job shop. If one divides the line throughput by the total number of machines in a network, one can determine the optimal number of machines in a network. In a flow line an optimal network would, according to this rule, be a 2A4P1MP network, however, such a network is infeasible when one wants to include product 4 as well. Therefore, a better alternative would be a 4A4P1MP network. For a job shop an ideal network would consist of 4 ADATs, 5 PHICOMs and 1 MP. Note that in practice NXP has many production lines, so instead of converting say 5 separate flow lines to 5 separate job shops, one could as well create one huge job shop. It is expected that one huge job shop performs slightly better then 5 separate job shop. However, the difference will be small and may not outweigh the challenges arising due to the increase of network and routing complexity.

	2A2P1MP	2A3P1MP	2A4P1MP	3A4P1MP	4A4P1MP	4A5P1MP	4A6P1MP
Flow line	16606	18716	21480	21416	27521	28211	28204
	±75	±96	±100	±142	±167	±95	±185
Job shop	17415	21332	24429	34540	34520	42661	48444
	±98	±103	±176	±451	±272	±815	±1026
Difference	809	2617	2949	13125	6999	14450	20241
Rel. diff.	4.9%	14.0%	13.7%	61.3%	25.4%	51.2%	71.8%

Table 5.5: Simulated line throughput for different network configurations.

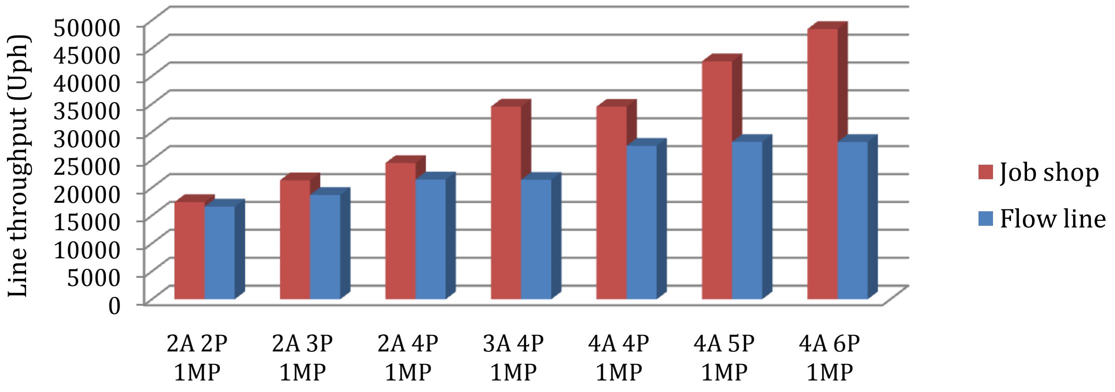


Figure 5.4: Simulated line throughput for different network configurations.

Summary

- In a job shop one can really profit from adding a machine to the bottleneck section, this is not necessarily true in a flow line.
- The optimal flow line for the NXP products excluding product 4 would have 2 ADATs, 4 PHICOMs and 1 Multi Plunger.

- The optimal job shop for the NXP products excluding product 4 would have 4 ADATs, 5 PHICOMs and 1 Multi Plunger.

5.4.2 All NXP products (job shop only)

This sub-experiment focuses on the influence of the number of machines in a job shop network on the total line throughput. Since there are no product restrictions in a job shop, product 4 will once again be added to the collection of products. Since for most NXP products it is not advised to use more than 4 ADATs in a network, the flow line will be ignored in this sub-experiment. The results for varying the number of machines in a job shop are given in Table 5.6 and Figure 5.5. The weighted throughput in this table is calculated by dividing the line throughput by the total number of machines in the network, the die-bond, wire-bond and mold-load will be explained later. From the line throughput data in the table it immediately becomes clear that the optimal configuration of a job shop would be a network with 5 ADATs, 6 PHICOMs and 1 MP. The reason why this network is preferred can be explained as well. The network configuration resulting in the highest weighted throughput is the network in which the die-bond, wire-bond and molding section are as balanced as possible. Intuitively, this makes sense since balanced sections imply that there is no bottleneck. Hence, all machines are used at full capacity. Based on the product characteristics, average machine up and downtime, and the number and speed of the machines, one can calculate the relative workload for every section (excluding capacity loss due to blocking, starvation and changeovers). A relative workload of 100% implies that that section is the bottleneck. These calculations have been performed for all the network configurations and are also listed in Table 5.6. It is therefore not surprising that the biggest difference in section workload is only 5% for a job shop with 5 ADATs, 6 PHICOMs and 1 MP.

	2A2P1MP	2A3P1MP	3A3P1MP	3A4P1MP	4A4P1MP	4A5P1MP
Throughput	17916 ±74	21310 ±105	28822 ±242	31920 ±303	38644 ±219	42571 ±559
Weighted throughput	3583	3552	4117	3990	4294	4257
die-bond load	83.4%	100.0%	83.4%	100.0%	83.4%	100.0%
wire-bond load	100.0%	83.8%	100.0%	92.8%	100.0%	98.3%
mold load	31.9%	39.7%	47.8%	58.7%	63.8%	77.8%

	5A4P1MP	5A5P1MP	5A6P1MP	6A6P1MP	6A7P1MP
Throughput	38274 ±439	47709 ±843	52918 ±1042	56929 ±1189	60189 ±1151
Weighted throughput	3827	4337	4410	4379	4299
die-bond load	68.8%	83.4%	98.0%	83.4%	86.6%
wire-bond load	100.0%	100.0%	100.0%	100.0%	89.9%
mold load	64.4%	79.7%	95.1%	95.7%	100.0%

Table 5.6: Simulated line throughput and section load for different number of machines in a job shop, the line throughput is weighted against the total number of machines in a network.

Summary

- A job shop offers great flexibility when it comes to adding and removing machines.
- The optimal job shop for the NXP products would have 5 ADATs, 6 PHICOMs and 1 Multi Plunger.

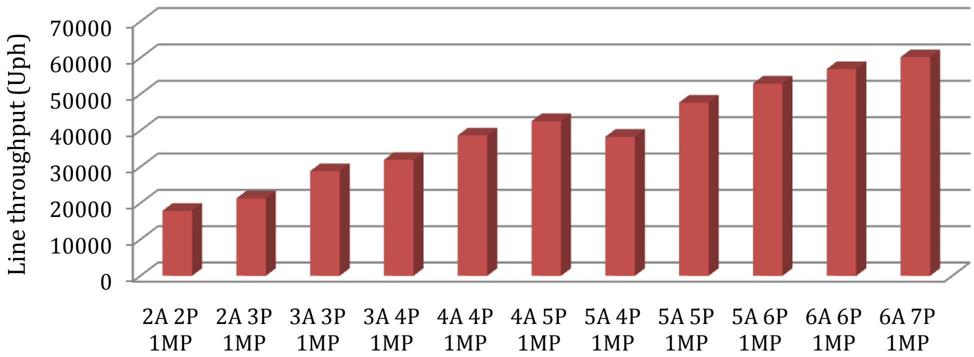


Figure 5.5: Simulated line throughput for different number of machines in a job shop.

5.5 Influence of buffer size

A comparison between a flow line and a job shop is not always fair. For example, a job shop has infinite buffers and a flow line has only small finite buffers. Note that in practice a buffer capacity of approximately 10 cassettes is more than enough for a regular job shop. This experiment is used to determine the gain in throughput when one could switch from regular finite buffers to infinite buffers in a flow line. The results are given in Table 5.7. As can been seen, when a flow line with infinite buffers is fully used (3d6w products), then the difference in throughput between a flow line and a job shop is very small (less than 2%). Especially in these balanced production environments large buffers can make a lot of difference. Unfortunately, it might be hard to increase the buffer size in a flow line. Solutions are more likely to be found in the field of mechanical engineering (e.g. increasing the height of a buffer) than in the field of mathematics. Furthermore, a deep investigation towards the buffer capacities has already been performed in the first project. Hence, this master's thesis does not contain an extensive investigation of the influence of the buffer capacities. The main reason for this experiment is to determine how important the infinite buffers in a job shop are when that production concept is compared with a flow line.

In practice the buffers in a job shop cannot be infinite, there is always some sort of limitation. More important, a situation in which there are at least 20 cassettes located at every machine is undesirable. In general, such a situation will not occur when the die-bond section is the bottleneck (recall that a new order is only released when an ADAT is free). If, on the contrary, the wire-bond or molding section is the bottleneck, then another order-release strategy has to be used to prevent high WIP-levels. A very straightforward policy would be not to release a new order when there are more than *C* cassettes in a production network. This strategy has been implemented in the simulator, but no experiments have been performed using this straightforward policy. The main reason is that it is believed that a reasonable boundary will not affect the line throughput at all. After all, if the ADATs are the slowest machines, then there already is an order release policy. If not, then either the wire-bond or the molding section is the bottleneck. Anyhow, the total line throughput is mainly determined by the throughput of the bottleneck machines. If there are many cassettes in a system, then they are very likely to be located at the bottleneck machines. Hence, starvation of a bottleneck will not occur and therefore total line throughput will not decrease.

Summary

- Infinite buffers result in an increase of line throughput of roughly 15%.

		NXP products	Rel. diff.	3D6W products	Rel. diff.
Flow line	regular buffer	20011 ± 71	-	17107 ± 83	-
	infinite buffer	22999 ± 133	15%	19585 ± 243	14%
Jobs shop	infinite buffer	31894 ± 576	59%	19888 ± 305	16%

Table 5.7: Simulated line throughput for different production concepts and buffer sizes.

- In a balanced flow line with infinite buffers the throughput is roughly the same as the throughput in a job shop.

5.6 Influence of routing algorithms (job shop)

The infinite buffers in a job shop are not the only advantage, another advantage is the flexibel routing. Since the main goal of this project was to get insight in the main characteristics of a flow line compared with a job shop, only straightforward routing algorithms have been implemented. The simulated line throughput for different routing algorithms is listed in Table 5.8. Since the line throughput for the clever algorithms (shortest queue and smallest changeover) only slightly differs from the random algorithm, one may conclude that the choice for the routing algorithm in a job shop does not matter a lot. However, one should bear in mind that all three algorithms give absolute priority to an empty machine. It is due to this restriction that the difference between the routing algorithms is very small. The main conclusion from this experiment is that NXP should, for now, not bother on spending too much time on improving the routing algorithms. The straightforward routing algorithms should, especially in the beginning, be sufficient.

Algorithm	Line throughput
Random	31757 ± 621
Shortest queue	31936 ± 327
Shortest changeover	32277 ± 439

Table 5.8: Simulated line throughput for different routing algorithms in a job shop.

Summary

- There is no significant difference in line throughput for the different routing algorithms.
- The most important characteristic of the routing algorithms is giving absolute priority to feasible free machines.

5.7 Influence of splitting algorithms (job shop)

In total two batch split algorithms have been implemented. Although the idea behind the smartSplit algorithm seems promising, the actual performance of this algorithm is very disappointing. When one uses the smartSplit algorithm, one will in practice always split an order into exactly 3 sub-batches (note that this is also the maximum number of sub-batches for the smartSplit algorithm). The reason for that is that the changeover time at a PHICOM is less than 5 minutes. Hence, the gain of splitting an order is very high for the wire-bond section. In fact, this gain is so high that one has to scale the changeover times with a factor 10 (using input parameter q)) in order to prevent the splitting of an order. Hence, it is advised not to use the smartSplit algorithm and no results of this algorithm will be included in this report.

The only alternative for the smartSplit algorithm is the fixedSplit algorithm. Luckily, the performance of that algorithm is not disappointing at all. Table 5.9 lists the results for different input parameters for the fixedSplit algorithm. The input for the simulation was the standard set of NXP product orders, each of size 80k resulting in 4 cassettes per order. Therefore, it makes no difference if the maximum number of cassettes in a sub-batch is 2 or 3. In both cases an order will be split in 2 sub-batches and since the number of products in the sub-batches are balanced, both sub-batches contain the same number of products. The most important conclusion that can be drawn from Table 5.9 is that the wormhole routing principle from Section 2.2.3 is really making a difference. For instance, when the orders are not split in separate sub-batches (fixedSplit input 4) and all cassettes are kept together instead, the average fraction of time lost due to changeovers decreases and the line throughput increases significantly. Hence, if NXP decides to switch from flow line to job shop production, then the wormhole routing principle should really be used.

	Maximum number of cassettes per sub-batch			
	1	2	3	4
Line throughput	27401 ± 73	30560 ± 180	30607 ± 135	31936 ± 327
Average setup fraction	0.08	0.04	0.04	0.03

Table 5.9: Simulated line throughput for different number of cassettes per sub-batch in a job shop.

Summary

- The performance of the smartSplit algorithm is disappointing. It is advised not to use this algorithm. One should use the fixedSplit algorithm instead.
- The wormhole routing principle does what it is supposed to do: limit the number of changeovers and therefore increase line throughput.

5.8 Using production data

In Section 4.3.1 the developed simulator has been validated by comparing the simulated flow line results with production date from line 48. A job shop is given exactly the same input that was used for the validation, the results are listed in Table 5.10. Recall that the throughput based on the empirical production data was 19361, the simulator estimated a throughput of 19300 units per hour. As can been seen, the simulated throughput for a job shop is equal to 19875 ± 168, yielding a relative difference of only 3%. That difference is rather small. However, that does not mean that the results from previous experiments are incorrect. In fact, the input data for the SOT23 and SOT457 line are very different. For example, the machines in the SOT23 line are down for about 5% of the time, in the SOT457 line the machines are down for at least 17% of the time. That is a huge difference and since more downtime means more blockage and starvation, it is not surprising that the difference between a flow line and a job shop are that big for the SOT457 production line and only small for this SOT23 production line.

Summary

- In a balanced flow line with reliable machines and no changeovers, the relative difference in line throughput is 3%.

	A1	A2	A3	A4	P1	P2	P3	P4	MP1
Utilization	0.86	0.86	0.88	0.88	0.87	0.85	0.86	0.86	0.89
	±0.07	±0.07	±0.07	±0.07	±0.06	±0.06	±0.06	±0.06	±0.00
Starved fraction	0.09	0.09	0.08	0.08	0.08	0.08	0.08	0.08	0.00
	±0.07	±0.07	±0.07	±0.07	±0.06	±0.07	±0.06	±0.06	±0.00
Down fraction	0.05	0.05	0.04	0.03	0.05	0.07	0.06	0.06	0.11
	±0.00	±0.00	±0.00	±0.00	±0.00	±0.00	±0.00	±0.00	±0.00
Setup fraction	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
	±0.00	±0.00	±0.00	±0.00	±0.00	±0.00	±0.00	±0.00	±0.00
Throughput	19651	19693	20028	20070	20036	19660	19911	19869	19875
	±1533	±1545	±1546	±1534	±1352	±1394	±1386	±1377	±168

Table 5.10: Simulated line throughput for a job shop producing SOT23 products.

CHAPTER 6

Conclusion

The main conclusion that can be drawn from all the experiments in the previous chapter is that a job shop assembly line results in a better line throughput than a flow line. The relative difference between line throughput in a job shop and a flow line is small (less than 5%) when

- no machines are set to indexing for a significant period of time,
- changeovers are not important (either because orders are huge, or because a changeover does not take a lot of time),
- the duration (and frequency) of a downtime is small,
- there is no obvious bottleneck section, that is, the die bond, wirebond and molding section are balanced.

In practice, these criterions are highly related to each other. For the NXP products in Section 2.2 the first and last criterion does not hold, resulting in a relative difference in throughput of 60% (in favor of the job shop) for an average order size of 80k. In a balanced flow line without indexing machines the relative difference decreases to 16% for the same average order size. The gain in line throughput in a job shop can be attributed to the following factors:

- Better machine utilization due to flexible routing and flexible network configurations.
- No more blocking of machines due to the infinite buffers.
- Less starvation due to a better and more flexible routing (a machine can receive input from several machines instead of only one).
- Possibly less changeovers since all wires of the same diameter are placed by one PHICOM.

Since there even is a significant difference in line throughput between a balanced flow line and a job shop, it is strongly advised to further explore the possibilities of changing a rigid flow line to a flexible job shop. Once one has more experience in changing a flow line to a job shop and one knows how to maintain a job shop, it might even be worth changing more production lines for other product packages to a job shop configuration as well. If NXP decides to change a flow line to a job shop, one should really use either the simulator or the results from Chapter 5 to determine an optimal number of machines. For example, it might be that a machine is redundant in a job shop although it actually is required in a flow line. To conclude, there are only few reasons not to change from flow line to job shop production (two reasons being high research costs and the high level of manual labor in a job shop). However, the gain in line throughput can be really worth the effort, but it is further left to the management of NXP to balance the increase in line throughput against the costs.

It should furthermore be mentioned that the result and final outcome of this project is not limited to the tables in Chapter 5. Most important, NXP can now use a very flexible simulator to investigate the effect of merely any parameter, both in a flow line and in a job shop. Furthermore, since the simulator has been developed in an object oriented way and is provided with proper inline comments, NXP can easily extend this simulator when needed. For example, if NXP wants to investigate the effect of certain routing algorithm, only this specific routing algorithm has to be added to the code. In addition, the simulator has been built up from scratch and many small details from a NXP production line have been implemented (e.g., slightly different PHICOM speeds for different production settings, the possibility to add a transportation time). In brief, the simulator can become a very powerful tool for NXP Semiconductors for any possible analysis. Especially the possibility to visualize the production is a very powerful tool to get insight in the way a production networks responds to small disturbances.

CHAPTER 7

Recommendations for future research

It is a common phenomenon that answering one question leads to ten other questions. The same holds for this project, in the search for answers some other questions arose that could not be answered. Either because they were beyond the scope of this project, or because they are better to be answered by an NXP flow line expert. This chapter does not only give a recommendation for future research, parts of the solution are also given. At the end of this chapter all questions are summarized.

The first two questions both relate to setting a machine to indexing. It has been stated before that there may be a difference between setting the first or the last ADAT to indexing. After all, if the first ADAT is set to indexing then the buffer between ADAT1 and ADAT2 is not used. However, if the last (third) ADAT is set to indexing, then the buffer will be used and the buffer in front of the first PHICOM is practically enlarged by the buffer in front of the first ADAT. Bigger buffers usually imply a higher line throughput, but it is not known how big the difference will be in practice. Actually, it may not be necessary at all to set a machine to indexing. In fact, instead of having one ADAT placing two dies (one per track) and one ADAT indexing, it is also possible to have two ADATs each placing one die. In a flow line the latter may result in a slightly higher line throughput since starvation of the third ADAT is less likely. However, the first and second ADAT will be blocked more often and that may result in a lower product quality. Again, theory favors two separate machines each placing one die, but the gain may, in practice, be very small.

Two other research possibilities are related to the routing and scheduling in a job shop. In Section 5.6 it was shown that there is no significant difference between the three implemented routing algorithms. However, that does not mean that improving the routing algorithms is not worth the effort. On contrary, in a big job shop it is expected that a good routing algorithm can save a lot of time, especially when it comes to order sojourn time. One possible (advanced) improvement would be the usage of *circular routing* as an extension of wormhole routing. The main difference is that the dies and wires are no longer placed in a fixed order. Hence, in the time that the head of a sub-batch is being processed at machine 1, the tail can be processed at machine 2. After a service completion the head will go to machine 2 and the tail will go to machine 1. Although this circular routing may not increase the line throughput, it will definitely decrease order sojourn time and WIP-levels. Another possible improvement is related to the planning of jobs in a job shop. In a flow line the orders are pre-clustered in order to decrease both the number and duration of changeovers. In a job shop this pre-clustering may be less important and using dynamical scheduling heuristics may result in a higher line throughput.

It is not only the scheduling flexibility in a job shop that may provide an improvement. The line configuration is important as well. Although this topic has partly been discussed in Section 5.4.2, it is not evident from the results in that section that it is better to have one big job shop instead of two small ones. Again, theory would favor one big job shop since a big job shop offers more flexibility. Unfortunately, the routing and planning does become more difficult in a big job shop and a mathematician should not be asked to decide whether the small gain in throughput is worth the increase of network complexity.

The results from this master's thesis are evident: a job shop will always perform better than a flow line. However, it might be possible to have best of both worlds. That is, (part of) the flexibility and high machine utilization of a job shop combined with the reel-to-reel production of a flow line. This can be achieved if one decouples the die-bond, wire-bond and molding section. This results in a job shop of mini flow lines. Unfortunately, such a "job line" network is totally different from the single production networks that were used in this project. Hence, no attention could have been given to this idea in this project. However, the simulator that has been developed in this project can be changed in order to replicate such a hybrid form network. Two possibilities will be explained. First of all, it is possible to implement a new and sophisticated routing algorithm. Together with some minor changes in the source code of the simulator this should be sufficient to investigate the performance of a "job-line" network. Another possibility would be to use a mathematical algorithm to aggregate the machines in one mini-flow line into one single machine. The characteristics of these aggregated machines can then be used in a regular job shop in order to replicate a "job-line". Although the aggregation of machines in a fluid model is very challenging and interesting from a mathematical point of view, it unfortunately could not have been included in this research project. Hence, it will be let to NXP to further explore the possibilities of a hybrid form production network.

Last but not least there is a research question that came hand in hand with the discretization of a flow line. Although it may not be interesting for NXP Semiconductors, from a mathematical point of view it is very interesting to investigate how well a general discrete model can approximate statistics such as line throughput or order sojourn time of a continuous model. The opposite is obviously interesting as well. That is, can a continuous model be used to approximate statistics such as line throughput for a discrete model?

Summary

1. Is there a difference between setting the first or last ADAT to indexing?
2. Is it better to let one machine place two dies (one per track) and one machine indexing, or should both machines be placing one die only? More general, how should one divide work in a flow line in order to increase line throughput.
3. Is circular routing a good extension for wormhole routing?
4. What, possibly dynamic, scheduling heuristics could be used in order to increase the throughput in a job shop?
5. What is better, two small job shops or one big job shop.
6. Is it possible to combine the flexibility of a job shop with the standard ways of working of a flow line?
7. How well can a continuous model approximate statistics such as line throughput or order sojourn time for a discrete model (and vice versa)?

APPENDIX A

Glossary

1d2w	A product consisting of 1 die and 2 wires. Other combinations are possible as well.
A$_i$	The i$^{\text{th}}$ ADAT in a production network.
ADAT	A machine that places a die on a leadframe.
AWACS	Advanced Warning and data Collection System, a collector, used by NXP, for machine related production data such as uptime duration, downtime duration, products produced, error type.
Batch	See Order.
Blocked	A machine is blocked when it cannot store its recently processed product at the buffer of another machine.
Bottleneck	A machine(type) is the bottleneck in a production network if the total throughput of the network is limited by the performance of that specific machine(type).
c_X	Coefficient of variation for random variable X, $c_X^2 = \frac{\text{Var}(X)}{\text{E}(X)^2}$.
Cartridge	See Cassette.
Cassette	A box for storing the leadframe strips in a job shop. In a flow line a cassette is just a chunk of leadframe of certain length.
Changeover	See setup.
Cross index	If a machine has one cross index it can perform tasks on two tracks instead of only one track.
Die	A small block of semiconductor material on a wafer.
Continuous flow line	A flow line in which machine process items at a certain speed and buffers are filled at a certain rate dependent on the state of the network. The term continuous is used since there typically is a continuous output from the network.
Discretized flow line	A flow line in which chunks of leadframe are routed through the network. The discretized flow line is used to approximate the continuous flow line, the term discretized is used since product will typically be produced in bulks.
ECDF	Empirical cumulative distribution function.
FCFS	First come, first served policy, orders are processed in order of arrival.

Flow line A production line in which an order has to visit every machine exactly once in a fixed order.

GUI Graphical user interface

Highrunner A product that is usually produced in high volumes.

IC Integrated circuit, also called a (micro)chip.

Indexing If a product leadframe only passes through a machine without being processed, that machine is said to be indexing.

Job shop A production network with no fixed routing for orders.

k Suffix for 1000.

Leadframe The product carrier for the final product, there are 9 different type of leadframes for the SOT457 package.

Magazine See Cassette.

Maintenance Some machines require a periodic maintenance, e.g. the molds in the Multi Plunger need to be cleaned every once in a while.

MATLAB A numerical computing environment and programming language developed by MathWorks. In this project MATLAB 7.11.0 R2010b has been used.

MP$_i$ The i$^{\text{th}}$ Multi Plunger in a production network.

Multi Plunger A machine that places a mold over every single product.

Network (production) A set of machines, raw orders enter the network as a result from customer demand and leave the network as finished products.

NXP NXP Semiconductors, a company fabricating semiconductor devices.

Order All identical products belonging to one customer requirement.

OOWI The Mathematics for Industry post master program from Eindhoven University of Technology.

Package A collection or family of NXP-products, e.g. SOT457 is a package.

PHICOM A machine that connects a die to the leadframe by soldering a gold wire to both the die and leadframe.

P$_i$ The i$^{\text{th}}$ PHICOM in a production network.

Product One specific diode or transistor produced, e.g. IP4220CZ6 is a product within the SOT457 package.

Production run The time between two different consecutive different machine setups.

ρ The occupation rate of a machine. Usually this is equal to the fraction of time that a machine is working.

Reel to reel The way products are produced in a flow line. At the first machine a big reel of multi-tracked leadframe is unwinded and after the last machine it is winded up again.

Reverse (single, double) For some products a special loop is required for the goldwire, NXP refers to these loops with single or double reverse.

Semiconductor (device) A component of an electronic circuit made from a semiconductor material.

Semiconductor (material) A material that is neither a conductor, nor an insulator. Semiconductor material is the base of semiconductor device.

Service Processing a product on a machine.

Setup The process of changing the setup of a machine.

Sojourn time The time an object (order, sub-batch, cassette, product) stays in a production network.

SOT457, SOT23 One of the many packages from NXP Semiconductors.

Starved A machine is starved when there are no products in the buffer to process.

Sub-batch A part of an order, in a job shop an order is split in several sub-batches each having a possibly different routing through the network.

Throughput (machine) The average number of products processed by one machine within a certain period of time; NXP usually measures throughput in 1000's of products per hour (kUph).

Throughput (network) The average number of products produced within a certain period of time. If there is only one Multi Plunger in the production network, then the network throughput is defined as the throughput of the Multi Plunger. NXP usually measures throughput in 1000's of units per hour (kUpH).

Track A leadframe in a flow line may consist of several tracks, if a leadframe consists of 2 tracks then every column of leadframe eventually yields 2 diodes.

TU/e Eindhoven University of Technology.

Uph Abbreviation for 1000 units per hour.

Wafer A thin slice of semiconductor material used as a base for the fabrication of integrated circuits.

WIP(-level) Work in progress (level), the remaining work that has to be done for all the orders in the network.

Wormhole routing A routing principle for a job shop: the first cassette of a sub-batch determines the routing of all other cassettes from that sub-batch and once a machine starts processing the first cassette, it is only allowed to process cassettes from that sub-batch and other cassettes have to wait.

APPENDIX B

Input for simulator

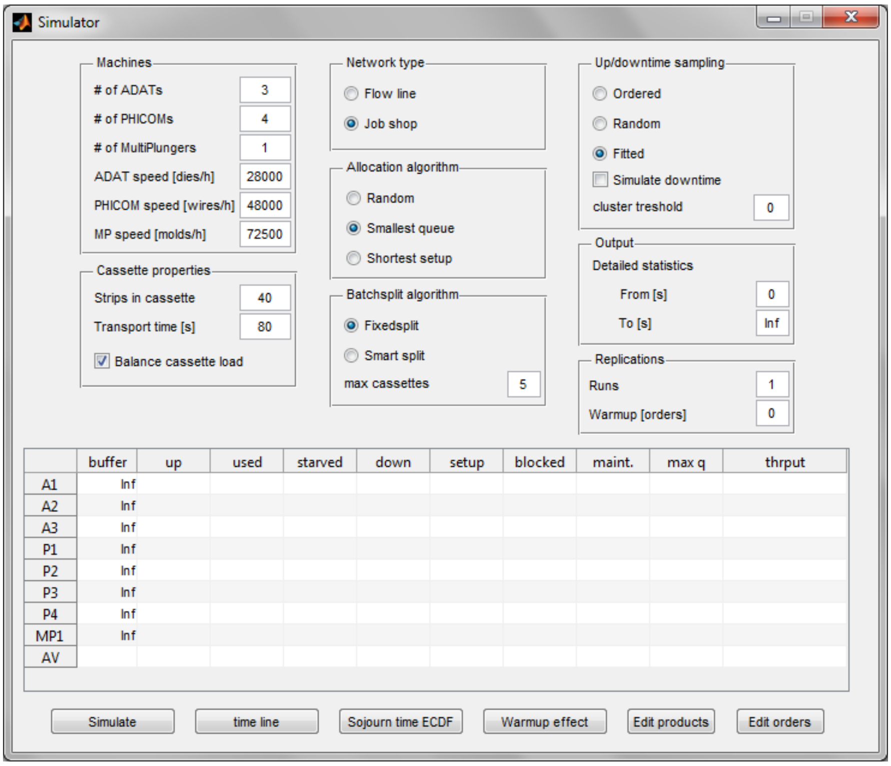
The simulator consists of a library of MATLAB functions that perform the relevant tasks. There are two ways to acces this library, one can use command-line direct input, the other method uses the a graphical user interface (GUI). This chapter will briefly discuss the different input types for the simulator. A general rule of thumb is that if one does not know what the influence is of changing an input parameter, then one can better not change the input parameter.

B.1 Graphical user interface

A GUI has been developed for the most common input, a screenshot of this GUI is depicted in Figure B.1. Since the name of the fields clearly explain the input parameters, only the possibly unclear input fields will be discussed in this section. The first uncommon field is the "strips per cassettes" (in case of a job shop) or "products per chunk" field with a default value of 40. When one wants to simulate a job shop then this field determines the maximum number of strips in a cassette. In case of a flow line this field determines the level of discretization (see section 3.1 for more information on the discretized flow line); a good value for this field would be $\frac{1}{4}, \frac{1}{2}$ or 1 times the smallest buffer capacity. The second field that requires some attention is the buffer capacity input. The user should be aware of the fact the buffer capacity is always given in number of products. Hence, if one wants to simulate a job shop with finite buffers, then one should remember that the buffer capacity is interpret as number of products and not as number of cassettes. Last but not least is the "runs" and "warmup" field, the "runs" field determines the number of simulation runs: if, for example, a value of 5 is supplied, then the order list will five times be fed to the network. The "warmup" field determines the number of orders that should be ignored. If, for example, a value of 100 is supplied, then the both the machine and the order statistics for the first 100 orders will be removed from the eventually displayed machine and order statistics.

B.2 Plain text input

There is some input that either does not need to be changed very often, or is too big to include in the GUI. All these files are collected in the Input folder of the simulator and will separately be listed in this section.



Simulator

Machines

# of ADATs: 3
# of PHICOMs: 4
# of MultiPlungers: 1
ADAT speed [dies/h]: 28000
PHICOM speed [wires/h]: 48000
MP speed [molds/h]: 72500

Cassette properties

Strips in cassette: 40
Transport time [s]: 80
☑ Balance cassette load

Network type

○ Flow line
● Job shop

Allocation algorithm

○ Random
● Smallest queue
○ Shortest setup

Batchsplit algorithm

● Fixedsplit
○ Smart split
max cassettes: 5

Up/downtime sampling

○ Ordered
○ Random
● Fitted
□ Simulate downtime
cluster treshold: 0

Output

Detailed statistics
From [s]: 0
To [s]: Inf

Replications

Runs: 1
Warmup [orders]: 0

	buffer	up	used	starved	down	setup	blocked	maint.	max q	thrput
A1	Inf									
A2	Inf									
A3	Inf									
P1	Inf									
P2	Inf									
P3	Inf									
P4	Inf									
MP1	Inf									
AV										

Figure B.1: Screenshot of the GUI for the simulator.

wire bondspeed.txt
This file contains the regular wirebond speeds for different setups. In fact, this file contains the same information as Table 2.4.

uptimeXi.txt, downtimesXi.txt
This file contains the uptimes or downtimes, measured in seconds, for the ith X machine, e.g. `uptimesADAT2.txt`.

orders.txt
This file contains the orders that will be fed to the network. A brief instruction for adding or removing orders is given in the file. This file can be edited using a text editor such a Notepad, or by clicking the orders button the in GUI.

products.txt
This file contains the product information. A brief instruction for adding or removing product is given in the file. This file can be edited using a text editor such a Notepad, or by clicking the product button the in GUI. Note that the wiretype mentioned in this file refers to the special loops that might be required (i.e., normal, single or double reverse).

B.3 Hidden input

Besides the GUI and the plain text input, the user can only change some constant parameters in the "Parameters.m" file which is located in the "Source" folder. The different parameters that can be changed are listed in the table below.

Parameter name	default	description
STRIPLENGTH	40	the length of a strip in a job shop
STRIPWIDTH	14	the width of a strip in a job shop
NOFTRACKS	2	the number of leadframe tracks in a flow line
ADATPRODUCTCHANGEOVER	10*60	the duration of a product changeover at an ADAT
ADATLEADFRAMECHANGEOVER	20*60	the duration of a product plus leadframe changeover at an ADAT
PHICOMPRODUCTCHANGEOVER	2*60	the duration of a product changeover at an PHICOM
PHICOMLEADFRAMECHANGEOVER	5*60	the duration of a product plus leadframe changeover at an PHICOM
INDEXINGCHANGEOVER	5*60	the time it takes to set a machine to indexing mode
MPMAINTENANCEDURATION	30*60	the time it takes to clean a Multi Plunger for the periodic maintenance
MPMAINTENANCEREQUIRES	8*60*60	the time between two consecutive periodic maintenances
INDEXINGSPEED	Inf	the speed at which a machine can index items
ARRAYLENGTH	2ˆ12	the number of cells allocated for storing detailed machine statistics for the timeline; only change this value if one really understand the simulators source code
ALPHA	0.05	used for determining a 1-ALPHA confidence interval for machine or order statistics
THROUGHPUTMOVINGAVERAGEWIDTH	10	the window width of a moving average filter for displaying the throughput per machine; only change this value if one really understand the simulators source code
SOJOURNTIMEMOVINGAVERAGEWIDTH	500	the window width of a moving average filter for displaying the cassette sojourn time per machine; only change this value if one really understand the simulators source code
EXPONENTIALSERVICETIMES	FALSE	if true, then the service times are no longer deterministic but exponential (with the same mean)
POISSONARRIVALS	FALSE	if true, then orders arrive according to a Poisson process; if false, then all orders are available at the start of a simulation.
ARRIVALRATE	50	the order arrival rate, only used when the previous parameter is set to true.
PREFERFREEMACHINES	TRUE	should the simulator give absolute priority to free machines or not when a new cassette has to be allocated in a job shop
MACHINEFAILURES	TRUE	if true, then machines break down once in a while; if false, then machines are 100% reliable and do not break down

Parameter name	default	description
CONSTANTPHICOMSPEED	FALSE	if true, then a speed penalty might be given for different PHICOM settings (e.g. placing in total two wires on two dies takes slightly longer then placing two wires on one die); if false, then no speed penalty is given
MAXCASSETTESSINJOBSHOP	Inf	the maximum number of cassettes allowed in a job shop, if there are more cassettes in a job shop then no new sub-batches will be released.
PREVENTIVEWARMUP	FALSE	this feature is experimental, do not change this parameter.

APPENDIX C

Validation fluid model

The sub-scenarios below will be tested in this appendix. Note that the relative error in the tables below is based on comparing the simulated throughput for a continuous and a discretized flow line.

Sub-scenario a: small network, balanced machines, infinite buffers.
Sub-scenario b: small network, balanced machines, finite buffers.
Sub-scenario c: small network, fast machines upstream, finite buffers.
Sub-scenario d: small network, fast machines downstream, finite buffers.
Sub-scenario e: medium sized network, balanced machines, finite buffers.
Sub-scenario f: medium sized network, fast machines upstream, finite buffers.
Sub-scenario g: medium sized network, fast machines downstream, finite buffers.

Sub-scenario a

Parameter name	value
Number of machines	1 ADAT, 1 PHICOM
Buffer sizes	b_1, b_2 infinite
ADAT speed (v_1)	25000 dies per hour
PHICOM speed (v_2)	25000 wires per hour
Uptime machine 1, 2	1033 s
Standard deviation uptime machine 1, 2	1033 s
Mean downtime machine 1, 2	243 s
Standard deviation downtime machine 1, 2	243 s
Replications	100 x 100 orders

Discretization	line throughput (kUph)	relative error
0 (continuous simulator)	20240 ± 57	-
1500	20197 ± 162	-0.2%
3000	20089 ± 124	-0.7%

Sub-scenario b

Parameter name	value
Number of machines	1 ADAT, 1 PHICOM, 1Multi Plunger
Buffer sizes	b_1 infinite, $b_2 = b_3 = 2500$
ADAT speed (v_1)	25000 dies per hour
PHICOM speed (v_2)	25000 wires per hour
Multi Plunger speed (v_3)	25000 molds per hour
Mean uptime machine 1, 2, 3	1033 s
Standard deviation uptime machine 1, 2, 3	1033 s
Mean downtime machine 1, 2, 3	243 s
Standard deviation downtime machine 1, 2, 3	243 s
Replications	100 x 100 orders

Discretization	line throughput (kUph)	relative error
0 (continuous simulator)	17357 ± 73	-
625	18156 ± 201	4.6%
1250	17406 ± 155	0.3%
2500	16369 ± 112	-5.7%

Sub-scenario c

Parameter name	value
Number of machines	1 ADAT, 1 PHICOM, 1Multi Plunger
Buffer sizes	b_1 infinite, $b_2 = b_3 = 2500$
ADAT speed (v_1)	20000 dies per hour
PHICOM speed (v_2)	22500 wires per hour
Multi Plunger speed (v_3)	25000 molds per hour
Mean uptime machine 1, 2, 3	1033 s
Standard deviation uptime machine 1, 2, 3	1033 s
Mean downtime machine 1, 2, 3	243 s
Standard deviation downtime machine 1, 2, 3	243 s
Replications	100 x 100 orders

Discretization	line throughput (kUph)	relative error
0 (continuous simulator)	15082 ± 47	-
625	15522 ± 178	2.9%
1250	15124 ± 120	0.3%
2500	14205 ± 85	-5.8%

Sub-scenario d

Parameter name	value
Number of machines	1 ADAT, 1 PHICOM, 1Multi Plunger
Buffer sizes	b_1 infinite, $b_2 = b_3 = 2500$
ADAT speed (v_1)	25000 dies per hour
PHICOM speed (v_2)	22500 wires per hour
Multi Plunger speed (v_3)	20000 molds per hour
Mean uptime machine 1, 2, 3	1033 s
Standard deviation uptime machine 1, 2, 3	1033 s
Mean downtime machine 1, 2, 3	243 s
Standard deviation downtime machine 1, 2, 3	243 s
Replications	100 x 100 orders

Discretization	line throughput (kUph)	relative error
0 (continuous simulator)	15065 ± 52	-
625	15420 ± 171	2.4%
1250	15156 ± 128	0.6%
2500	14300 ± 78	−5.1%

Sub-scenario e

Parameter name	value
Number of machines	3 ADATs, 4 PHICOMs, 1Multi Plunger
Buffer sizes	b_1 infinite, $b_2 = \ldots = b_8 = 2500$
ADAT speed (v_1)	25000 dies per hour
PHICOM speed (v_2)	25000 wires per hour
Multi Plunger speed (v_3)	25000 molds per hour
Mean uptime machine 1, 2, 3	1033 s
Standard deviation uptime machine 1, 2, 3	1033 s
Mean downtime machine 1, 2, 3	243 s
Standard deviation downtime machine 1, 2, 3	243 s
Replications	50 x 100 orders

Discretization	line throughput (kUph)	relative error
0 (continuous simulator)	15429 ± 56	-
625	16198 ± 223	5.0%
1250	15814 ± 220	2.5%
2500	14524 ± 115	−5.9%

Sub-scenario f

Parameter name	value
Number of machines	3 ADATs, 4 PHICOMs, 1Multi Plunger
Buffer sizes	b_1 infinite, $b_2 = \ldots = b_8 = 2500$
ADAT speed (v_1)	20000 dies per hour
PHICOM speed (v_2)	22500 wires per hour
Multi Plunger speed (v_3)	25000 molds per hour
Mean uptime machine 1, 2, 3	1033 s
Standard deviation uptime machine 1, 2, 3	1033 s
Mean downtime machine 1, 2, 3	243 s
Standard deviation downtime machine 1, 2, 3	243 s
Replications	50 x 100 orders

Discretization	line throughput (kUph)	relative error
0 (continuous simulator)	13439 ± 23	-
625	14131 ± 116	5.1%
1250	13849 ± 133	3.1%
2500	12715 ± 102	−5.4%

Sub-scenario g

Parameter name	value
Number of machines	3 ADATs, 4 PHICOMs, 1Multi Plunger
Buffer sizes	b_1 infinite, $b_2 = \ldots = b_8 = 2500$
ADAT speed (v_1)	25000 dies per hour
PHICOM speed (v_2)	22500 wires per hour
Multi Plunger speed (v_3)	20000 molds per hour
Mean uptime machine 1, 2, 3	1033 s
Standard deviation uptime machine 1, 2, 3	1033 s
Mean downtime machine 1, 2, 3	243 s
Standard deviation downtime machine 1, 2, 3	243 s
Replications	50 x 100 orders

Discretization	line throughput (kUph)	relative error
0 (continuous simulator)	14108 ± 45	-
625	14710 ± 173	4.3%
1250	14381 ± 155	1.9%
2500	13441 ± 128	−4.7%

APPENDIX D

Order list for comparison of flow line and job shop

The following order set has been used for the simulations in Chapter 5.

ProductID	ordersize
1	80000
2	80000
5	80000
1	80000
3	80000
5	80000
2	80000
1	80000
4	80000
3	80000
4	80000
3	80000
5	80000
4	80000
2	80000
4	80000
2	80000
5	80000
1	80000
3	80000

Bibliography

[1] Image taken from `http://www.aandrijvenenbesturen.nl`.

[2] Image taken from `http://en.wikipedia.org/`.

[3] Image adapted from `http://nxp.com/`.

[4] H. Tijms, *Stochastic models: an algorithmic approach.* Wiley, Chichester, 1994.

[5] J. Little, “A proof of the queueing formula $L = \lambda W$,” *Operations research*, vol. 9, no. 3, pp. 383–387, 1961.

[6] J. Jackson, “Jobshop-like queueing systems,” *Management Science*, vol. 10, no. 1, pp. 131–142, 1963.

[7] J. Li, D. Blumenfeld, and J. Alden, “Comparisons of two-machine line models in throughput analysis,” *International Journal of Production Research,*, vol. Vol. 44, no. No. 7, 2006.

[8] S. Gershwin, *Manufacturing Systems Engineering.* Prentice Hall, 1994.