Fitness of Simulation

Advancing the application of Discrete Event Simulation within the Automotive Industry

Michael Higgins

School of Engineering

Cardiff University

This thesis is submitted to Cardiff University in fulfilment of the requirements for the Degree of Master of Philosophy

December 2013

Abstract

A series of case studies are used to explain current simulation approach in a large automotive manufacturing plant. The effect of data input is investigated and quantified, highlighting requirements for accurate simulation. A study on bottleneck analysis within Discrete Event Simulation is undertaken and a framework for more accurate investigation created. Finally, optimisation studies are carried out on real world engine assembly manufacturing lines in order to demonstrate the benefits of advances made in this study.

Acknowledgements

I begin by thanking Ford Bridgend Engine Plant for allowing this project to proceed and in particular Dave Canfield and Dan Doherty who provided support as industrial supervisors. Thanks further to all those at Ford whom assisted in data collection and the operators on the line who tolerated my inquisitiveness.

My sincere thanks are also given to Paul Prickett whom supervised this project and provided direction and focussed my optimistic nature in to the key areas presented. Thank you for supporting my development as project supervisor. My gratitude extends to Anthony Soroka who provided support throughout the Knowledge Transfer Partnership structure.

Thank you to all whom have contributed through support, data or otherwise. I appreciate everything you have done to help me to reach my goals.

Contents

Acronyms and Abbreviations

BEP	-	Bridgend Engine Plant
CML	-	Continuous Moving Line
DES	-	Discrete Event Simulation
EOL	-	End Of Line
FAST	-	Ford Alternative Simulation Tool. Previously, Ford Assembly Simulation Tool
FTT	-	First Time Through
GUI	-	Graphical User Interface
HPV	-	Human Process Variation
IE	-	Industrial Engineer
JPH	-	Jobs Per Hour
KPI	-	Key Performance Indicator
KTP	-	Knowledge Transfer Partnership
MODAPTS	-	MODular Arrangement of Predetermined Time Standards
MTBF	-	Mean Time Between Failure
MTTR		Mean Time To Repair
OEE	-	Overall Equipment Efficiency
PAG	-	Premier Automotive Group
QMS	-	Quality Monitoring System
TSI	-	Time Series Investigation
VBA	-	Visual Basic for Applications
V+V	-	Verification and Validation
WCL	-	Witness Command Language
WIP	-	Work In Progress
WS	-	Work Standard

Chapter 1

General Introduction

Discrete Event Simulation (DES) is a decision making tool that is widely used throughout the manufacturing industry. The reasons for utilising simulation differ, but it is the case that the overall use of simulation in manufacturing is increasing. This can in part be accounted for in a shift from the early DES systems, requiring in depth programming code knowledge, to the graphical user interface (GUI) of today's modern systems. The evolution of the DES software has resulted in packages that are more accessible to non-specialists and so expensive consultancy is not always required. The recent advances in technology mean computers are more than adequate to meet the demands of the DES calculations. This results in simulation experimentation taking less time and a great reduction in cost since expensive computer systems are no longer required. Further to this point, a greater amount of detail can now be modelled. Finally, advances in the DES software result in accurate simulation being available through the use of standard modules which can be edited to meet the user's requirements. As computer processing power increases the level of detail that can be captured and simulated within models improves. Thus, it is now possible to accurately simulate whole assembly lines with large amounts of complexity built-in. It is however important that such models accurately represent the true situation and can be modified when necessary to reflect real life changes.

This thesis concentrates around the usage of simulation within Ford of Europe and in particular the Ford Bridgend Engine Plant (BEP) in South Wales. In doing so it also identifies the benefits of applying simulation to a modern manufacturing system and the requirements associated with its effective deployment.

The Ford Motor Company relies heavily on its UK engine production with over 1.5 million engines being assembled in either Bridgend or Dagenham, East London, engine plants in 2013 (WalesOnline, 2014). With the assistance of the Welsh Development Agency the BEP plant was opened in 1980 and since then has been a major employer in the local area currently employing over two thousand staff across several facilities within the plant. Within BEP there is a mixture of both machining lines and assembly lines building engines for not only Ford cars but also many of the former Premier Automotive Group including Jaguar Land Rover and Volvo. Currently there are three engine assembly lines in use producing a variety of different petrol engines for these companies. This includes many derivatives of V-Configuration engines for Jaguar Land Rover on the internally named AJ Engine Assembly

Line and in-line 6 engines for Volvo on the SI6 line. The third line, and one with highest production capacity, is the Ford Sigma engine line producing engines in various litre capacities spanning from the base 1.25 litre engine to the 1.6 litre GTDi engine found in the highly successful Ford Fiesta ST. To highlight the volume of engines produced at BEP Table 1.1 is included showing the 2009 production volumes:

Table 1.1 – Production Volumes of Ford BEP for 2009 retrieved from (Ford UK, 2010)

Engine Line	Manufacturer	Engine Details	Volume of Engines
Sigma	Ford Fiesta, Focus, Fusion and Volvo S40	1.25/1.4/1.6-litre	575,289
Si6	Volvo and Land Rover	3.2-litre Si6	64,613
AJ	Jaguar and Land Rover	3.5-5.0-litre V8	43,438
		TOTAL	683,340

Since 2009 there has been significant investment, from Jaguar and indeed Ford, into BEP such that the figures in Table 1.1 no longer accurately reflect the production capacities of the individual lines. Due to a lack of publicly available information on production volumes there are no more recent figures available across all derivatives. However, the 2009 figures are included to highlight the size of the operation at BEP. The Sigma line is now the highest capacity line and Si6 line is the lowest as demand for the engine has fallen off. This sets to continue as the Sigma line is primed to produce an additional, high profile, 1.5-litre derivative Ford engine from early 2014 through funding from the Welsh Assembly Government (BBC, 2013 and Ford UK, 2013). The major parts these engines are assembled from are also machined within BEP including the cylinder block and head as well as the crank and cam shafts. Further to this, there are two cylinder head assembly lines inserting various components ready for assembly. In order to illustrate the derivative of engines built at Ford BEP a diagram has been created and is included as Appendix 1.

Each of the assembly lines at BEP contains complexity such as multiple derivatives requiring different batch and manning strategies. These are just a few of the areas that DES projects could be applied to increase understanding and find optimum solutions. At the time this project was undertaken there was no simulation engineer, Ford staff or otherwise, based at the BEP site. This is not uncommon throughout Ford assembly and machining facilities since the Ford staffing framework does not include a position for simulation staff within its plants.

Instead a team of simulation engineers are contracted in at the Ford Dunton Technical Centre (here forth Dunton), Essex, and support plants across Europe, and South America, through the use of DES. Working at the hub of Ford Europe's centre of innovation allows simulation to be used throughout the design phase whilst supporting plants remotely in continual improvement tasks. If simulation is required within BEP contact must be made to Dunton through management and a project arranged. It is for these reasons that the majority of simulation work undertaken at BEP, in the past, has been completed remotely with the simulation engineer often never visiting the facility. The effectiveness of the methods used in this manner is reviewed throughout this thesis.

Ford, in conjunction with Cranfield University, has undertaken work improving the accuracy of the DES they create. This includes the development of a bespoke spreadsheet to DES interface called FAST (Winnell, 2003) which is used to increase the accessibility to simulation (Ladbrook, 2001). A large amount of work has been applied to produce this standard approach to modelling as a platform to improve simulation application. This includes modelling of humans and in particular the variation found for different operators at different ages (Baines, 2004) which concluded that the human process has a clear impact on the performance of a simulated production system. The FAST interface and resulting modelling approach has been critiqued by simulation experts who concluded that the interface allowed access to non-simulation experts and in doing so moved forward towards Component Based Simulation (Page, 1999). The DES tool utilised throughout Ford is the Witness software, created by Lanner. The current simulation toolset utilised by Ford is discussed in more detail in Chapter 2.

Manufacturing engineers at BEP have little confidence in the use of simulation due to scepticism over both the amount of and lack of detail that can be captured within a simulation. The simulation engineers at the offsite Dunton Technical Centre cannot fully understand the assembly line and associated details due to their lack of exposure of the real line.

Through research and advancing the application of DES the aim of this work was to increase confidence and credibility in future simulations. In order to do so there was requirement for a full understanding of the current methodologies of DES within Ford Europe as well as to gain a real insight into the way assembly lines function within Ford. The best way for this to occur was to gain a placement at a functioning Ford engine plant and visit the offsite Ford simulation team as required. For these reasons the BEP site was chosen and a placement began in 2012. This has allowed an external review of the modelling approach that Ford are currently utilising across Europe. This has led to the review and improvement of the FAST interface. In order to support this development of knowledge and theory many models have been created of real world assembly lines and machining lines within BEP. These are used as case studies throughout this project but have also been used by BEP to gain a better understanding of their processes.

This project is particularly involved with advancing the methods used when simulating assembly lines. This was decided to be the focus of attention due to the greater complexity and variation of assembly lines; the additional human variation factors in assembly lines are not as widespread in machining. It was decided that improving application of assembly line simulation could provide a framework to improve many of the processes in machine line simulation. In order to support future modelling an instruction manual has been created for use by new simulation engineers. A review of the methods currently in use by Ford to build, verify and validate the model is discussed and new methods developed to approve the models as representative of the real system. This was decided to be necessary due to an initial lack of confidence in simulation found at BEP. Dissemination of results is discussed throughout these sections based on the extended experience gained by the author within the Ford simulation plant. All of the theory, researched and newly developed, is combined leading to a substantial cost saving for the BEP plant.

Whilst being based within Ford the work undertaken is of wider interest to the simulation community as it identifies the role that can be played by DES and the associated requirements for accuracy this creates.

1.1 Plan for Thesis

The remainder of the thesis is structured as follows: Chapter 2 discusses the current applications of DES within Ford Motor and the toolset that is utilised throughout the thesis is introduced. Chapter 3 discusses the application of DES through the modelling of several real engine assembly lines and the elements used in a real manufacturing facility examined.

Chapter 4 reviews the current processes used in verifying and validating assembly lines. Within this chapter a new approach is developed to address gaps in current process leading to greater credibility of simulation models within BEP. Chapter 5 explores key data inputs and assesses their value when producing simulation models of assembly lines. Chapter 6 uses all theory from previous chapters in an optimisation case study leading to a substantial saving within a large automotive plant. Chapter 7 summarises the benefits of the work included in this thesis, to the host company, BEP. The final chapter, Chapter 8, presents the research findings, contributions to knowledge and future work to conclude the research.

There is no formal literature review; instead work is reviewed where relevant within individual chapters.

1.2 Overview of Initial KTP Project Objectives

The work contained within this thesis has been part funded through the Knowledge Transfer Partnership (KTP) Project 8270. The initial KTP project planned to address the gap in knowledge between the manufacturing plant, BEP, and Cardiff University. The author joined this project as a second 'associate' and work conducted within this thesis has been undertaken over a fourteen month period. There were several key objectives to be addressed with the overall goal of increasing the effective application of DES within BEP to increase the efficiency of manufacturing processes and leave a legacy to enable future work of similar nature. In particular the project specification submitted to the funding body aimed to undertake the following activities:

> *The examination and analysis of the operating manufacturing and inspection systems and processes deployed to enable their simulation.*
>
> *The modelling of the manufacturing and inspection systems deployed using a combination of discrete-event simulation packages.*
>
> *The testing and validation of the simulation models against actual system performance using appropriate metrics and observed operational characteristics.*
>
> *The discussion of results and identified process characteristics with management and engineers to enable the selection of the elements suitable for improvement and implementation.*
>
> *The extraction of data and preparation of models for performance optimisation.*

The application of suitable optimisation techniques and their application in the simulation of systems.

The discussion of results and preparation of plans to support the implementation of system optimisations.

Training of engineers in the use and application of the tools and techniques.

Throughout this thesis these points are expanded upon and form the basis for this research.

Chapter 2

A Review of the Current Application of DES in the Ford Bridgend Engine Plant

There are a number of DES packages available that are capable of simulating manufacturing processes. This project focusses around the use of Witness which has been in use since the mid 1980's (Lanner, "Ford Engine Assembly Line Simulation Tool Case Study"). In particular, this thesis considers the bespoke excel to Witness interface that is Ford Alternative Simulation Tool, or "FAST" initially Ford Assembly Simulation Tool. The following chapter discusses the use of a DES package within Ford, considers why Witness DES is utilised in particular and discusses the motivation for creation of the FAST interface. In a wider context this allows discussion of the information requirements when establishing a DES model and the link between model depth and the accuracy of the results produced.

2.1 Formal Definition of DES

Before discussing how Discrete Event Simulation (DES) is utilised within Ford it is important to understand what is meant by this term. For the purposes of simulation, DES is the process of representing the behaviour of a complex system as an ordered sequence of well-defined events (Greasley, 2004). The simulation moves from event to event as changes in the system occur over a progressing time scale. This is in contrast to continuous simulation which varies continually with time (Greasley, 2004). As an explanatory example, the modelling of a shop could be attempted using DES whereas the modelling of a flow of water would require continuous simulation.

In a simulation context and referred to throughout this thesis, an event is an instantaneous occurrence that has the potential to change the state of the system. An entity is an object which travels through the simulation activating events and a DES is a mathematical model of a system represented by entities that cause events to change the state of that system.

Simulation is a tool that is being more widely utilised within the manufacturing industry as the competition increases and manufacturers try and benefit from the latest technology in order to remain competitive. There are a variety of purposes and uses for simulation within a manufacturing context. Mohsen Jahangirian et al (2010) required 24 different categories to capture their research on the different uses of simulation within manufacturing. This highlights the diverse use of simulation.

Within the manufacturing industry simulation has applications within both continuous improvements of the current system and forward planning for future changes. Generally, simulation modelling is used to assist decision making by providing a tool that allows a representation of a system to be built and thus analysed and better understood. Further to this it allows a cost free way of predicting changes within a system by modelling different scenarios and so is a useful tool in forward planning.

The benefits do not stop at process improvement however as there are many gains made indirectly through the implementation of a simulation tool. Greasley (2004) summarises the reasons for adopting simulation within process improvement; highlighting benefits achieved both directly and indirectly through the implementation of simulation. Table 2 shows these benefits interpreted for within the manufacturing industry.

Table 2 – Benefits of implementing simulation. Modified from (Greasley, 2004) for the manufacturing industry.

Direct Benefits	Indirect Benefits
• Prediction of future business performance • Avoids disruption to the real system • Reduces risk of failure on implementation • Allows overview of whole process performance	• Stimulate creativity • Increases communication between departments • Assists acceptance of change • Encourages data collection • Can be used as a training tool • Acts as a design aid

The direct benefits are those that the company aims to achieve through implementation of a simulation project. Within a large industrial plant, such as Ford BEP or other large organisation, a key benefit is gaining better understanding of future changes. As a result of such investigations management can gain a holistic view of both the current operation, and under different proposals. However, the indirect benefits that result, from seeking the direct benefits, can be equally valuable. It is the authors' experience that the data collection process, required for accurate modelling, can result in development of better data collection methods that have wider application outside the confines of the simulation project. It is often the case that the indirect benefits can supersede the direct benefits, especially in the shorter term. The environment and attitude to simulation within BEP, expanded upon in Chapter 4, can account for direct benefits being less well received in comparison to those gained indirectly that can be implemented with ease.

2.2 DES in automotive manufacturing

Simulation was once referred to as the "technique of last resort" but throughout manufacturing has now become an indispensable problem solving method (Banks, 2007). There are articles outlining the usage of DES throughout several areas of manufacturing including automotive, aerospace, robotics and production industry (Bangsow, 2012) where there are a large number of case studies showing the benefits of DES. This thesis will concentrate on automotive manufacturing and thus examples cited are within this sector only.

Many large global manufacturers in the automotive industry including Nissan, Volvo, Toyota, General Motors, Chrysler and Ford as well as many more utilise simulation as a tool. Nissan has published work outlining how DES is used in its automotive manufacturing plant Sunderland to support cycle time generation, general system understanding, capacity calculation and continuous improvement (Timmiss, Nissan Motor). Volvo, Sweden, has utilised DES in evaluating changes in not only manufacturing facilities but also in their distribution and process facilities (Faget, 2005). An automated DES bottleneck analysis toolset has been developed by Toyota and utilised (Roser, 2001). Ford has undertaken a great deal of work on increasing their application of DES within their engine assembly plants as well as increasing accuracy of simulation to move towards emulation. This includes modelling of details such as human performance variation (Mason, 2005) and complexities such as machining transfer systems such as overhead gantries. The individual manufacturers may utilise their simulations in different ways, however, all have the same goal in increasing their manufacturing efficiencies through the use of DES.

This project is mostly confined to considerations of the way that Ford utilises simulation as a tool within Europe although it should be noted that there are also simulation teams based in America and support for many of the Ford manufacturing facilities around the world.

The majority of the work undertaken by the Ford Europe Simulation team concentrates around engine production including both machining and assembly lines. This is due to the locality of the Simulation team whom are based in the UK where the two engine assembly plants, Dagenham Engine Plant and Bridgend Engine Plant (BEP) are located. Management at the plants request particular tasks to be simulated on a case by case basis and the simulation team complete it remotely from their location in Dunton. This is because in the current Ford structure there is no position for a simulation engineer to be based at a manufacturing facility.

The current usage of DES can be split in to two main categories; continuous improvement and forward planning. The continuous improvement application within Ford is focussed mainly around bottleneck analysis and how best to increase performance for the most effective monetary investment. The forward planning application includes simulating the changes proposed for a system whether it be how best to increase capacity or to look at the overall effect of introducing additional complexity. The forward planning application of DES is particularly important to Ford as the company continually releases new product changes to remain competitive. DES can therefore be used to spot any changes in performance of a system during the design phase. Spotting potential errors before implementation is extremely important to Ford as huge levels of investment are required to correct problems once a system is installed and this can also delay the launch of new products.

The usage of DES within Ford is increasing greatly as confidence in the simulations created is improved and in turn manufacturing plants reap the benefits. This is reflected in the number of team members employed to work on simulation in Europe increasing from five to twelve full time staff in the last five years (“Simulated manufacturing at Ford Motor Company”, 2013).

There remains however some concerns on the usage of simulation which is discussed by the author and others (Soroka et.al. 2013) based upon a recent in-house review. This review highlighted the need for better communication between the manufacturing engineers and the off-site simulation team. The investigation took the form of a survey gaining opinions across many key simulation stakeholders, including: manufacturing engineers, simulation engineers and plant management within Ford. The results of this survey highlighted gaps in understanding and/or differences in perspective; especially between the manufacturing and simulation engineers. The manufacturing engineers surveyed had a lack of awareness regarding the effort required to undertake a simulation project and were unrealistic with what could be achieved using simulation. Another key finding of the study was the element of trust held by the manufacturing engineers towards simulation application within BEP. The manufacturing engineers did not believe the simulation engineers had sufficient knowledge of the real assembly lines. As a result of this they felt there was a negative impact on model accuracy.

It is beyond the scope of this project to compare the varying DES software that could have been used for this project and as such the decision was made on the availability of software at Ford. However, it should be noted that there are several DES packages available on the market and reviews have been undertaken to compare the relative merits of the different software. A review in 2003 found there were 56 commercially available simulation packages, 18 of which used the DES method. A comparison of the three well-known DES packages, Arena, Simul8 and Witness has been undertaken (Semanco, 2013). Coincidentally these are the same three pieces of software that Greasley (2004) utilises throughout the referenced literature. The DES software utilised by Ford is the Witness simulation package. This was chosen by Ford due to ties back to the early systems developed, discussed below, and since then a toolset has been developed to supplement the use of the DES. Thus, although reviews of other DES packages are undertaken regularly Ford continue to use Witness due to the complete solution that has been developed to work specifically with Witness.

Witness software is DES software created by Lanner Inc. who can trace their roots back to 1978 with the SEE-WHY software (Lanner, “About Lanner”). British Leyland adopted this software early on in order to simulate the effects associated with introducing a new car to the plant (Hollocks, 2006). The system has evolved over the years moving from the basic DOS input of SEE-WHY, shown in Figure 2.1. Rebranding to Witness in 1986, through several company merges and acquisitions, implementing 3D capabilities in 2000 eventually resulted in the Witness 13 system that is available today (Lanner, “About Lanner”); the Witness 13 user interface is shown in Figure 2.2.

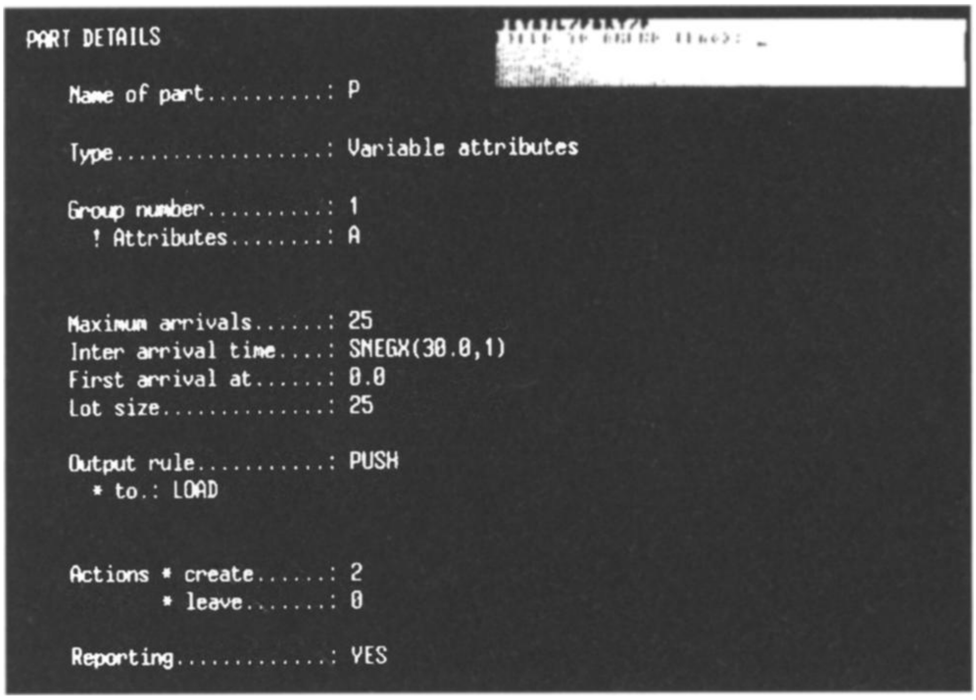


Figure 2.1: The early SEE-WHY software (Hollocks, 2006)

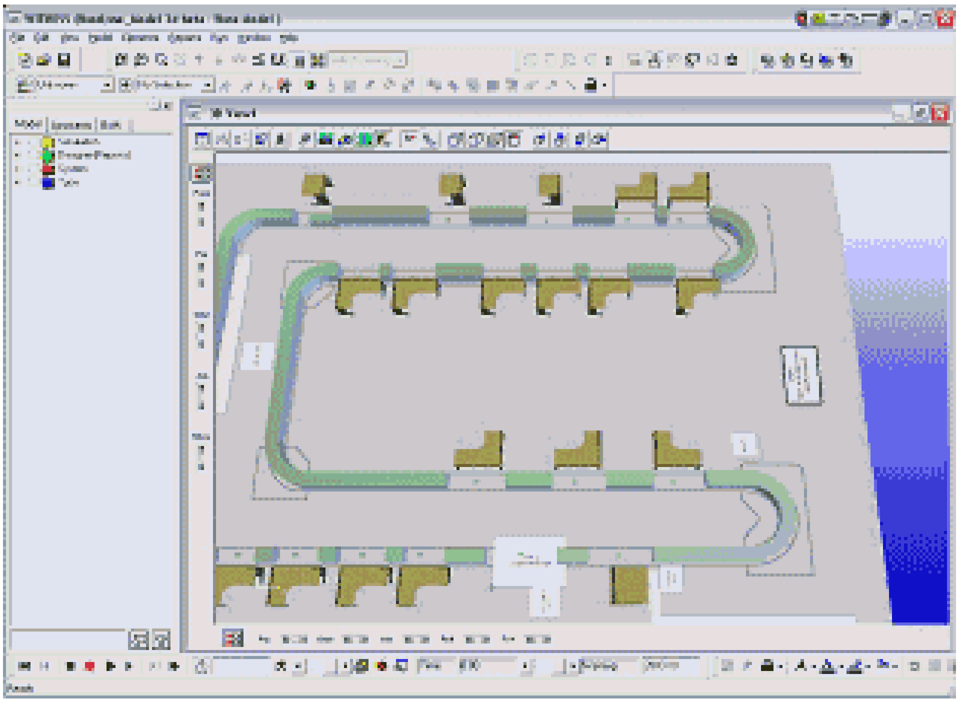


Figure 1.2: the Witness 13 Interface (www.lanner.com)

Lanner is always evolving the Witness software and there is no sign of this stopping as they look to continually improve and implement new technologies ("Lanner Forms Partnership", 2013). Witness is not orientated solely around the simulation of manufacturing facilities, although this is certainly one of its most popular applications. There are several case studies available highlighting application of the software in pharmaceutical and health care, defence and aerospace and many more categories outside of manufacturing, as shown in Table 2.1.

Table 2.1: Areas of application for DES (Diamond, 2010)

Discipline	Fields	Applications
Manufacturing	Aerospace, Biotech, Agriculture, Semiconductors, Food, Automotive, Pharmacy, Consumer products	Inventory and resource management, Six sigma, Lean initiatives, scheduling, capacity planning, evaluation
Service industries	Retail, Banking, Finance, Restaurants, Hotels, Insurance, Utility	Service Levels, scheduling, throughput analysis, evaluation, Six sigma/Lean initiatives, workflow
Communication/Networks	Call centres, satellite systems, Airborne Systems	Capacity planning, performance, evaluation, determination of reliability and fault tolerance
Transportation/Material Handling	Airlines, railroad, Freight, Cargo, Warehouse, logistics	Emergency planning, scheduling, service level, six sigma/Lean initiative

The use of witness in the automotive manufacturing sector is an established use of DES and the Lanner Company feel this application has reached “simulation maturity”. The introduction of Witness within the public sector is more recent and thus the software will adapt and improve to meet the demands of these alternative sectors. This is exemplified with new partnerships being formed, between Lanner and Frontline, to allow effective simulation of the NHS and other large government facilities (IT-Director, 2010).

Although Ford use the Witness DES to model their manufacturing facilities the model building aspect of the software is not always used directly. Instead a custom interface has been created for the purposes of easing data input and output. The full consideration and justification for this approach is given in Chapter 3. An overview of the reasons and requirements for this custom interface are now discussed.

One of the most complicated tasks undertaken by the Ford Simulation team is the modelling of the processes and the inherent variation of assembly lines. Although it may be argued that automotive assembly lines can be built from the same ‘jigsaw pieces’ and all lines share similar features, there are sufficient differences from one assembly line to another to mean that each line has to be built independently. It is not always possible to modify one model to fit another assembly line. Almost certainly it is not normally an efficient method to do so. The standard simulation approach therefore is to build an assembly line from scratch using the standard modules within the DES software. This is an effective approach in building models as it gives complete freedom to the modeller who can pick and choose the detail they feel. This allows a huge amount of detail to be modelled accurately using the DES platform (Law, 2007).

There are several major drawbacks to this approach. Firstly, the time involved in building custom models is high. This results in high costs and, perhaps more importantly, in the dynamic world of manufacturing and automotive assembly where the rate of innovation is high, the delay in producing results is costly. It is also possible that by the time the model building process is complete the data held within it may be out of date. The high level of freedom offered within Witness can actually result in subtle, yet costly, mistakes to be made. These are often hard to debug or are potentially never highlighted yet have the potential to severely skew results. Building models within Witness, although more user friendly than ever, still requires high levels of skill and experience to model manufacturing processes effectively.

Having worked with the Witness software since its early origins, Ford's European Simulation Team highlighted these problems and developed a framework to use the advantages of Witness and control the issues associated with directly building models within the Witness software. The FAST interface was developed with the help of a Lanner Senior Consultant (Lanner, "Ford Engine Assembly Line Simulation Tool"). FAST is a Microsoft Office Excel Macro Enabled Workbook which was designed to ease model building by providing a familiar format to the end user, as shown in Figure 2.3. The model is built within a well-designed spreadsheet and transferred to the Witness software automatically using Visual Basic for Application (VBA) code. Throughout the whole model building process the end user needn't even open Witness. This is why the term 'Simulation Engineer' changes to 'End User' when discussing FAST as it is designed to be user friendly enough for all engineers to access and not limited to those with simulation knowledge. The created macros can be used to navigate around the various data input sheets.

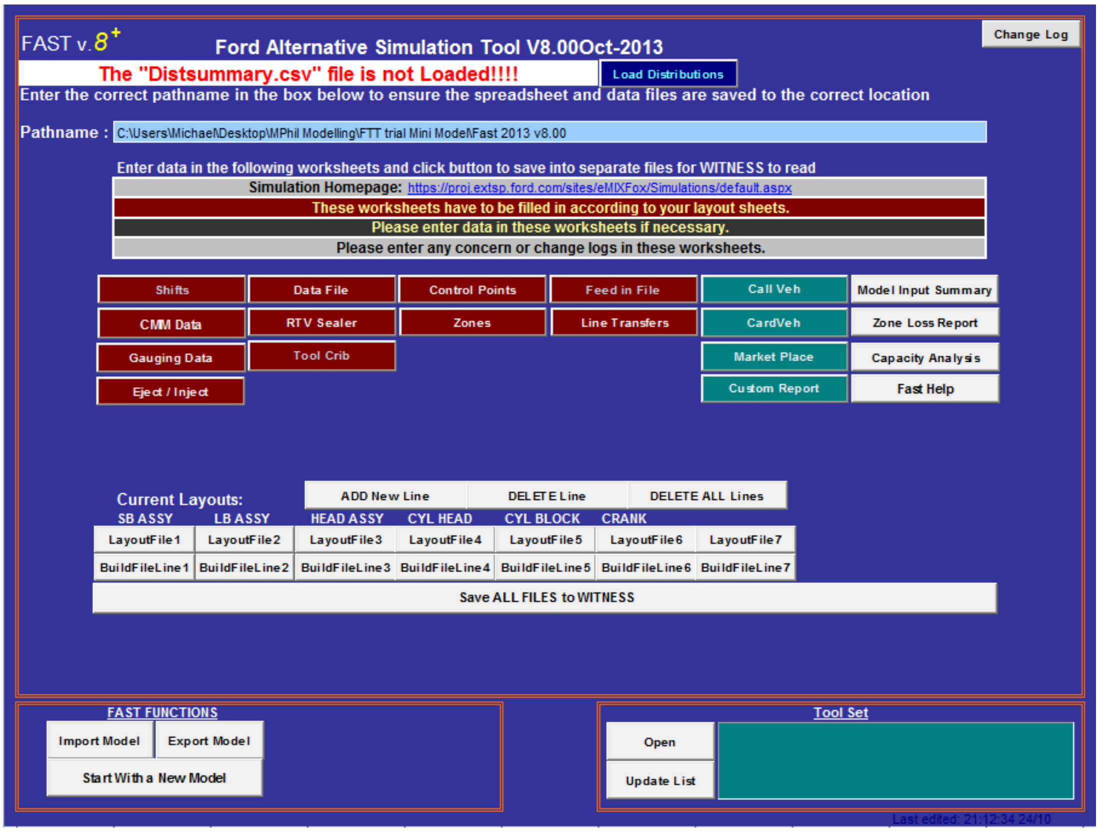


Figure 2.2: The FAST Interface splash screen.

The software was initially developed to address the modelling of engine assembly lines. The capability of FAST has been expanded and the software is no longer limited to assembly line simulation. Full consideration for machining line simulation, and material handling within the plant, has since been added. The FAST acronym was subtly rebranded as "Ford Alternative Simulation Tool" to reflect these changes.

The introduction of FAST has meant that the building of accurate models has been greatly streamlined and it is possible to create a representation of a manufacturing line with the use of just a few documents. All that is required to build a model is a scaled drawing of the plant layout for the line in question and a Work Standard (WS). The plant CAD layout is used to identify the relative positions of the operations and therefore routing and buffering can be derived from it. The WS is an official Ford controlled document, usually also in spread sheet format, outlining all of the operations and significant details of those operations including cycle times, manning levels and stock handling requirements. This limited exposure and modelling from a distance would not be possible using Witness directly.

Once all of the operations and relevant data have been added to FAST a macro is run to automatically transfer the data to a file that can be read by Witness, known as the Data File. All that is then required is for the end user to open Witness and run the model. This triggers Witness to read the Data File and automatically build the line as designed within FAST. The model is then able to run just the same as if it was built in Witness directly.

The Ford simulation toolset is not only used to speed up the model building process, but, as a result of a standard format for modelling implemented through the use of FAST, the running, data collection and analysis has also been simplified. This is possible since FAST imbeds particular triggers within all models that these external tools can refer to. This is done without option to the modeller. The toolset includes the 'Command Generator' which is another spreadsheet utilised when designing experiment runs. The user enters information such as the length of the run and if breakdowns of machines, stock handling, quality and various other elements of the simulation are to be activated. This information is then converted to Witness Command Language (WCL) and held in a file. When run, this file opens the simulation, which has been created with FAST, and turns on/off the various options as defined within the Command Generator. These model runs needn't be run on the computer the simulation was initially created on. The Command Generator spreadsheet has built in functionality which can run the experiments on a remote computer system known as the High Performance Computing (HPC) held within America. This allows the user to continue using their desktop PC whilst waiting for their simulation runs to finish.

Running the experiments using the Command Generator also has the advantage of activating the recording of the data autonomously at user defined intervals. When the model is run using this method a report file is generated from Witness. This report file holds information on the various elements used throughout the Witness simulation such as the percentage busyness of machines or the average number of parts held within a buffer. There is a lot of information held in such a file, especially in larger models made up of many different elements. The analysis of such information would be daunting if completed from scratch; operations would need to be separated by type, then analysed on each individual statistic requiring careful and advanced statistical treatment. This process has again been streamlined through the use of yet another spreadsheet known simply as 'Result Collector'. The Result Collector, again through the use of VBA, allows the user to read in to any spread sheet particular items of interest individually. This may be the throughput of the model or information on the equipment utilisation. From here the modeller can either choose to use the spreadsheet's inbuilt functions to chart the information or there are many Ford custom simulation charts available, again through the use of macros and VBA.

Overall, the toolset developed by Ford supports the development of a simulation project throughout; from building the model, designing the experiment, running the model and finally analysing the results. This allows a greater accessibility to simulation from non-specialists and eases and speeds up the process from project conception to delivery of results.

2.3 Summary of chapter

There is nothing unique about utilising programs external to the DES to control and record simulation runs. Many companies choose to use the same approach (Lu, 2007 and Ulgen, 2007). However, it is important to note that Ford has developed a full solution with the aim of allowing accessibility to all parties, whether they are simulation trained or not, through the model building process, experimentation and analysis. The interaction of these different tools form a complete solution that has not been highlighted in research of other automotive plant DES usage. It forms the starting point for this research, which considers some of the ways in which the existing approach can be modified to take advantage of the innovations in DES that have evolved since FAST was conceived. It also confronts the lack of buy-in apparent within the manufacturing engineering functions regarding the benefits of DES.

Chapter 3

Improving DES Modelling and the understanding of its application to Automotive Assembly Lines

The purpose of this chapter is to consider the potential to advance the modelling of assembly lines using DES. Although the models referred to in this chapter have been built to meet the requirements of Ford BEP engine assembly lines, the techniques and methods discussed and developed are relevant to many other manufacturing facilities. To begin with, a discussion of the various terminology associated with assembly lines is provided.

An assembly line is a process whereby there is an input of unfinished parts which travel through a well-designed sequence of workstations, where additional parts are added to produce a finished assembled product. Workstations in this context could refer to either an automated machine or a human operator and these workstations are connected together usually by a conveyor. There are two main types of assembly lines that are implemented within Ford; the Continuous Moving Line (CML) and Controlled Release Line (CRL). Although similar in their approach there are major differences between these two types of line.

The CML pulls parts along, generally on a chain driven system, such that the part being assembled is continually moving. The operators move along with the part as it is assembled until the work is complete and the part moves into the next station for further work. CMLs were initially implemented within Ford to increase efficiency in assembling the Model T in 1912 (Ford, 2005) and are still used extensively in the final vehicle assembly plants. The CRL is built up of conveyors, that pull the part along normally using friction rollers and mechanical stops, which hold the engine in a given position for work to be carried out. The engine is released depending on a particular condition being met depending upon the operation being undertaken. There are three triggers for release; manual release where the operator is required to press a switch or foot pedal to indicate the job is complete, timed release where the engine is automatically released after a particular period of time has elapsed or automatic release whereby some physical condition is met triggering the release e.g. when 5 bolts have all been rundown to the correct torque the release is triggered.

The CRL is the main line utilised in the Ford engine assembly plants as although more expensive to implement, due to the control systems and software involved, it maximises customisation and minimises the effect of stoppages. Unlike the CML which comes to a halt if one station breaks down the CRL can utilise buffering to continue production around the broken down operation. Within BEP in particular there is only one section of one assembly line utilising CML, this is the oldest line in the plant highlighting CML as a legacy transfer system.

3.1 Assembly line terminology

It is difficult to discuss the details surrounding assembly lines, and indeed modelling of such systems, without the following definition of the important terminology commonly used.

Operation: Within Ford plants, any process that is carried out on the engine throughout the assembly line is referred to as an operation. Each operation on the assembly line is given an 'OP' number and a description of the job to allow easy reference. For example: OP190 – Fit Spark Plugs.

Station: A station is a defined area on the assembly line which has been allocated for a particular job (operation) to be undertaken. Stations are given a pitch number and are included within a zone, with a zone being a grouping of stations.

Manual Operation: A station that requires the intervention of a human operator to complete a task is considered to be a manual operation. This could be purely manual such as clipping a wire harness to an engine. There is some complexity to the definition of a manual operation within the Ford infrastructure. There does not seem to be a clear line where the use of automated machinery by an operator causes the operation to be considered Semi-Automatic. For simulation purposes a manual operation is one which is completed by a human and no automatic process is run requiring human interaction at any stage.

Automatic Operation: An operation that is undertaken without the use of a human to complete and is completed automatically. Automatic operations are used to carry out tasks such as running down several bolts in sequence to a particular torque. An operation performed by a robot arm is considered to be an automatic operation by the same definition.

Semi-automatic Operation: An operation that is neither solely automatic nor manual. This is an operation that requires the human element generally to start an automatic process. An example of this is an automatic machine which requires accurate positioning by a human.

Once positioned the automatic process is carried out and the operator may no longer be required.

Operator: A term used to describe the role for people working on the assembly line. Operators are required for manual and semi-automatic machines. Operators are grouped into teams which they rotate within to avoid monotony. There are many different 'grades' of operators within Ford and each grade increment has additional responsibilities. The higher graded operators are responsible for not only carrying out work at a station but are required to reset machines they are assigned to 'machine mind' i.e. reset local machines when required.

Group Leader: The group leader is an operator on the line with additional responsibilities for overseeing the running of their particular team. They may also be required to repair engines, cover operator relief breaks and carry out some management tasks such as ordering stock for the line.

Buffer: A buffer is a space reserved for the temporary storage of engines, generally between two operations. In the context of the Ford facilities, and the simulations created, the buffer refers to the capacity ahead of the particular operation referred to before the next operation. This may be counterintuitive to some since the buffer may be considered to hold stock prior to being processed for a particular operation. However, the use of the term buffer in this dissertation matches that used in Ford. For example, in Figure 3.1 the line flowing left to right, as indicated, if M1 and M2 are some operations then B1 and B2 are their respective buffers. To conclude, the buffer B1 is the space ahead of the operation M1 in the direction of line flow.

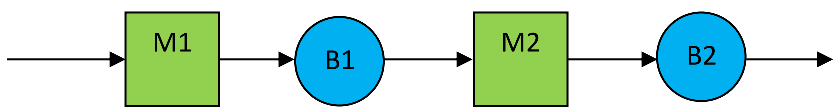


Figure 3.1: The operation of a buffer.

Cold and Hot Tests: The cold and hot tests are semi-automatic operations that have their own category. A cold test is performed on a newly assembled engine; the internal parts of the assembled engine are run using a mechanical device and the engine is not actually fired. Specialist equipment monitors the harmonics coming from the engine and passes or fails the engine on these measurements. An operator is required to rig the engine to a specialist set up, initiate the test process and then de-rig once completed. A hot test is carried out on a newly assembled engine which, like cold test, is used to test the engine and check if the engine is defective. The hot test cell, however, actually uses fuel and the engine is started in a controlled, specially designed cell. An operator is required to rig, start and monitor the engine throughout the multistage test process which is carried out autonomously. Modern engine assembly lines are fitted with the cold test cells as the test is more efficient in time, resource and floor space. Due to the amount of work required, for both cold and hot tests, the cycle times are generally greater than assembly line “TAKT” time and so require several of the same operations in parallel. (TAKT time is the minimum production rate required in order to meet customer demand and is calculated as the ratio between the time available to work and the customer demand.)

Conveyors: Used to connect and transport parts between the various operations on the line automatically. On a CML the conveyor used is chain driven where a hook catches the part and pulls it along. The hooks are separated on the chain by a distance such that the parts are kept a designed distance away from one another. On the CML if one conveyor stops the whole line will stop. The CRL conveyors use friction to drive the engines along using friction rollers. This allows the parts to stack up against each other. The speed of the line is generally around twelve metres per minute within Ford engine assembly plants although this is controllable and can be easily changed.

Pre-Stop: A pre-stop is used on a CRL conveyor to hold back an oncoming engine when work is being undertaken at a station. This stops the oncoming part from colliding with the in progress engine. The pre-stop is a mechanical stop that is activated and deactivated using proximity sensors, known as line full and line empty switches, on the conveyor. Figure 3.2 illustrates the effect of the pre-stop (red) which can be seen holding back parts (blue) as the work stations ahead (green) are occupied. The grey rectangle represents a conveyor flowing in the direction indicated by the yellow arrows.

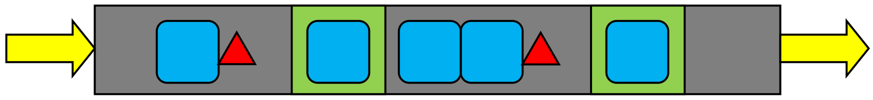


Figure 3.2: The operation of a pre-stop.

Turntable: A piece of hard automation that is used to re-orientate an engine. This is necessary since operators are generally found on the outer side of the line but work needs to be carried out on all four sides of the engine.

Tag: A computer chip that is written to automatically after every operation. The information recorded contains details of times the operation was completed and if it was completed without fault. This information gets uploaded to a central system at various points on the line.

Platen or Pallet: A platen is a specially designed rig that holds the engine steady as it transfers along the line on the conveyors. The design of the platen varies depending on the conveyor it is designed for; CML platens have a catch for the hook on the underside for example. The platen facilitates the assembly of the engine by providing the correct working height without impeding the assembly process. The platen also holds the 'tag' for the engine. Platens are sometimes referred to as pallets and the two terms are used interchangeably within Ford.

Repair Loop: A repair loop is an area off the main conveyor that is reserved for repairing engines. The repair loop, shown in Figure 3.3 is linked to the main line using a series of turntables. The turntable reads the tag for the engine and diverts the engine to the repair loop if it has failed any operation. Once in the repair loop the engine can be repaired by the group leader, or if the engine has a serious problem the engine is offloaded from the platen to a racking on the shop floor. The repair loop generally surrounds key test operations which have a high importance in the assembly process, for example leak test machines. Once repaired, if possible, the engine re-joins the main line and is again tested by the machines within the loop. Some lines do not use loops instead choosing to install 'repair spurs'. These are points on the line that allow removal of engines for repair to be carried out at a dedicated repair area.

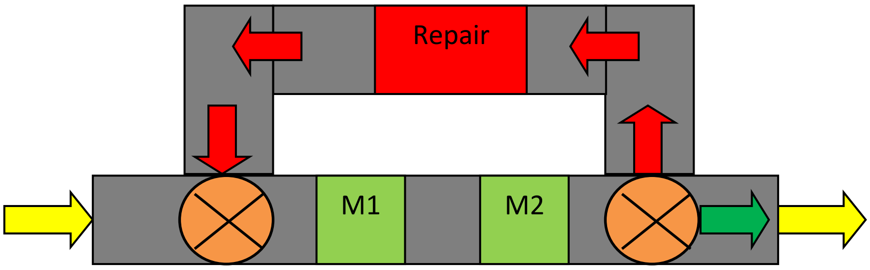


Figure 3.3: The operation of a repair loop.

An assembly line is normally made up from a combination of the above 'jigsaw' puzzle pieces, using elements that are linked together in a particular sequence. From this perspective the modelling of assembly lines can seem as simple as placing the individual elements in the correct order; this is the basis of the Ford FAST software. However, it is often the case that each of these elements are built up of many finer details and as a system there are additional measures that need to be modelled and recorded. If this data is not input, or incorrectly input, then the model may not represent reality. This is essentially the weakness in the FAST system. In the next section a brief overview of these items is undertaken.

3.2 Operation Specific Inputs – Local Variables

Depending on the aim and purpose of the simulation project different amounts of data input are required. In the most simplistic of cases, the model may well only require the elements to be built in the required sequence with buffering and cycle times input to FAST. This data should be sufficient to get the model running but it will more than likely require further data to be added to represent reality. After all, it is the modelling of stochastic behaviour (variation) within DES and its ability to predict output that makes it such a powerful resource. Variation comes in the form of adding breakdowns and quality into the model. Each of the inputs must be handled by FAST and Witness if they are to be used to model an assembly line. Each is associated with a particular operation on the line and thus each operation requires these inputs.

Cycle time: Is the time required to complete one cycle of an operation. That is, the time it takes from start to finish to carry out the allocated task on an engine. This is assessed differently depending on the operation. A manual operation undergoes MODular Arrangement of Predetermined Time Standards (MODAPTS) analysis, whereby the work content is broken down in to small movements and each allocated a predetermined number of 'MODS'. These MODS are then totalled forming a total time that is required for the work to be conducted or the cycle time. On automatic operations the cycle time is recorded from point to point using a stopwatch. This information is then recorded in the work standard, generally in decimal minutes. Each operation within FAST requires a cycle time to be inserted in decimal minutes. Within WITNESS the modeller can use their discretion to insert cycle times in their preferred unit of measurement, minutes is the default time unit although this can be altered.

Buffer capacity: The number of parts the can be held ahead of that particular operation. Within Witness this is entered as the integer number of parts that can be contained within the buffer. In FAST a length is input and converted to a capacity automatically. This length is measured, in metres, from the centre of one operation to the centre of the next, generally from a layout and scaled. There is a buffer override capacity within FAST such that the true, measured, number of parts can be entered if the simulation engineer desires.

Zone, Group Leader and Maintenance Team: These three parameters are entered as integers and allocate that particular resource to an operation. The zone, or team number at BEP, is the group that the operation belongs to. The group leader number is entered and is the leader responsible for any problems at the particular station – this person is called in the event of a breakdown or quality issue occurring within the simulation. The number of maintenance engineers is also entered such that particular maintenance personnel are allocated to particular stations.

Breakdowns: In order to model realistically the reliability of operations, breakdowns are included in the simulation. These breakdowns can be in the form of: machine failure where a machine is temporarily out of service and requires maintenance to repair causing unscheduled down time, and, operator breakdown where the operator is not 'functioning' within the designed parameters i.e. is missing from their station or over-cycling. The inputs for FAST are in the form of empirical probability distributions gathered from the assembly line by Fords custom data monitoring system POSMon. Two distributions are required within FAST the Mean Time Between Failures (MTBF) and Mean Time To Repair (MTTR), both based on real world data over a particular period of time and analysed. There are many approaches to adding breakdowns within Witness directly, most often data is collected and analysed to fit one of the many standard probability distributions. The breakdown input provides variability to the model.

Stock Handling: This is where the operator allocated to a manual station is required, in addition to their workload for the particular station, to handle stock at particular intervals. Manual stations within FAST have the ability to model these frequency based events by inserting the number of parts before a stock change is required, the time associated with the change as well as the labour required for the work to be carried out. The FAST system can also handle stock handling by derivative for mixed batch type assembly lines. The FAST system increments independent variables for each derivative to model stock handling accurately. A similar approach can be achieved through the use of Witness machines and the setup mode 'number of operations'. The information is collected by the simulation engineer through the work standard or MODAPTS analysis sheets.

Quality: Another source of variability within the model are the various quality metrics input into FAST. Within FAST the parameters required depend on the particular station. For example a turn table does not require any quality data, automatic/manual operations require an "FTT" percentage and number of retests, and a repair station requires a repair time. The First Time Through (FTT) percentage is used to assign the number of parts that have passed through that station without any issue. For example a 90% FTT figure input on to a station means nine out of ten parts, on average as based off random numbers, are unaffected. Within FAST, if the parameter 'number of retests' is input as greater than zero for that particular station then the one out of ten parts, on average, that fails the initial test is then given an additional chance to pass the test with a specified percentage of parts passing this retest. These retests can occur up to three times for each part and each retest is assigned an individual pass rate. If the part fails the initial test and fails to pass any retest it is stamped with an attribute "eject" such that it can be diverted for repair. When it reaches the repair point the length of repair is assigned based on a bespoke distribution that is default to all FAST models.

Material Handling – There are various inputs regarding the material handling such as when line side stock should be replenished from the store and replenishment cycle times and vehicles involved as well as the weight of boxes and many more. This functionality is not utilised within this project.

3.3 Model Specific Inputs – Global Variables

Warm-up and Run Length: These are parameters that are inserted within the Witness interface, directly. This is true for models built in FAST or Witness. The Run Length is the total amount of time the modeller wishes the model to be run for. As there is generally a bias on the initial run period of a simulation, before the model reaches steady state, a warm-up period is used. Once the warm-up time has elapsed all statistics are wiped clear leaving the model still populated and ready to start recording the unbiased results. These parameters are entered in time units consistent with the cycle time input. The optimum times for these are discussed in Chapter 5.1.

Number of Platens: Simply the quantity of platens that are circulating on the particular assembly line.

Control Points: This is a combination of global variables held within its own tab in the spreadsheet. Inputs include: the position on the line where parts are shipped or transferred to alternative lines, cold test cells position and hot test cells position. These inputs are used to provide additional logic for these specific areas.

Using the above conventions several DES models of assembly lines have been created in this research project. In order to investigate the relative merits of each system both Witness directly and FAST have been used. The step by step details of how each individual line was created is not included as it is covered within the training manual, provided in Appendix 2. Instead an overview of the lines created is presented and brief details about each included as well as the motivation behind each study. These models are then utilised through latter parts of the thesis. For greater clarity video captured clips of these models are available on the CD included at the back of this thesis.

3.4 The AJ Engine Assembly Line

The AJ Engine Assembly Line model, shown in Figure 3.4, was developed in late 2012 with the initial purpose of modelling proposed changes that had been planned to be implemented over the Christmas Shutdown 2012. The model, created in FAST, reflects the batch-model operation of the assembly line and three model variants have been fully detailed within FAST reflecting their different process times at the various stations. The model has been split into zones to reflect the bow tie distribution that the line uses. Quality and rework is effectively modelled using repair spurs and constrained by a variable containing the number of trolleys available to transport engines to the repair area. Breakdowns and stock handling are also modelled.

The changes initially simulated included the introduction of hard automation to replace a shuttle; a trolley on rails carrying two engines across a gap in fixed conveyor. Replacing this shuttle with a fixed length of conveyor resulted in an increase in the available buffering in this area of the line. The AJ assembly line model was used to predict the effect of the increase in buffering and further assessed if additional platens were required. Since the initial development of the model it has been used to support decisions for forward planning purposes for future model years and, as discussed later in Chapter 6, successfully utilised to optimise production.

3.5: AJ Cylinder Head Assembly Line

The AJ Cylinder Head Assembly Line was developed to supplement a study that was being undertaken by the Industrial Engineering (IE) department at BEP on this particular line. It was suggested by the author, that a model could be used to show the operators the effect of certain aspects of their job and also make use of the data that was being collected for the study. A series of data entry sheets were developed and completed by the IE team allowing easy conversion of data for the model. The model was developed in FAST to allow a familiar interface to be shown to the operators. Breakdowns are included with empirical distributions and stock handling has been modelled to a great amount of detail. Due to the many differences in cylinder heads built on this line the different heads require different platens. In order to capture this effect the model has been manipulated, by hard coding the FAST interface, to model platen change overs.

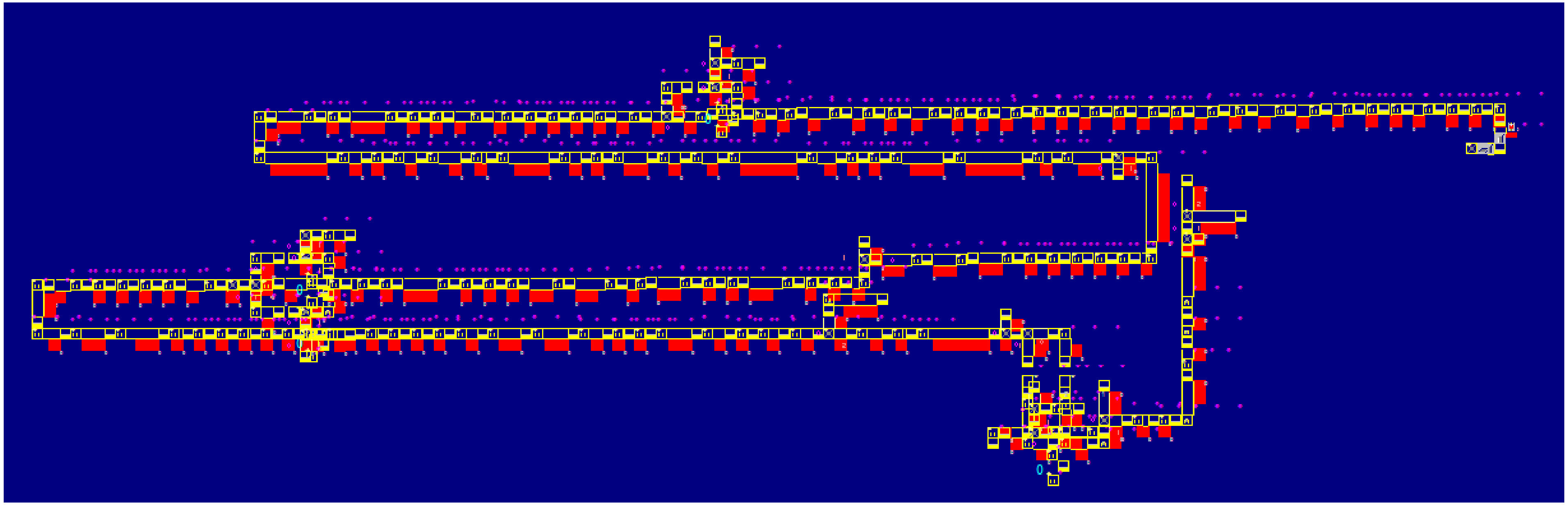


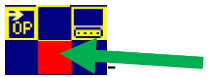
Figure 3.4: The AJ Engine Assembly Line model

<u>Key for FAST Models</u>

- Operation (Manual, semi-automatic or automatic)

 - Conveyor start. Yellow lines are used to connect conveyors

- Engine / part

 Graphic highlighting utilisation of previous operation

Three head types are modelled; the AJ V6 and V8 and the Si6 heads are modelled with different cycle times at the various stations as required.

The Witness model of the AJ Head Assembly line, shown in Figure 3.6, was also created by the author. The motivation for this was twofold; the visual display of the FAST model was poor and it was not possible to model many important features, unique to this line, within FAST. As can be seen by comparing Figures 3.5 and 3.6 the Witness display is far more visual, operators can be seen working at stations and different icons are used to represent key machines. To illustrate this, the Witness model includes dual operator machines, which require two operators to be present and two parts to cycle. It also shows platen change overs being initiated after a particular model batch has completed and robot arms that can have several parts in progress simultaneously as in reality

None of these features could be achieved through using FAST without significant modifications. Perhaps the most important aspect of the Witness model is the way it has been developed using functions to control the cycle times of the machines depending on the derivative. The function identifies the part and returns the required cycle time to the machine. This allows all of the cycle times to be entered in one file meaning the model changes can be made more quickly. Further to this, unlike FAST, the model can make full use of Witness optimiser to change cycle times autonomously and seek optimum solutions. Details at the right of the image include a method developed to control variation in batch sizing. As a result Witness optimiser can be used to help seek optimum batch strategies.

3.6 The Sigma Engine Assembly Line

The Sigma Engine Assembly Line model represents the most complex model built by the author. The complexity of the line means that the model has been split into two parts and modelled in two separate files within the FAST interface. The 'Main Line' section, labelled A in Figure 3.7 is the area from cylinder block load to the transfer across to the hot test cells. The 'Hot Test' section, labelled B in Figure 3.7 includes the 20 hot test cells plus the After Test Dress area of the line where the engine is primed for shipping. The Sigma Assembly line is one of, if not, the longest and quickest assembly lines in Ford Europe. In order to meet demand for the engines, there are a large amount of operations required to split the work

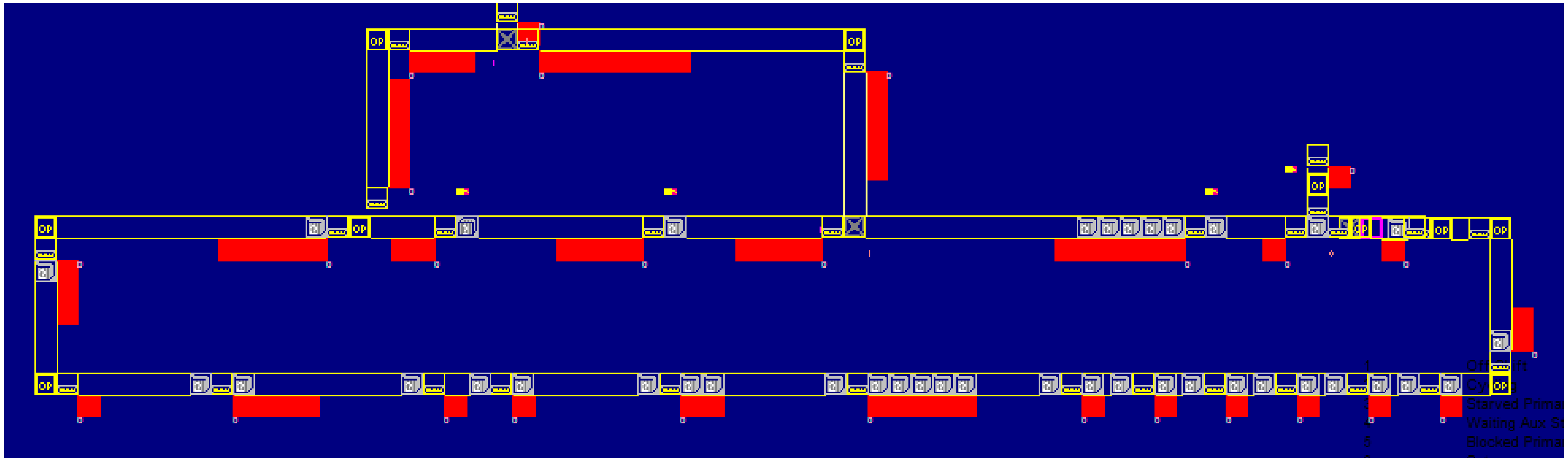


Figure 3.5: The AJ Cylinder Head Assembly Line

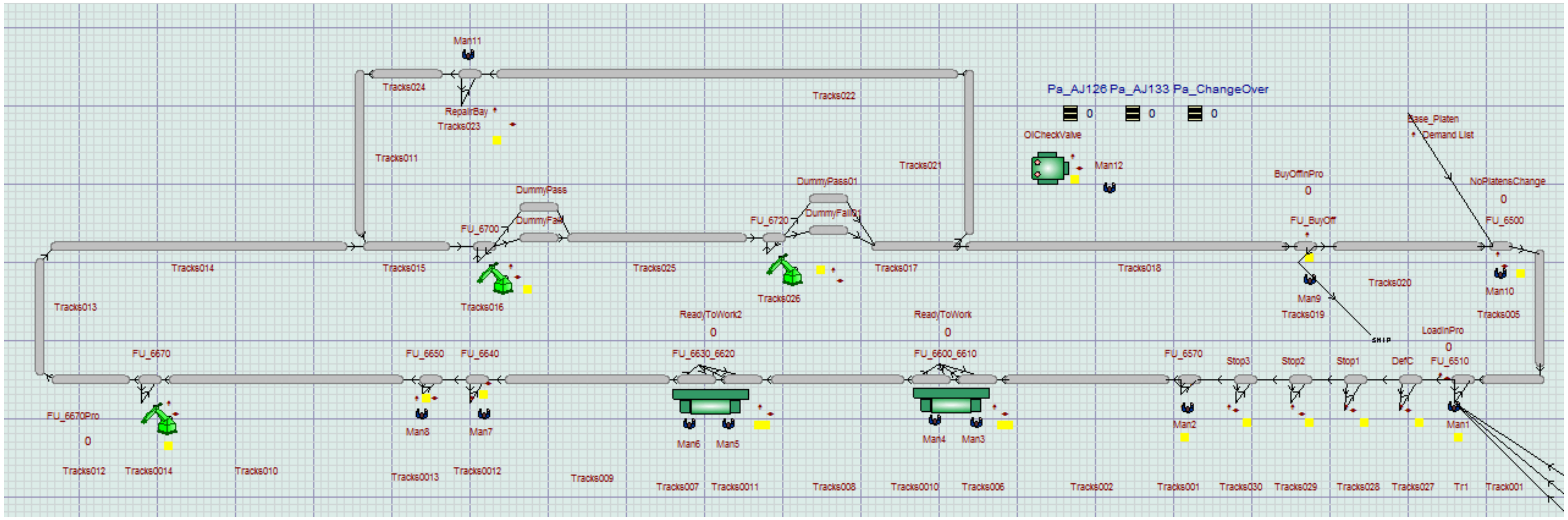


Figure 3.6: The Witness model of the AJ Cylinder Head Assembly Line

and remain below TAKT time. Since each of these stations requires programming with the associated data inputs there is a large amount of data held within the model. The model was built by the author on-site as the Ford Simulation Team could not create a valid model due to their location and the amount of data required to build such a model accurately. The model has been built in FAST to allow transfer of the model back to Dunton for their investigations.

Simulation tasks performed using this model have included modelling the effect of adding the 1.5 litre derivative to the line expected in early 2014. As a result a great amount of detail has been modelled including breakdowns, custom rework distributions, buffer over ride capacities and many more aspects outside of the scope of this discussion. In order to build the model some assumptions were required including not modelling the Sigma Cylinder Head Assembly, crank machining, piston con-rod and block machining lines as the increase in time required could not be justified. Thus, black box modelling has been used where appropriate.

In order to compare the relative merits of building models a Witness version of the Sigma Assembly line was built. The view shown in Figure 3.8 highlights the main assembly line and x-loop. The fullmodel can be viewed using the CD provided with this thesis. The full line contains the hot test area and after test dress as in the FAST model. Breakdowns are modelled through the use of distributions and there are eighteen different derivatives reflecting the batch model assembly line. The details of this simulation are kept to a minimum as although the simulation looks very visual it is so complex it is almost unusable.

The complexity of the models required to represent this very challenging assembly line meant that a more systematic approach was developed to modelling. Key sections were separated and modelled, including for example the Sigma Engine Assembly X-Loop Extension.

3.7 The Sigma Engine Assembly X-Loop Extension

The XLoop extension is a section of the Sigma Main Assembly Line as highlighted in Figure 3.7 and 3.8, labelled A1. The aim was to create a more detailed model of the X-Loop section of the sigma assembly line than would be possible in a full line model. This was first considered utilising the FAST software.

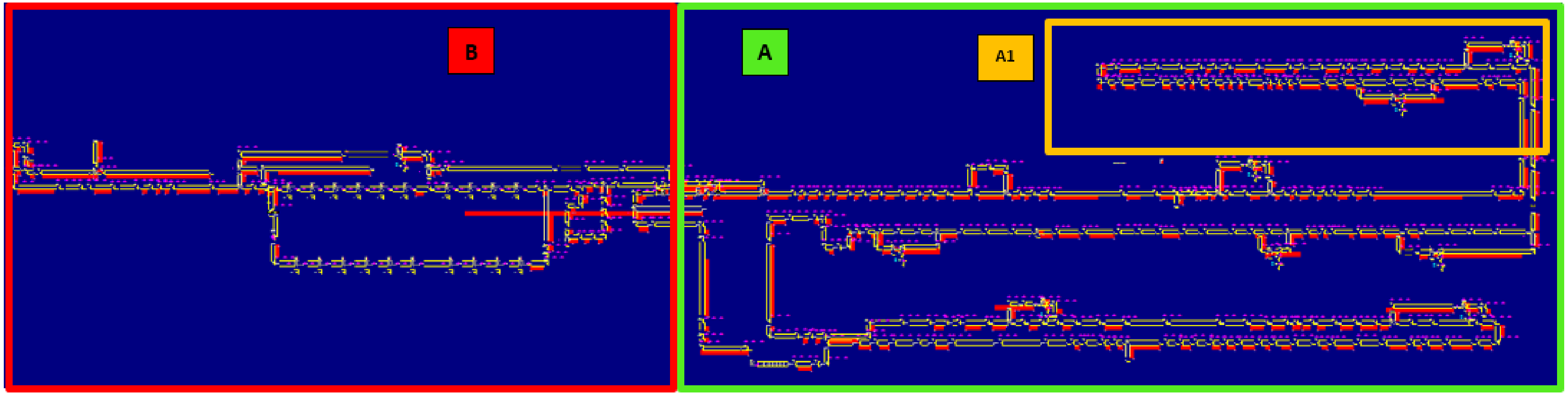


Figure 3.7: The Sigma Engine Assembly Line model

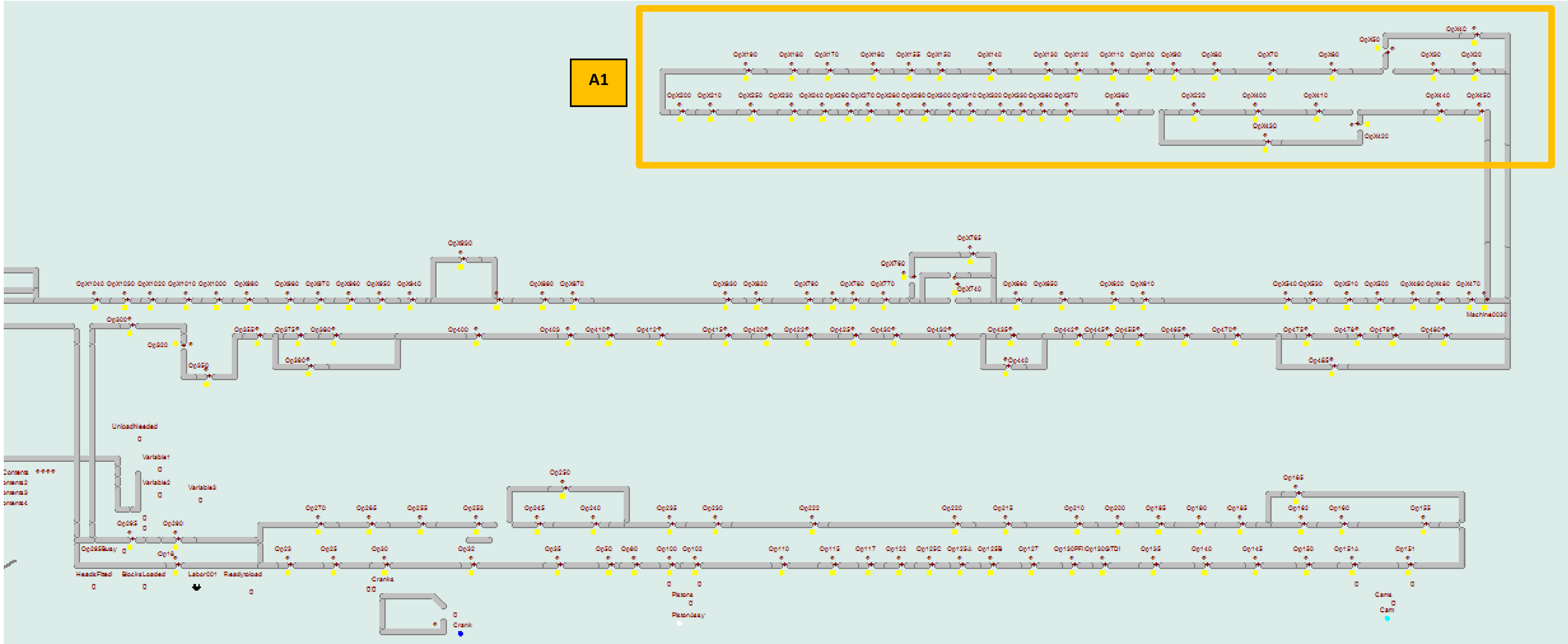


Figure 3.8: The Witness model of the Sigma Engine Assembly Line model

The motivation for creating the model, shown in Figure 3.8, was that, in the opinion of plant management, and based upon operational data this was a bottleneck. It was decided to build the model initially using FAST to give results similar to what had been seen before. The model includes details of FTT, stock handling (from detailed reports), breakdowns (from POSMon). As this model was not going to be used for display purposes the detailed modelling of operators was excluded, using a machine instead, except those used to model repairing machines. The FAST model highlighted areas of concern to plant for continuous improvement consideration. It was also, later, used to validate big changes made to the full line Witness model when it deployed to consider if breakdowns would have an effect.

The Witness model, shown in Figure 3.10, was developed to provide a visual tool for use within Ford constraint meetings. Constraint meetings are held within Ford BEP to review analysed data on key performance metrics such as the JPH and percentage availability of machines. Utilising an array of different skill sets the changes in performance are analysed and where appropriate plans are made to make both immediate and long term improvements. In order to aid this visual representation of the operation of the line the model was developed with the use of vehicles and tracks. To run the model unconstrained by platens a 'dummy track' with a large capacity was created. Parts are pushed in and pulled out of the model unconstrained by other areas before or after this section. The initial model did not include breakdowns or overcycles since these are less controllable by the team attending the constraints meeting. However, details of stock handling were included in great detail, with the option to activate or deactivate it using a user friendly interface. The model also included representation of the operators, again for visual purposes. The inputs can be added to the model non-directly using functions and attributes. Although this means that it takes a greater amount of time to build the model it does allow the copy and pasting of the data from a spreadsheet. The final model was used for modelling the subtleties such as changing the size of buffers and comparing the different key performance indicators (KPI). There is no mathematical variation included in the model so similar studies could have been undertaken numerically. However, the use of Witness produced a much better tool, a deterministic model, for displaying proposed changes. It was also later used to investigate the effect on personnel for union issues such as operator rotation.

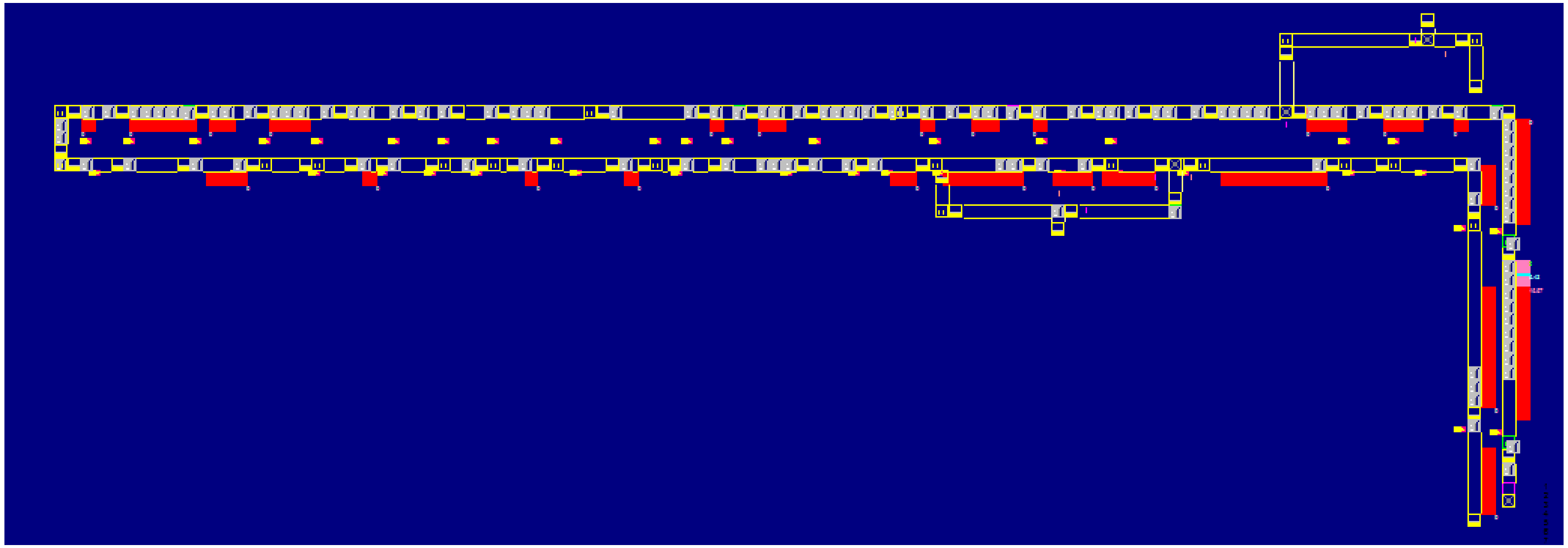


Figure 3.9: The Sigma Engine Assembly X-Loop Extension

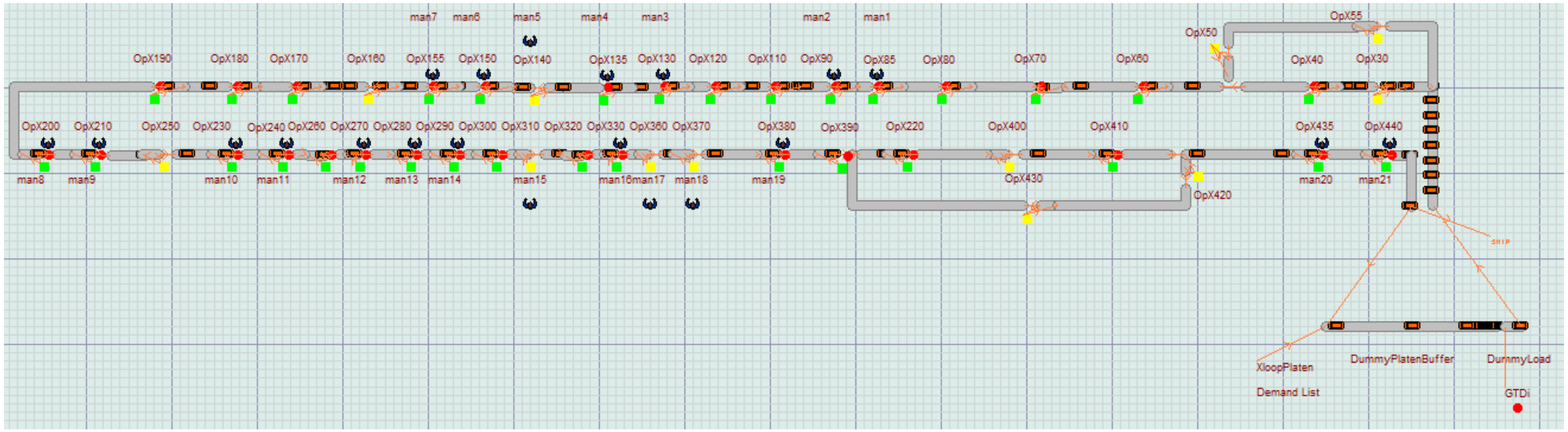


Figure 3.10: The Witness model of the Sigma Engine Assembly X-Loop Extension

3.8 Observations on the selection of the modelling process

If another person, such as a Witness Consultant or Ford Simulation Engineer, were asked to simulate these same lines they may well decide to use different methodologies to do so, even within the same FAST or Witness tools. After creating all of the models and using them it is very apparent that there are many ways to create a seemingly correct simulation. What is important is the model answers the questions the simulation was set out to answer. As explained within this chapter there are times when the visual display from the model is vital, and thus Witness is employed directly. On other occasions the visual aspects can be ignored as it is purely the statistics output that is required and FAST is used instead. To conclude, the tool used and methods employed are irrelevant if the model fulfils the requirement set out for its development in the first place.

In order to transfer the knowledge and processes developed a full training manual has been written. The manual is primarily based around the use of Witness for the modelling of assembly lines, with brief discussion on the use of Witness to model other manufacturing lines such as machining. Although Lanner provides several comprehensive manuals for the use of Witness, they remain generic to all possible applications and not just manufacturing. Thus the motivation for creating the manual was to build a Witness user guide that was designed for assembly line simulation, and in particular those real world assembly lines as found in Ford BEP. The manual has been disseminated within Ford with users at BEP as well as Dunton technical centre. Further to this, the manual has been used as a guide for academics, students and staff, at Cardiff University who are learning simulation.

As mentioned in Chapter 2, the use of Witness within Ford Europe is limited to using the FAST interface. Although the training manual created as part of this project is based in Witness directly, and does not reference FAST at all, it has proved a great resource to simulation engineers using FAST. The manual allows the Ford simulation engineers to gain a better insight into how Witness actually works and thus they can link the FAST inputs to the various inputs within the DES. This is especially useful when debugging and investigating a model created through the use of FAST.

The manual and associated material is included printed as an appendix and on the CD rom at the back of this thesis should the reader wish to see the models running in Witness.

3.9 Chapter Summary

A review of assembly line terminology has been undertaken and in particular the meaning of these terms and operation of elements has been matched to their representations used in simulation. The entities incorporated within a simulation have been explained fully to review the processes involved when building a simulation. In doing so the work has highlighted that if a simulation is to work as desired the modeller needs to fully understand the intricate details of each entity. When using FAST this process is made easier as steps have been designed into the process used for modelling. This allows the modeller to select particular standard functions that are modelled automatically. After reviewing the different methods for creating assembly line models it can be concluded that overall FAST is capable of modelling the facts. However, it is lacking in visual output. Building models in Witness allows much greater visual output through the use of both 2D and 3D representation. However when the model increases in size Witness models become almost unusable due to the level of complexity. This was found when modelling a large assembly line.

Chapter 4

Improving the Understanding of Simulation

Discrete Event Simulation packages are extremely powerful and keep track of a huge amount of statistics for even very short simulation runs. It is important as a simulation engineer to consider the data output and carry out thorough analysis before presenting the results to interested parties. Assembly lines are such complex systems that the issues surrounding data collection and analysis are of heightened importance. This results in models being created which are too complex for manufacturing engineers to understand yet are often over simplified and contain too many assumptions for the model to be accurate. This raises the question: is it better to have a simulation engineer trained in the operation of an assembly line or is it better to harvest the knowledge of a plant manufacturing engineer and train them in the operation of DES and the associated analysis? The following chapter discusses the model building process and its importance in the context of large complex assembly lines.

4.1 Identifying the perception of DES modelling

A simulation engineer has a very important role in not only producing a model but also advising on optimum changes or potential pitfalls with plans. However, within Ford and generally across the industry, the simulation engineer is often an external personnel and thus is only used to provide data to assist in the decision making process. Thus, it is highly important that simulation reports that are provided are easy to understand for all decision making parties and not just the simulation engineer. The use of complex models and indeed, on occasion, bespoke data analysis tools opens a gap in understanding between the simulation specialist and the manufacturing engineer. The uniqueness of the simulation tools such as FAST found only within Ford does not help in this regard.

There is currently no standard way of documenting a simulation model despite work being carried out to try and build a framework of information to be included (Sargent, 1994, 1996) (Johansson, 2010). Research has been conducted in this area and even projects undertaken to develop such literature for Ford (Dewson, 2006). However this work has not been adopted within the Ford Europe Simulation Team. As a result two different simulation engineers can produce two entirely different reports to plant, in both format and content. Consequentially there is a lack of accessibility to the results by non-simulation specialists. It is not viable to train plant side engineers in reading simulation reports since any area of statistics could be used in the production of a report as it is up to the simulation engineer to consider what is included. Further to this, the area of statistics and operational research is too large and complex to train all stakeholders handling simulation results.

Overall, current simulation reports do not satisfy the requirement of being easy to understand for the general engineer. This is a problem since research in this area stresses documentation of the validity of a model is critical in convincing the end user of the credibility, or "correctness", of a model. In general it is the individual simulation engineer's responsibility to provide reports that are accessible to all parties involved; however without any formal framework this goal is often not achieved.

It is believed that this lack of framework impacts on overall plant-level confidence in the model. During the early stages of this project there was a distinct lack of confidence in simulation within BEP and a level of scepticism on the worthiness of simulation as a whole. This was considered to be due to several key factors:

- Simulation engineers were based off site and did not have the same level of detailed knowledge of the assembly lines as plant side engineers.
- There was a lack of understanding from plant side engineers in the work involved with producing simulation.
- An overall lack of understanding in the data required for accurate simulation.
- A lack of understanding in the results output.

This agrees with the recent review undertaken comparing the opinions of the Ford simulation department against productivity departments (Soroka, 2014). The main conclusion of this paper was that there was a lack of communication to blame. The work involved in streamlining the data analysis and presentation carried out at Ford was motivated greatly by this study.

To address engineers concerns in the quality of model output from Dunton a qualitative study was undertaken. The basis for this research is the many simulation presentations formed by Ford simulation engineers at the Dunton technical centre. The presentations chosen were for presentation to Ford plants on models built within FAST by Ford Simulation team members. The process used for verifying and validating the model, higlighted in the presentation and included in the minutes have been studied. Where absent or more detail was required telephone conversations were undertaken to establish the verification and validation process used. Further to this research the FAST training manual was examined and the experience gained through an extended period of time, by the author, spent in the environment and utilising the tools is called upon. The following section contains a review of how verification and validation of a model are used within BEP. It begins with a review of the key metric used within Ford DES; Jobs per Hour.

4.2 DES model validation and verification

There are many Key Performance Indicators (KPI's) available to measure and track efficiency within industry. The main measure of production in BEP is the number of engines that are taken off the line fully assembled and fit for shipping every hour, known as the Jobs per Hour (JPH). This is a measure of throughput and reflects the performance of the line within that hour. This information is recorded by BEP foremen for each line, for each hour of production and compiled to form a 'Shift Report'. These reports are easily accessible to those both inside and outside of plant and are reviewed regularly. As a result of the wide spread use and accessibility to data, as well as the relative ease with which it can be assessed from the model, the JPH has become the primary KPI used within Ford simulation results. The Ford Simulation toolset has been built with the capability to record and output the JPH measures automatically through the use of functions. The JPH KPI is utilised not only within the final presentations but also to track model performance throughout the validation and verification processes.

Within simulation and in particular automotive manufacturing simulation there are many papers referring to varying KPIs, however JPH or throughput is the most utilised measure. There are of course many other KPIs that could track performance and efficiency of the model, many of which come as standard within the majority of DES software. These include the Work in Progress (WIP) or the average queue size (Robinson, 2007); however these are not currently implemented within Ford simulation as standard. There is also the flexibility to allow many more industry standard measures to be built using code within the DES. Overall Equipment Efficiency (OEE) is one such metric that could be considered. However, as these measures are not utilised within plant there does not seem to be any plans to implement these measures within FAST. This is no requirement as changes in the throughput of the line are the overall consideration when making changes to manufacturing facilities. An increase in OEE or availability of a machine does not guarantee any impact on the overall throughput of the line. Although additional metrics, such as OEE, may assist in explanation and presentation of results, the plant does not currently record these measures to allow for real to simulated comparisons to take place. Overall, and for these reasons, choosing the hourly throughput is a sensible metric to base simulation changes upon.

If a change is simulated and the results highlight an impact on the End of Line (EOL) JPH it is possible that, within the simulation at least, the change implemented has had an overall effect. This, of course, does not guarantee that the same change implemented in reality will influence the JPH in the same way. In order to be confident in the output from any model it is necessary to firstly assess the model's accuracy at representing the real system. This is done through the processes of verification and validation (V&V). The following section of this chapter discusses the process of V&V through a brief review of this and in particular the methodology utilised by Ford, with the aim of improving this process and the acceptance of models by engineers.

4.2.1 Model Verification

Verification is considered to be the process of confirming that the computerized model and its implementation are correct (Sargent, 1994). In the context of Ford this is not limited to within the DES model itself but also needs to include the FAST interface. Verification is an important step within the building process as errors within the model can be highlighted and corrected. In this context verification is necessary to ensure that the following minimum criteria are satisfied:

- FAST has input the correct elements in to Witness in the correct order.
- The logic behind these elements provides the correct routing of parts.
- The correct numerical inputs have transferred from FAST to Witness successfully.
- Functionality such as quality, breakdowns & stock handling is correct.

There may be many more, further detailed aspects related to individual models and to the FAST software. There is no standard methodology used within Ford for the verification of an assembly line. Often the only verification undertaken is through the method of Animation Inspection (Greasley, 2004) where by the model is run and the flow of parts through the model observed. This is in no way sufficient to confirm all of the elements are working as expected or required when modelling the complexity of an assembly line. However, many engineers put complete trust in FAST gained through long term use. This is the disadvantage of using an interface such as FAST as the building of models becomes routine and, perhaps, the efforts in verification are reduced. As a result it is not uncommon for engineers to bypass stringent verification and move straight to validation, test if the model output is correct, and if not return for more careful verification.

The use of the FAST software reduces the amount of time required in verifying a model when compared to utilising the DES software directly. This is due to the modules, or elements, pulled in to the DES through the use of FAST having already undergone testing and therefore having been subject to verification in their own right. This is often stressed as a major advantage of using such an interface when verifying large models since less stringent verification may be required. However, it should be noted that verification is still a highly important process even under these circumstances due to the uniqueness of each assembly line built. There are occasions where modules are used in ways that were not considered during initial development of the FAST interface. This has been seen where two modules, verified and working in their own rights, are not compatible when placed next to each other on an assembly line. It is this type of error that verification highlights.

The FAST toolset does have facilities implemented to aid the verification process. The inputs inside FAST that are normally entered to build the model, such as cycle times, breakdowns and stock handling for example, can be manipulated to undertake 'Static Analysis' within FAST. This static analysis is a series of mathematical calculations, undertaken automatically, that predict the JPH of the model without actually running it in Witness. The calculations undertaken are based on approximate intervals for breakdowns and many other assumptions are made. These then calculate to provide an "educated guess" as to how the model should perform in terms of its EOL JPH. The FAST model is then built within Witness and run to check if the DES output is approximately the same as that predicted in FAST. It is modeller discretion to decide how close the static analysis results should be to model output. In the authors experience a difference of no more than plus or minus five percent between the results is generally accepted and allows progression through the verification process. However, closeness of comparisons should be considered on a model by model basis and take into account the variation included in the model. If the results are not an acceptable match then further model verification is necessary, often in the form of peer review. If the numeric figures do match then it may be assumed that the DES is performing as per the inputs provided within it and FAST. It is important to note that the actual JPH figure achieved at this stage is largely irrelevant and requires validating. The research showed little to no use of the Static Analysis feature within FAST and is not favoured by the Ford simulation engineers. The lack of use of static analysis within Ford can be explained by the approach usually taken by the team in choosing to not allocate time to stringent verification. The simulation engineers, generally, bypass verification altogether and proceed to validation. If their attempts to validate the model fail, they return to undertake thorough verification. One further point, which can help explain the lack of use, is the actual JPH figures achieved through static analysis are not final. Since the model has not been validated at the verification stage, any results are not usable within presentation to plant; doing so, even with attempt to explain, could cause confusion. Thus, there is a lack of motivation for the static analysis study.

There are several alternative methods for verification available within literature (Greasley, 2004 and Sargent, 1994) which may be referred to for further detail. Some of these verification methods are not possible or feasible to implement when using an interface such as FAST. Model design verification where by the model is built incrementally is one such example. This approach would involve building a small section of the assembly line and verifying it is working correctly before continuing building. It is not normally possible to build the model in this fashion when using FAST. This is because the interface requires the model to be complete and looped such that the default logic imbedded works correctly and platens can circulate around the line. It may be possible to get around this by manipulating the way the model is built, such as building a stretch of the assembly line as seen on the layout and then artificially adding elements to link the model back on itself. This would add complication to the model building process as well as increasing the time to build a model and is thus considered inefficient.

This complexity provides some insight in to the lack of use of model design verification as no record was found in presentation or otherwise of model design verification being used by Ford Simulation. It is possible to carry out some verification within the FAST interface if a model does not run correctly. The simulation engineer can utilise the options within the command generator and turn off the breakdown, stock handling, quality, and cycle time options in order to narrow down where the issue lies, but this is not an efficient way forward due to the inherent complexity introduced.

4.2.2 Model Validation

Validation is the process of ensuring that the assumptions and simplifications made within the model building process produce a model that is representative of the real world system (Sargent, 1994). In short, a validated model is found to behave similar enough to reality for the purposes of the planned experiment. Validation is a very important step in producing any model and this is emphasised within Ford where the models are built at distance from the assembly line, often by persons who have never visited the line. This results in a greater number of assumptions being made. To allow greater ease of reading, this discussion is split in to three aspects of validation; conceptual validity, operational validity and believability (Leal, 2011).

Conceptual Validity: The purpose of conceptual validity is to determine if the model represents the real-world system effectively within the scope of the initial plan. The caveat, “within the scope of the initial plan”, is important as it is not necessary for the simulation to match real life identically for it to be valid for a particular purpose. The purpose of the model needs to be considered before building it and used as a reference when considering conceptual validity. For example, it may be perfectly acceptable to model the real world conveyors simply as buffers within a simulation model for most cases as they both have a capacity to hold a number of parts so are similar in that respect. However, if the experiment is to investigate the effect of conveyors breaking down then the buffer element may no longer be a valid option. This is what conceptual validity decides.

Within Ford the validation process is limited through the use of FAST which gives a one dimensional approach to building a model. There are many assumptions made within the FAST software that the user has no control over, the use of buffers as conveyors is one such example. Thus, both the way the model is built and the level of detail included is not within the users' control. When building a model within FAST the simulation engineer is required to keep track of any assumptions that have been made. The assumptions then form part of the final report to plant. In the documents reviewed in this study these assumptions were most often presented in bullet point form, in some cases spanning several pages. Explanation for the inclusion of these assumptions was generally omitted and they were not addressed with the assembly plant engineers until a final presentation. This is a missed opportunity since if the plant engineers are involved through this process it will highlight any invalid assumptions immediately. One example of this within Ford is the question of the need to model machining lines that feed parts directly to assembly lines. The options are not to do so, to use a black box method or through the creation of a detailed model. The plant side engineers will be able to provide information that will validate any conceptual assumptions made.

An additional bonus of engaging the engineers in the conceptual validity process is it will give them greater input and therefore provides potential to increase credibility of the model. In practice, the FAST interface has the ability to output a "Model Data Sheet". This is a matrix of all information within the model such as cycle times and buffer sizes. The model data sheet is not currently reviewed by plant as it holds an unusable amount of information. The use of FAST minimises the chances of making errors in concept through the use of the GUI, the user does not input a machine they input an automatic or manual station that is manipulated accordingly. Thus, providing those FAST components are valid themselves, conceptual validity of the overall model is often increased. Overall it was found that conceptual validity is not truly considered until the final presentation made to plant, by which time a great deal of time has been undertaken in experimentation which may be invalidated by plant engineers.

Operational Validity: Having undertaken a study of the conceptual validity, the goal of reviewing operational validity is to ensure now the data output from the model is characteristic of the real world system. Again, this validity is restricted to within the limits set in the simulation project's objectives. There are several papers (Sargent, 1996; Law, 2007; Robinson, 2007) outlining the approaches that can be undertaken to reach an operationally valid model, however it is beyond the scope of this thesis to review all of these.

Within the literature reviewed, the most common method in this stage of validation, and the one used by Ford, is the comparison of model output to real world historic data. This involves retrieving data and analysing it in a form that is comparable to the simulation output, or vice versa. As mentioned previously, the main KPI in Ford is the EOL JPH of an assembly line and this is retrieved through assembly shift reports. The current process employed by Ford is to analyse the actual JPH achieved over a three month production period and tabulate the data by shift. The simulation model is then run for a corresponding time period resulting in two data sets: the real data and simulated data. A Ford H-Chart, as shown in Figure 4.1, is then created using the Ford Result Collect Tool.

The Ford H-Chart is a visual output that is used to compare the model output, coloured red in Figure 4.1, to the actual results, coloured blue. The top graph plots the profile of both sets of results with the JPH on the x axis and weighting (i.e. the number of occurrences) on the y axis. The bottom graph shows an illustrative plot of the confidence interval of the results with the confidence interval and mean values plotted for both sets of results. This graph is known as the H-Chart, within Ford, due to the shape formed when two sets of results are considered to have "no statistical difference". That is, when the mean of one set of results falls within the confidence interval of the other, as seen in Figure 4.1. There is literature discussing the use of such an approach (Law, 2007) but the user-friendly format of the graphic has not been seen within literature. The method of comparing profiles has been discussed in (Robinson, 2007) although this concerned using histograms.

The combination of the two graphs, found in Figure 4.1 is utilised when seeking operational validity and is the main measure used to compare two sets of results, whether it be reality to a model or indeed a model to model comparison. The H-Chart is heavily relied on by Ford and if the model passes this test the operational model is said to have achieved operational validity and the experimentation process is undertaken.

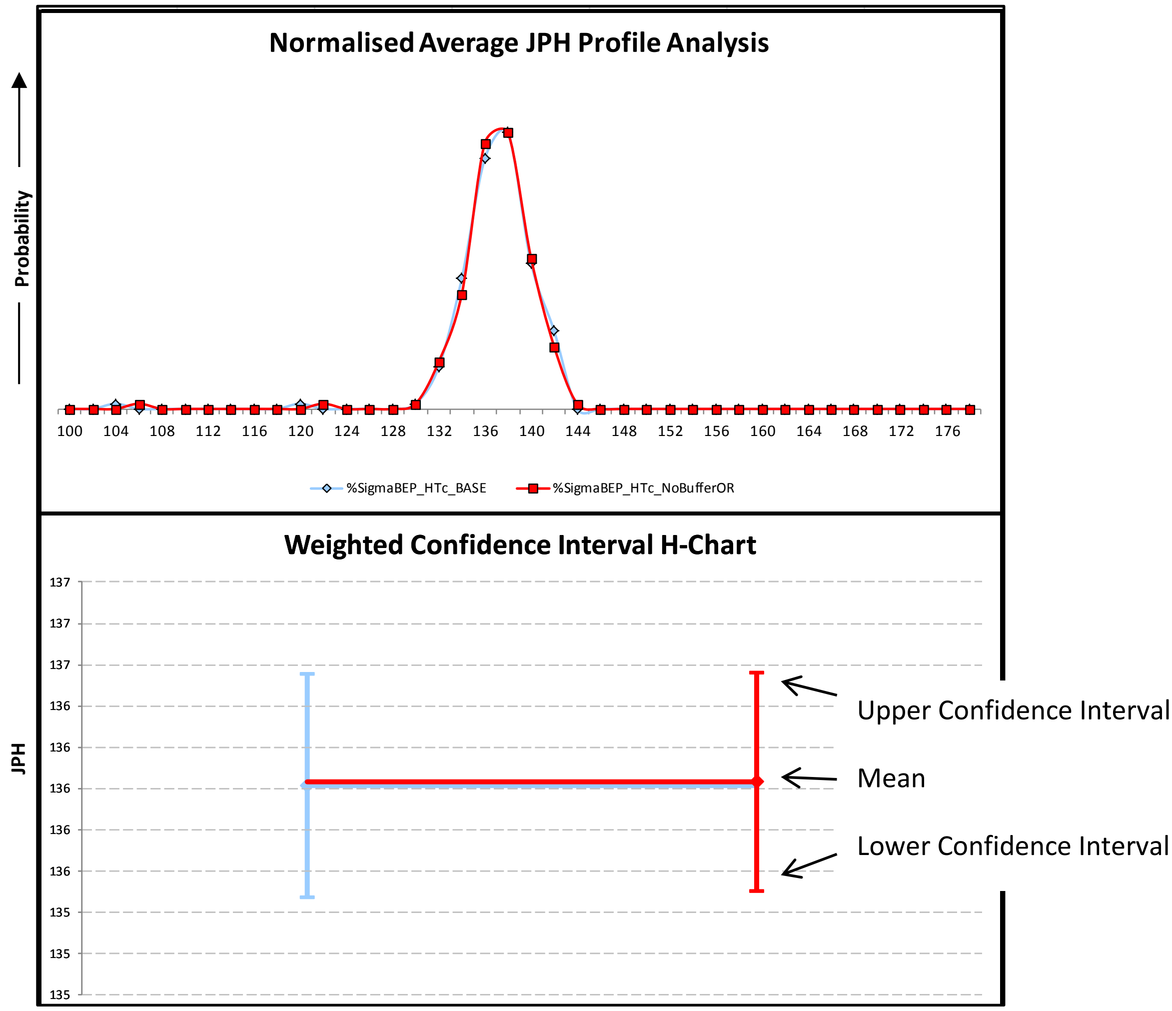


Figure 4.1 – Ford H-Chart output from the Ford Result Collect Tool.

All of the Ford Simulation reports reviewed for this chapter contained multiple H-Charts. Initially these H-Charts compared the model to actual data as a means of convincing plant management that the model was an accurate representation of reality. The presentation then went on to include further H-Charts where experimentation was carried out when model to model comparisons required validation. The problem associated with this method of validation is it only validates the model in terms of its end of line JPH value and associated variation. This is perfectly acceptable if this was the only measure under investigation as set out in the initial project plan. However, it is common practice for experimentation to be undertaken and presented on measures that lie outside of where validation is possible through use of the H-Chart. This includes carrying out detailed experimentation such as activating only one particular zone and carrying out investigation where no operational validity has been achieved. A further point on the creation of the H-

Chart is the reliance on accurate data from the shift reports. The review of current methods did not highlight any statistical analysis on these reports to remove outliers or verify the quality of the data.

The study undertaken did not reveal the use of sensitivity analysis within Ford. Sensitivity analysis is a common approach to seeking operational validity within DES, especially on simulating equipment or facilities which do not yet exist. This is something that the Ford Simulation team regularly deal with. Sensitivity analysis involves changing particular inputs on equipment between extremes to see the effect on the overall performance of a system. This could be done on a particular operations cycle time or buffer size and see if there is an effect on the EOL JPH.

This method was applied within the models created by the author and the results highlighted issues not only within the particular model but also within the FAST interface. This is presented and discussed, briefly, in Appendix 3. One example of this was discovered when carrying out sensitivity analysis on the Sigma Assembly model, shown in Figure 3.6 and in particular the quality inputs. Small changes in the First Time Through (FTT) percentage on an operation had an unprecedented effect on the EOL JPH. This was unexpected but was not impossible, however the result motivated further investigation in that particular area of the assembly line to gain a better understanding. It was noticed that the utilisation percentage of the machine where FTT was applied was also affected by a change in the FTT percentage. The increase in station utilisation was not expected since rework is not undertaken at a station level and FTT failures simply continue on through the line to be repaired at a designated area. The increased utilisation highlighted an error in the way FAST was handling the quality aspects of not only the Sigma model but all models created in the FAST interface. A discovery on the scale of this confirms a lack of sensitivity analysis undertaken within Ford.

Believability: Perhaps the most important stage when seeking validity within Ford is that of believability. Ultimately, if believability of a model is not achieved then the work becomes redundant as it is the plant team who have the power to implement changes and not the simulation engineer. The aim of the believability study is to decide if the simulation model end user, in this case BEP line management, have confidence in the model. All models created by the Ford Simulation team undertake an assessment on the believability of a model through presentation with the plant team associated with the project. In the Ford reports reviewed, most often the presenting of the model results was done through the use of an H-Chart and some additional bottleneck analysis. A discussion is then opened to confirm the model matches plant side expectations or, where there is a difference in opinion, the engineer reasons if the project is even plausible. If the plant does not believe the results then the model loses credibility. It is for this reason that it is recommended by the author that close co-operation is made between the modeller and plant management throughout the modelling process and not reserved for the final presentation. If the model matches plant experience then the engineer continues presenting further results to address the initial purpose of the simulation project.

The model is not run during this presentation although a 2D static shot is displayed to plant. Instead, it is proposed that to enhance believability the flow of parts in particular areas of importance is recorded in video format, using the tool within Witness, and included within presentations. Through experience gained by the author this method can gain believability as plant can see the simulation running and match this to what they view in reality. If they do not see what they were expecting they question it and provide feedback to the modeller.

Another factor to consider in this aspect of validation at Ford is the use of the FAST tool. The repeated use of the same tool affects the credibility of all models made; past and future. If a model has been shown as incorrect due to an error in the FAST software then this will be remembered for future models too. If modelling within Witness then the effect is lessened since all simulations start from a blank canvas.

4.3 Changes Implemented and/or Recommended by this study

In order to improve the verification and validation processes within Ford a number of changes were suggested and are summarised below. The changes have been implemented and tested when seeking verification and validation within the models created by the author. This has been found to highlight errors within the FAST software which have been corrected through, in some cases, major code and logic changes. Several new approaches have been designed to address many of the issues discussed throughout this chapter.

4.3.1 Changes to model verification

It is recommended by the author that verification is initially conducted by the Simulation Engineer without the plant's involvement. The plant staff have little to no knowledge or understanding of the details of simulation and thus can often misunderstand what information is being conveyed to them. There is little benefit gained through involving them in the actual verification process. In fact, involving the client at this stage can introduce negatives and affect credibility of the model, even if the problem is then corrected. This was observed on several occasions throughout this project. Instead the plant should be kept up to date and made aware that thorough testing is being undertaken to keep the plant staff engaged in the project.

In order to aid the verification process and increase its effectiveness when using FAST it is suggested that static analysis is carried out for all models. As discussed previously this tool is built in to the FAST interface and utilises inputs that the user must provide when building the model regardless of whether static analysis is or isn't used. As a result the use of the static analysis requires minimum investment in time and effort. Through application throughout this project, the calculations provided by static analysis have been useful and can highlight major errors within the model. The subtleties are not captured through this method however so further verification is indeed required.

The use of FAST and its prebuilt modules have benefits in streamlining the verification process. However, it is concluded that it is necessary to conduct verification on each model built due to the uniqueness of all models. In order to meet the required verification a further change to the verification process when using FAST or Witness is suggested. The following approach has been developed for use within the FAST software to provide a means of verifying the individual modules, their interaction and also the statistics recording. This is a custom adaption of model design verification which can be used to form part of a standard model verification process.

When following this verification method the model should be built within the FAST GUI, as per the layout, including all operations and buffer lengths. It is recommended, by the author, that the model should be built with false cycle times initially input to the FAST model with a flat cycle time distribution, as in the example in Table 4.1.

Table 4.1: Suggested parameters for initial FAST model verification.

FAST Input	Input Effected	Required Change
Buffer Length	Unaffected	No Change. Build as layout.
Derivatives	Batch Input	Reduce to 1 variant only.
Automatic Stations	Cycle Time	Set to TAKT time
Manual Stations	Cycle Time	Set to TAKT time
Cold Test and Hot test Stations	Cycle Time	Set to TAKT time.
Parallel operations doing the same job	Cycle Time	Balance appropriately* across the number of operations.
Line Automation e.g. Turntables	Cycle Time	Set to 50% of TAKT time
Sequence of Machines	Unaffected	No Change. Build as layout.
Variation	Quality, breakdowns & stock handling	Disable

* Here it is suggested if there are N identical machines completing the same job in parallel then each machine can run at N*TAKT and overall meet the same TAKT capacity.

This allows the simulation engineer to run the model for a short run and check two outputs quickly and easily within the Witness statistics; the EOL JPH and expected utilisation on the machines. It is necessary to run the stations at exactly TAKT time to remove varying cycle times that are seen in reality within a work standard. Having done so we expect all Auto and Manual operations on the assembly line to produce at the same rate and be the constraint of the line. Reducing the number of model derivatives to one is necessary to avoid confusion caused by part specific routing where present. A simple calculation for expected EOL JPH can be carried out manually and compared to the number of parts output from the model.

$$Expected\ EOL\ JPH = \frac{60}{TAKT\ Time\ (mins)}$$

Due to the lack of variation within the designed model a further check can be undertaken to assess the statistics are working correctly. Under the suggested circumstances all operations should be equally cycling 100 % of the time since no machine can work faster than any other and all machines are 100% available as there are no breakdowns activated. The addition of automation, e.g. turntables, running at a faster cycle time deliberately complicates this to allow for further simple checks. As the line automation is running quicker than the remainder of the line we, do not expect them to show one hundred percent utilised. In the above suggested distribution, we expect these operations to show no more than fifty percent utilisation, and the remainder of statistics for that operation should show it blocked or starved depending on their position in the line.

Using the above steps in verifying the AJ Assembly model, shown in Figure 3.4, highlighted some highly important problems with FAST. This included the FAST interface not turning off quality variation (FTT) due to a discrepancy in the code as well as a problem with the reporting of elements due to the way FAST builds models with buffers, even if they don't exist in reality. The above described method is original and not described in any literature reviewed. Appendix 4 includes more details on how this approach was used successfully within Ford BEP.

4.3.2 Changes to model Validation

There has been a great deal of success in gaining credibility for the use of FAST through the creation of the models associated with this project. This has been achieved through involving key stakeholders throughout the model building process. Thus, when requiring their 'believability' validation and assessment they have a greater understanding of the data that has been included in the model and the outputs resulting from the use of FAST. This aspect of validation has been enhanced by the presence of having a simulation engineer (the author) on site. This has allowed face to face meetings to take place where the model can be shown running live and even manipulated in real time to show the effect of changes. Although the Ford simulation team are at a remote location and face to face meetings are not always possible, it is recommended, that through the use of technology, meetings are held regularly and validity sought through the use of animation.

The use of sensitivity analysis as explained in Section 4.3.1 highlighted a major error in the way FAST was making assumptions regarding quality inputs. This caused a substantial change in the output of all FAST models, and in one particular case caused a ten percent EOL JPH change. Brief details surrounding this particular case are given in Appendix 3. It is recommended, by the author, that sensitivity analysis is carried out on every model built by the simulation engineer as model run time can be kept relatively short and highlight errors with the validity of a model.

When seeking believability validation, and especially when it is carried out remotely, it is recommended that the amount of statistical analysis and model data output included is minimised. On the presentations reviewed there are too many H-Charts used which have no proven research to establish their effectiveness. The study undertaken in this project is in agreement with the previous study that communication of results greatly affects the clients' confidence in these results (Soroka, 2010). The current graphs and statistical measures, with the exception of the H-Chart are difficult to communicate. In addition to the H-Chart, it is recommended that use of approved statistical measures such as the paired t-test or further confidence measure is implemented. It is the author's preference for the use of confidence intervals, as it assists the simulation engineer greatly when deciding if a simulated change has any true effect or is just a statistical anomaly. It could also be implemented simply within the current Ford Simulation Toolset and is a measure that can be easily explained to a non-simulation specialist.

4.3.3 General modification to modelling

There are several more general modifications that would help increase accessibility to the simulation reports or presentations. The length of the current presentation is very long, due to the unnecessary amount of data and graphs explained above. Instead of waiting for the end of the project to convey these and gain validation it is proposed that a series of more regular meetings take place to inform of progress and gain input and seek validation for approaches used. When creating the DES models for this project, meetings were taken on a weekly or fortnightly basis depending on progress made.

The terminology and graphics used within presentations or reports should be based on what is used within plant. Currently the reports refer to the names of inputs within the FAST interface; one example is using the term zones within simulations which plant refers to as teams. If not properly explained this can cause confusion or misunderstanding. On a similar note the presentations use graphs which are not used by plant staff. Instead where possible graphs should match the plant side analysis in both style and colour scheme.

It is important to note that no change to the processes implemented at Ford will result in a model that is completely valid. Box et al (2009), cited in Leal (2011, p59) summarised this by stating that all models are wrong, it is just some models are useful. This is particularly true within complex systems such as assembly lines which contain many assumptions in order to allow a model to be built.

4.4 Summary of Chapter

This chapter has reviewed the current methodologies used by Ford Simulation when verifying and validating models. From this a series of recommendations have been created and two new techniques implemented through the use of models built for this thesis. These new techniques identified some major errors within the FAST software, not limited to the models in question but within all FAST models, leading to correction and improvement of all future models created by Ford Simulation. The fact that these bugs had not been discovered prior to this point highlights the insufficiency of the verification and validation steps in place currently. An increase in the frequency of meetings provided to plant has resulted in better explanation of the model building process and a greater understanding of the potential uses of simulation as reflected in the author now being approached to work on new projects not previously completed by Ford staff.

Whilst conducted within Ford, it is clear that similar considerations would benefit the application of DES modelling wherever it is deployed. Most useful is the establishment of a verification and validation process that can engage the appropriate interested parties at the most appropriate times using appropriate levels of detail.

Future work is recommended to design a standard report/presentation format for Ford Simulation with the minimum statistical content requirements. It is also recommended that a problem statement is created to set out the goals of a simulation to determine the scope of verification and validation before model building even begins. It is recommended that a presentation format is designed, and approval sought from Ford Europe to make an official document, to enable better communication by:

- Minimising the amount of numerical figures included.
- Minimising the number of experiments included within presentations.
- Explaining in greater detail the statistical measures utilised.

- Where statistical measures are used give values required for validity. Confidence intervals are easy to calculate and explain to non-simulation specialists and should be used.

The following chapter utilises the developed approaches whilst undertaking a series of investigations to verify the default assumptions within FAST and advance the application of DES.

Chapter 5

Optimisation of Simulation Input

The aim of the work reported in this chapter was to increase the accuracy and detail of simulations by improving the representation and detail of key areas which may currently be missing or may not have been modelled effectively. Where there was difficulty in building a line, due to lack of data or otherwise, a new approach was developed and the effects of the changes to not only the model but also to the modeller were analysed.

This chapter provides an introduction to Run Length Analysis; the process undertaken to decide on the minimum length that an experiment must be run to produce confidence in the result. This is a highly important process when building models and is especially true with assembly lines where there is buffering and rework occurring, as the system takes time to reach a steady state. The use of available statistical tools will be explored briefly and the introduction of an approach, new to Ford, is discussed using a case study of the FAST Sigma Assembly Line. The model was used to analyse the effect of the calculation of buffer capacities using the FAST interface and to determine any statistical difference in the performance of the model when the true buffer capacities are used.

This chapter next explores the best way to deal with limited data when building a model. In particular the investigation concentrates on the quality metric FTT and how best to model this when data cannot be provided on a station by station basis. The Sigma Xloop FAST model is used as the case study for this experiment. Finally, the chapter considers the real effect of operators rotating on an hourly basis on the real assembly lines. An experiment that could not be conducted without consequence in reality was undertaken.

5.1 Run Length Analysis

The purpose of this section is to discuss the various approaches available for selecting warm-up and run length parameters. A brief literature review of generic modelling is provided in the context of large automotive assembly lines. This has allowed the correct approach to be determined for utilisation in the following investigation. There are several approaches to assess the required run length for a model and this depends on the individual model and the data held within it. The focus of this thesis is assembly lines which are non-terminating systems, i.e. there is no natural end point to the simulation. This is opposed to terminating systems which have a clear point to end, for example a shop closing at 5pm (Robinson, 1994). When an assembly line simulation model is first opened after being built there are no parts held within the model. As soon as the model is run parts begin to flow along the line interacting with different entities and triggering events. Of course, this means that initially the system performance is not representative of its normal state and any measured JPH is biased. This problem is known as the “problem of the initial transient” (Law, 2010). In order to deal with the problem of the initial transient it is necessary to sufficiently warm-up the model. Selecting the warm-up period can be done in several ways including graphical methods and statistical methods (Mahajan, 2004). Further methods have been proposed (Robinson, 2002) with no single method considered suited to all circumstances and models (Robinson, 2007).

The method used for the remainder of the thesis is the graphical Time Series Inspection (TSI). This involves initially running the model without any warm-up period, for a period of time. The claim made initially that only one replication is needed (Robinson, 1994) was modified by the same author to a minimum of five replications (Robinson, 2007). In other work it has been shown that one replication is in general insufficient to smooth the data sufficiently to spot the transient period (Law, 2007). When applying TSI, data is acquired and a time series is plotted for the hourly throughput (JPH). This is averaged over the number of replications, and a decision is made for the most appropriate warm-up period based on the time where the model begins to reach representative throughput. This is subjective and decided by the modeller. A particular instance of TSI, known as Welch’s Method (Law, 2007 and Robinson, 2007) uses the moving average with a particular window size to smooth the data. Other than the extended run time there is no disadvantage in running the model for longer than the minimum required warm-up period, so where uncertain a longer period of time is chosen (Robinson, 2007).

WIP can also be a useful measure and may be plotted on the same axis to represent system performance. As the assembly lines used in this model have platens circulating around the line, the WIP cannot exceed the number of platens. Once the allowable number of platens has entered then WIP settles to a constant. This means that the WIP metric is not as useful when assessing the warm-up of assembly lines using platens since it quickly settles to a constant. The reason for choosing Welch’s method is it is easy and quick to utilise and by over estimating the warm-up period we can be reasonably confident sufficient initial bias is removed.

Having decided upon the minimum time required to remove the initial transient and reach a steady state it becomes important to decide on a 'run length'. With non-terminating systems there is no internal limitation within the model for how long the model can be run for. Instead it is external factors such as demand for results and associated time constraints that motivates the study for run length. Overall, run length can be summarised as, what is the minimum time the model should be run to gain a true reflection of the system performance?

In order to minimise the run length a number of replications can be used. A replication is where the pseudo random number streams within the DES are changed such that different samples, the replications, can be taken from the same population, the model. As a general rule of thumb it is recommended in the literature to run between three and five replications (Robinson, 1994 and Robinson, 2007) and the thesis proceeds with using four replications. The run length is selected by running the model for an additional period, with the decided warm up in place, for the four replications and using a graphical method to investigate where the replications converge to a satisfactory level. Another rule of thumb is that the anticipated run length will be around ten times the analysed warm-up length (Banks, 2001) though this period must be considered on a case by case basis.

The run length analysis is best illustrated through a case study. The results are gathered and analysed to create a cumulative mean for each of the replications. This is then plotted as a time series and visually inspected to trace the behaviour of each replication. As the time goes on the four replications should converge. A measure is introduced for convergence, using Equation 5.1, which is taken from (Robinson, 2007).

$$C_i = \frac{Max(\overline{Y_{i1}}, \overline{Y_{i2}}, \overline{Y_{i3}}) - Min(\overline{Y_{i1}}, \overline{Y_{i2}}, \overline{Y_{i3}})}{Min(\overline{Y_{i1}}, \overline{Y_{i2}}, \overline{Y_{i3}})} \qquad Equation\ (5.1)$$

Where:

C_i = convergence at period i

$\overline{Y_{\iota 1}}$ = cumulative mean of output data at period *i* for replication *j*.

It is important to note that the convergence is not a measure of confidence. Confidence interval analysis is suspended until results are being analysed and not used when deciding run length analysis. In order to conclude on a suitable run length, a convergence of less than

5% that is maintained consistently is required (Robinson, 2007). In order to ease the analysis for future experiments a simple spreadsheet has been designed to automatically calculate these outputs and decrease the time required to produce useful charts.

When utilising FAST it was noted that due to the way the model is built within Witness the time that results are recorded from is not zero. During the first one day of simulation time statistics are not recorded within Witness models built with FAST. This was built in to the FAST code to reduce the amount of wasted real time that the model is run for during the initial transient period. At the time of creating FAST, during the 1990's, efficiency of computer processing was important, but today collecting statistics for this period would have no significant effect on the time to run a model. As a result of this, when a model is run without a warm-up period the first JPH output is not gained until the 1500th (1440+60) minute. As this does not change between models the results presented in this thesis consider the first output to be taken as the first hour of production, where in the simulation the time will have reached 1500 minutes. This is done consistently and thus warm up analysis and run-length analysis are unaffected. When a model is run with a warm-up greater than 0 the effect is no longer present. The only negative effect of this approach is when looking at graphs the first 24 hours of simulation time is missing, so it is not possible to see some of the initial transient period. The scale on graphs start from 0 hours for ease of reading and understanding.

Having reviewed the various approaches to warm-up and run length analysis the various measures in the context of a complicated stochastic, non-terminating model of an assembly line have been tested through case study. This has confirmed that graphical approaches of Welch's Analysis and Cumulative Replication Analysis are sufficient to determine these parameters and resolve the initial transient problem for further experiments. The case study follows.

5.2 Case Study – Warm Up and Run Length Sigma Assembly Line

As an example of the run length analysis used for this project, the required warm up and run length for the Sigma Engine Assembly Line is discussed in the form of a case study. The Sigma Line model, like all of the assembly lines modelled in this project, is a stochastic model representing a non-terminating system that eventually reaches a steady state. The model is utilised in the following chapter for a comparative experiment and so it is important the model is set-up with sufficient warm up and run length to give confidence in the results. Due to the sheer size of this model and therefore the amount of calculations being undertaken the model takes a long time to run. For example, in order to simulate 75 days on the Sigma Assembly Line model it takes a modern computer over 10 hours to process four replications. Thus, it is important to minimise the warm-up and run length whilst producing results.

5.2.1. Warm Up Period

The warm-up period was determined by considering three factors; the throughput of the model measured as the JPH, the work-in-progress and an additional visual check of the animation to confirm the point when all platens have entered the line.

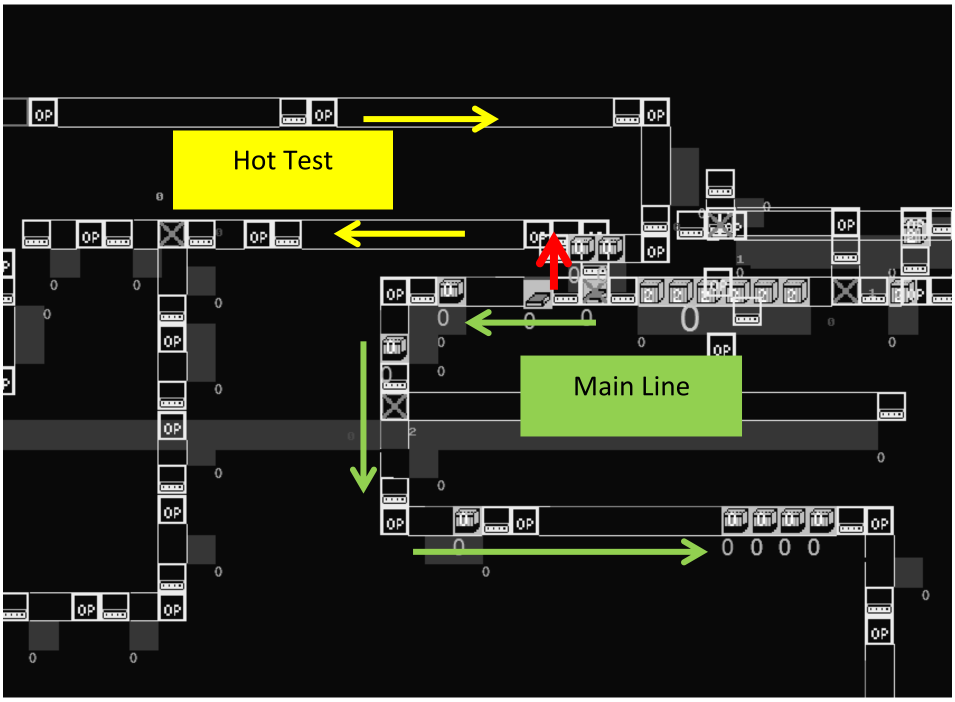


Figure 5.1: Main Line to Hot Test Transfer

There are 508 platens on the main area and 151 platens on the hot test area of the assembly line. There is a visual change observed in the model when all platens have entered the line. It takes 90 minutes of simulation time for all the main line platens to enter the line. Since the two lines are linked, as shown in Figure 5.1, for all of the hot test platens to enter the system 151 engines have to travel through the main line and reach the hot test area and be transferred to all 151 hot test platens, this takes until 116 minutes simulation time. After all of the platens have entered the line the work in progress reaches the maximum of 659 and then settles.

Figure 5.2 shows the hourly throughput over a two day period. Using the TSI method, a warm up period of 24 hours was chosen for this line and is highlighted in green on Figure 5.2. A minimum warm up period of around 5 hours may have been chosen but selecting the longer period of 24 hours increases confidence in the removal of the initial bias as well as providing a round number for presentation purposes. For comparison purposes Welch's analysis has also been conducted to smooth the output using the moving average technique for a window of 5. This is included as figure 5.3 and shows a minimum warm-up requirement, represented by the red line at around 9 hours, where the graph becomes near horizontal. This is slightly longer than that indicated through TSI, however Welch's analysis is known to be the more conservative estimate (Robinson, 2007). We can be confident that sufficient initial bias has been removed by the selected warm-up of 24 hours.

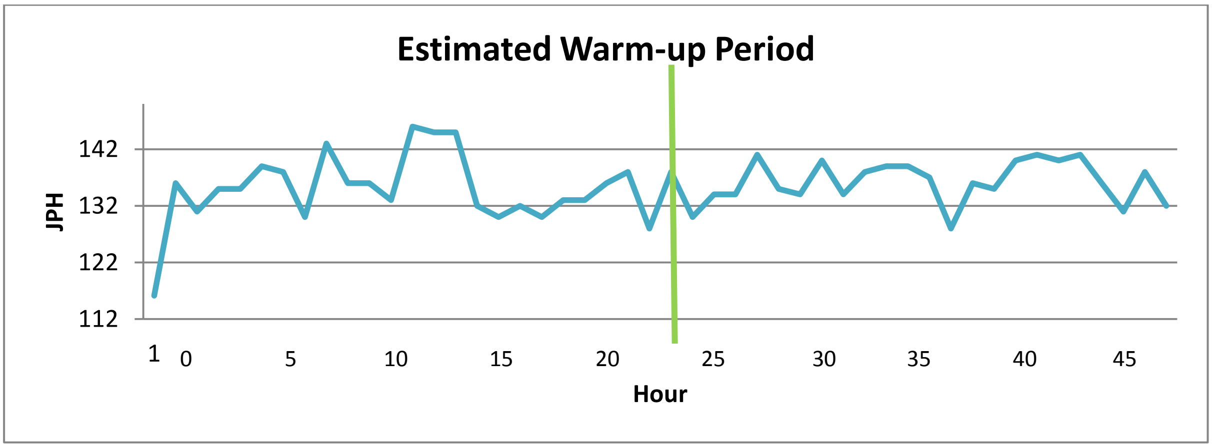


Figure 5.2: The hourly throughput over a two day period

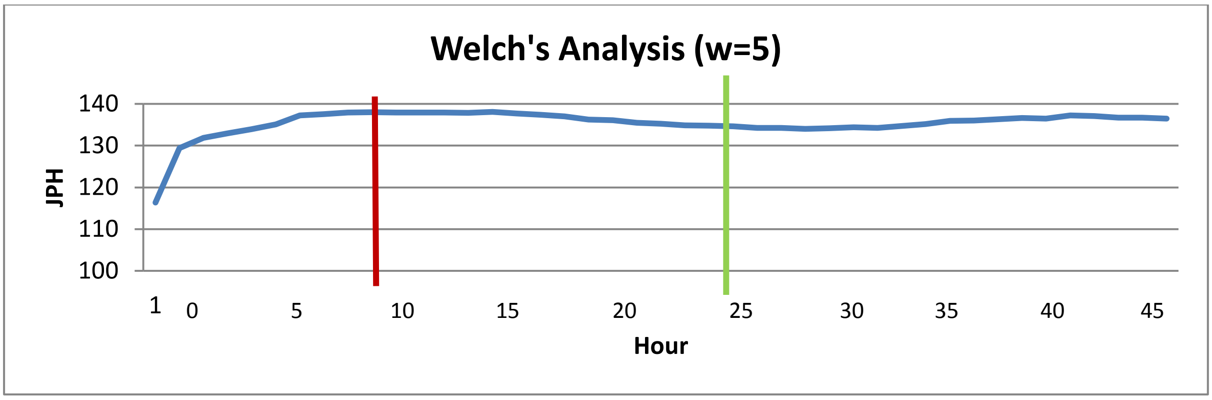


Figure 5.3: The hourly throughput using Welch's Method for a window of 5.

Due to the conservative nature of Welch's analysis and the smoother graph, further experimentation will utilise this as the main approach for deciding warm-up analysis.

5.2.3 Identifying a suitable Run length Analysis

Figure 5.4 provides plots of the cumulative mean of the four replications with the convergence also shown. A period of 7 days was decided as an approximation of where the model would begin to converge based on previous experience, however any suitably long period could have been chosen. The model was therefore run for the 1 day warm-up period, as decided above in Figure 5.3, plus an additional 6 days across four replications, each with a different pseudo random number stream. The results were gathered and the cumulative mean for each replication was calculated and compiled in Figure 5.4.

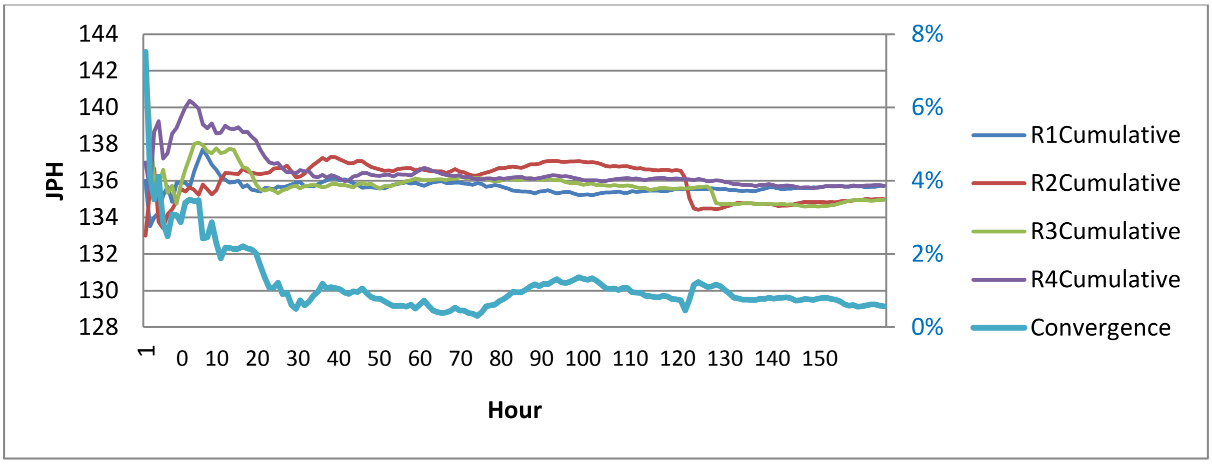


Figure 5.4: The hourly throughput over a seven day period

The plots of cumulative mean and convergence have been combined in to one graph to save space however for normal presentation purposes two separate graphs may be more accessible. The convergence of the graph falls below 5% within the first nine hours however there is still a lot of variation noted in the individual replications until around 48 hours. The graphical analysis, undertaken in Figure 5.4, points at a run length of around 48 hours being sufficient, giving convergence much less than 5% and similar behaviour across all four runs.

To avoid any doubt a longer run length would normally be chosen at the expense of waiting longer for simulations to complete, this would be decided by the modeller depending on the purpose of the experiment (comparative or otherwise) and time scale available.

5.3: The effect of assumed Buffer Capacities on a Large Assembly Line

The following investigation has been designed to assess the way buffer capacities are handled by FAST when assembly lines are modelled at distance. The motivation for this study is a lack of confidence portrayed by the plant side engineers at BEP when assembly line models are built at distance. Having considered the V+V methods in chapter 4 attention now turns to quality of data and if improving data quality results in greater model accuracy. The first experiment concentrates on buffer sizing.

Two models are built one with true buffer sizes and one with those predicted in FAST based on measurements taken from a 2D CAD layout. This study has never been previously conducted by Ford due to a lack of data. The study aims to clarify any impact of modelling buffer sizes remotely.

The remotely located simulation engineers at Ford are limited by information and have to rely on what they are provided from manufacturing plants such as BEP. Two key documents are utilised when building a model, assembly line or otherwise, in the form of 2D layout outlining the position of operations and a work standard outlining the times and processes undertaken at each station. Buffer capacities included in FAST simulation models are based wholly on the layout. This reliance on the document introduces two errors in to the simulation; the layout could be wrong or the conversion from a space indicated on the layout to a buffer capacity may be incorrect.

Due to the length of time required to build a model versus the constant changes being implemented on an assembly line there is always the possibility that any layout will become out of date very quickly. Being reliant on this layout and basing the model upon it introduces the possibility that changes may have occurred plant side. This can be resolved through good communication between the plant team and modeller although as previously discussed this is not currently the case (Soroka, 2014). Further to this a disclaimer, or assumption, can be added by the modeller that the model is based on a particular layout number, date and time. As such this is a controllable problem.

The second error based on the conversion between a distance and a capacity is the error that this study aims to investigate. When building a line within FAST the end user simply measures the distances on the layout between two operations and enters these as a distance in the interface; this is in itself a source of error as measurements could be made incorrectly or the number input incorrectly. Utilising the calculation abilities within the spread sheet, FAST then takes the distance between the two operations and calculates the buffer capacity, using Equation 5.2, by manipulating the platen length, entered separately into FAST, and the distance between operations input.

$$C_i = \frac{L_i - p}{p} \qquad \text{Equation (5.2)}$$

Where:

C_i = Capacity of buffer *i*

L_i = Length between centre of station *i* and *i+1*

p = Platen length depending on the orientation set

Although simplistic on first glance the FAST interface also factors in to the calculation that there is half of a platen held within each station when modelling centre to centre (hence the subtraction of p on the numerator), the orientation of a platen and whether there is a kit box loaded. However, there is some complication to this matter. Since a buffer can only store whole parts the buffer capacity must be an integer. As a result the calculation the capacity down to the nearest whole number. This means there can be occasions where a platen may fit in reality and a very slight discrepancy in the layout can cause the buffering to be reduced. A further issue with this approach is the calculation assumes all platens can stack up against each other such that there is no wasted space between platens. The reality is often very different as there may be kitting boxes which may or may not overlap one another and perhaps more importantly, pre stops are used on CRL to hold off oncoming platens from the following machine. This introduces an unaccounted for space in the line meaning the true buffer sizes are often less than that included in the simulation.

Although the buffer calculations are carried out as the default method for buffer sizing within FAST, there is also the ability to override the buffer capacity manually. To allow a comparative study to be undertaken the true buffer capacities for all buffers have been gathered for the Sigma assembly line. This was done by cautiously stopping parts of the assembly line temporarily to fill buffers to maximum capacity. This was no easy feat as it was necessary to avoid affecting the production and so was done only when operations were brokendown or there were parts in buffers downstream to allow production to continue. As an aside, this information has been added to the Ford 2D layout of the Sigma assembly line and provided an in-direct benefit of this investigation. During the data collection processes some spot check measurements on the lengths of the conveyors were taken to validate the layout. The overall purpose of this experimentation is to verify whether inputting true buffer capacities, gathered from line side investigation, is worthwhile and affects the model. Any one of the assembly lines could have been used for this experiment however the fast cycle times of the Sigma assembly line eased the data collection process and so was selected. Conducting the experiment on just one assembly line does not validate or invalidate the modelling approaches used for buffering but can provide the basis for further work and give confidence in stakeholders utilising the model.

5.3.1 The buffer modelling experiment.

In order to aid the understanding of the different buffer modelling approaches two models have been created in FAST; Model D, which is the base model using the **D**efault calculation within FAST, and, Model E, which is the **E**xperimental model modified to contain the buffer override capacities. Model D was created using the FAST method of measuring the layout, for this reason the layout was compared to the real line to confirm it was sufficiently accurate using spot check measurements. In order to control the external inputs to this experiment Model E was created as a copy of Model D, such that the same distributions are in both models and therefore the potential variability is equal within both models. Buffer override capacities were then added to model E. Further to this, transfer times between operations have been kept the same between the two models.

It was predicted that there could be an impact on the performance of the model due to the discrepancy in buffer sizing between the two models. It was further hypothesised that the buffer override capacities inserted into the model would make a difference to the flow of parts within the model. In particular, it was predicted that this would be noticed in areas where there was no physical buffering between two stations and the affect not captured in the standard FAST buffering calculation due to there being sufficient distance for a buffer shown on the layout. This interdependence could hinder flow through the area which would be captured by overriding the buffer capacities. Overall it was predicted the JPH at the buyoff could be affected and there would be potential for other experiments to be affected although the results would very much depend on the differences in the subtleties.

The run length analysis carried out for Model D has been presented previously as Figure 5.4. Based on the case study a warm-up of 1 day and run length of 7 days was chosen. This run length is larger than that suggested from the analysis however a longer run has no negative impact and there is no time constraint on this experiment. Similar experimentation has been conducted for Model E, the details are omitted as the process used is identical to that of the case study but for completeness Figure 5.5 depicts the cumulative mean for the four replications as a time series with the convergence on a secondary scale.

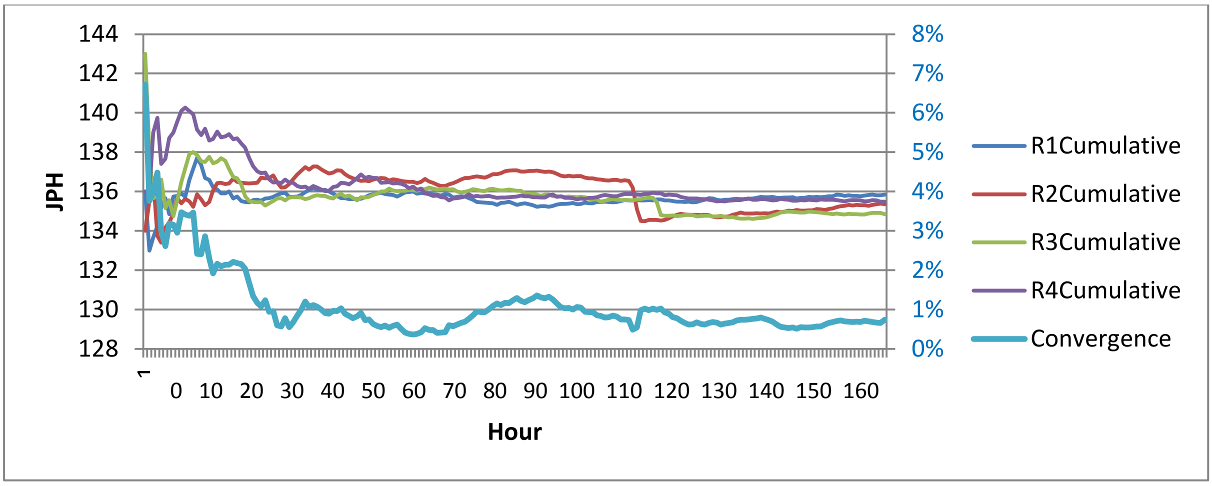


Figure 5.5: The hourly throughput for Model E over a seven day period

The analysis producing Figures 5.4 and 5.5 indicated that there was no need to run Model E under different warm-up or run lengths to that selected for Model D. To confirm, both models were run for one day warm up plus seven day run length. Four replications were used with common random numbers utilised to increase the sample size. In order to investigate the hypothesis the measure of interest was the end of line (EOL) JPH and in particular the mean over the run length. A summary of the results for the two models is given in Table 5.1.

Table 5.1: Summary of EOL JPH for Model D and Model E

	Model D JPH					Model E JPH				
	R1	**R2**	**R3**	**R4**	**Overall**	**R1**	**R2**	**R3**	**R4**	**Overall**
Mean	135.71	134.99	134.95	135.72	135.34	135.72	135.04	134.97	135.57	135.32
Min	124.00	0.00	48.00	124.00	0.00	124.00	0.00	50.00	119.00	0.00
Max	146.00	147.00	147.00	146.00	147.00	146.00	147.00	147.00	147.00	147.00
SD	4.65	13.44	8.60	5.05	7.94	4.49	13.42	8.48	5.64	8.01

Data is displayed to two decimal places to show the variation, or lack of. The results, even in table form, immediately highlight that there is no significant difference between the two models' EOL JPH. This was the case for all replications. The overall column includes the mean, min, max and standard deviation for each model averaged across the four replications. Confidence intervals have been plotted for $\alpha = 0.05$ and are included in Figure 5.6. The "mirror-like" overlap for each replication allows us to be fairly confident in stating that the results show no difference between the EOL JPH of the two models.

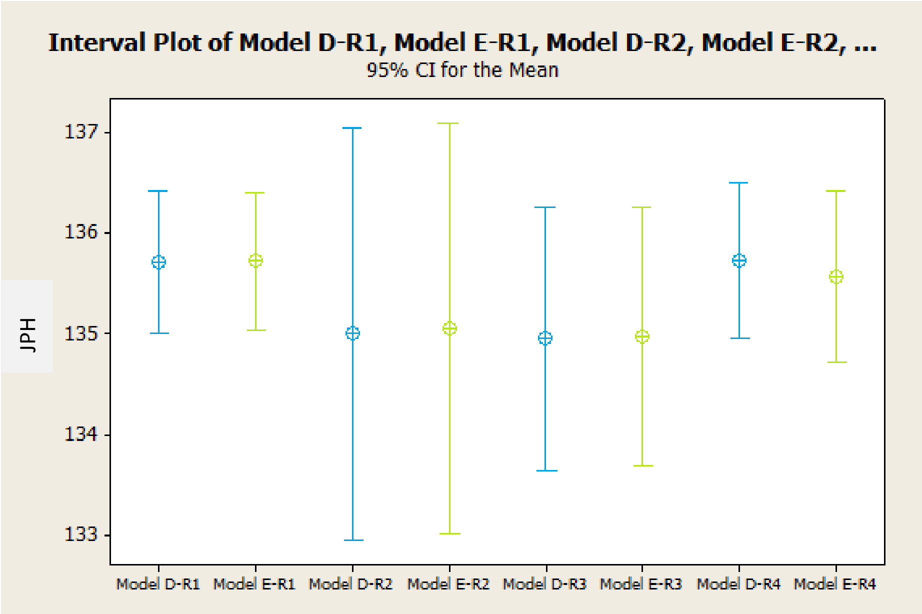


Figure 5.6: Confidence Intervals at α = 0.05

Thus, it can be concluded that the EOL JPH was not affected by the introduction of buffer override capacities on the Sigma Assembly Line. This is a key result as it shows that, for this model and under these conditions there is no effect on the EOL JPH when modelling buffer sizes using the FAST calculation. Further to this it confirms that basing the model on a layout can be a suitable substitute when it is not possible to visit the line. This result can be passed on to the stakeholders in the aim of increasing credibility and believability of models created remotely.

The results highlight that there is little sensitivity to the buffer changes implemented through using the two buffer capacity modelling approaches within a large assembly line model. It would be interesting to be able to see the differences in the total buffer capacity between Model D and Model E. However, this is not an easy task to achieve since there is currently no function to calculate the total buffer capacities within Witness or FAST. In order to achieve these total capacities with the current toolset, a huge investment of time would be required to complete the task manually and is considered of little benefit for the scope of the experiment. Instead, as further work, a macro could be created within the FAST interface to automatically calculate the total buffering on the line.

It is important to consider the limitations of this experiment. The results show that if the layout is an accurate reflection of reality, as validated through the data collection process in this investigation, then the FAST calculation and method can correctly predict buffer capacities. Some words of caution are required. The errors associated with the incorrect measuring of layouts and indeed incorrect layouts are not considered. Further to this since the Sigma line platens do not overlap and there is no complexity added through the use of pre-stops the calculation would be expected to work as the space between operations is fully occupied by parts. Further work needs to be done to establish whether other assembly lines, such as the AJ Assembly line using kit boxes show the same lack of sensitivity. The findings expressed within this subchapter provide the basis to consider the effectiveness of the calculation not only within the BEP plant but also within any manufacturing facility which contains standardised sized parts.

To summarise, the results highlight the need for accurate layouts to be created since the results contained within this chapter demonstrate that the FAST approach can be equivalent to a more involved study. Further to this point, if the layout is sufficiently accurate, the building of the model buffering can be undertaken without visiting the line. Thus the FAST approach can lead to savings in personnel time and travel expenses. The engineers' concerns over the modelling of buffer sizing has been shown to be superfluous in the case of the Sigma assembly line. The investigation now addresses a further area of concern.

5.4: The application of First Time Through (FTT) and Rework with Limited Data

Engine assembly lines contain a large number of stations where multiple processes are undertaken and hundreds of parts are fitted. As a result of this inherent complexity there is much opportunity for failure to occur. The final output capacity, or JPH, is directly affected by the amount of failures occurring within the line both in reality and within simulation models. The modelling of quality is therefore of high importance within all simulation models. The aim of this investigation is to develop a methodology to allow models to be built, for the purposes of process improvement, even under limited data which is often found in reality.

The First Time Through (FTT) Percentage for a station is a metric used to track quality for a particular process. In the simplest case the FTT is the number of good parts leaving the station divided by the total number of good parts entering the station. The FTT data is of interest to plant for general monitoring and control of quality and is often gathered within assembly plants for quality meetings to address constraints. However, due to the time and difficulty involved with monitoring failures, the data reviewed is often only from a high level view such as on a zone basis. The advent of technology has allowed data, such as FTT, to be monitored and recorded autonomously providing details of the individual fault, derivative of engine in questions and time of arrival as well as many other factors. However, this technology is expensive to install and maintain. Although new Ford assembly lines are installed with the data recording systems implemented, it is often difficult to justify the costs involved with installing/updating old lines which may have a limited life span.

FAST allows for a FTT percentage to be input on every automatic and manual station which provides the platform for accurate FTT modelling. However, when a simulation engineer requests information from plants with older assembly lines, such as BEP, data may be limited to provision of FTT percentages at the entry to repair bays, or worse, the whole line. Station level data is not available. Indeed, it is with these limitations many of the models created for this project have had to be built. The following investigation considers how this zone based FTT data should be best applied to the simulation model and if there is any difference in using station level data recorded over the same time period.

In order to test the impact of modelling with limited quality data, and in particular FTT percentages, experiments were run on the Sigma X-Loop FAST model. This model was chosen as opposed to the full line Sigma model since there is a greater amount of automated data recording in this area of the real assembly line. The recording systems in the recently built “Xloop extension”, shown in Figure 5.7 allow retrieval of FTT percentages for each individual station. The remainder of the Sigma line is not covered by these modern systems.

Although data may be available for harvesting on a station by station basis throughout parts of the assembly line, data recorded and provided from plant staff is still provided on a repair bay basis. Since two sets of data were available an investigation could be carried out to discuss the differences between FTT data on a zonal level, gathered from plant reports, or alternatively on a station level by utilising the technology available. The technology used for such purpose is the Ford Quality Monitoring System (QMS).

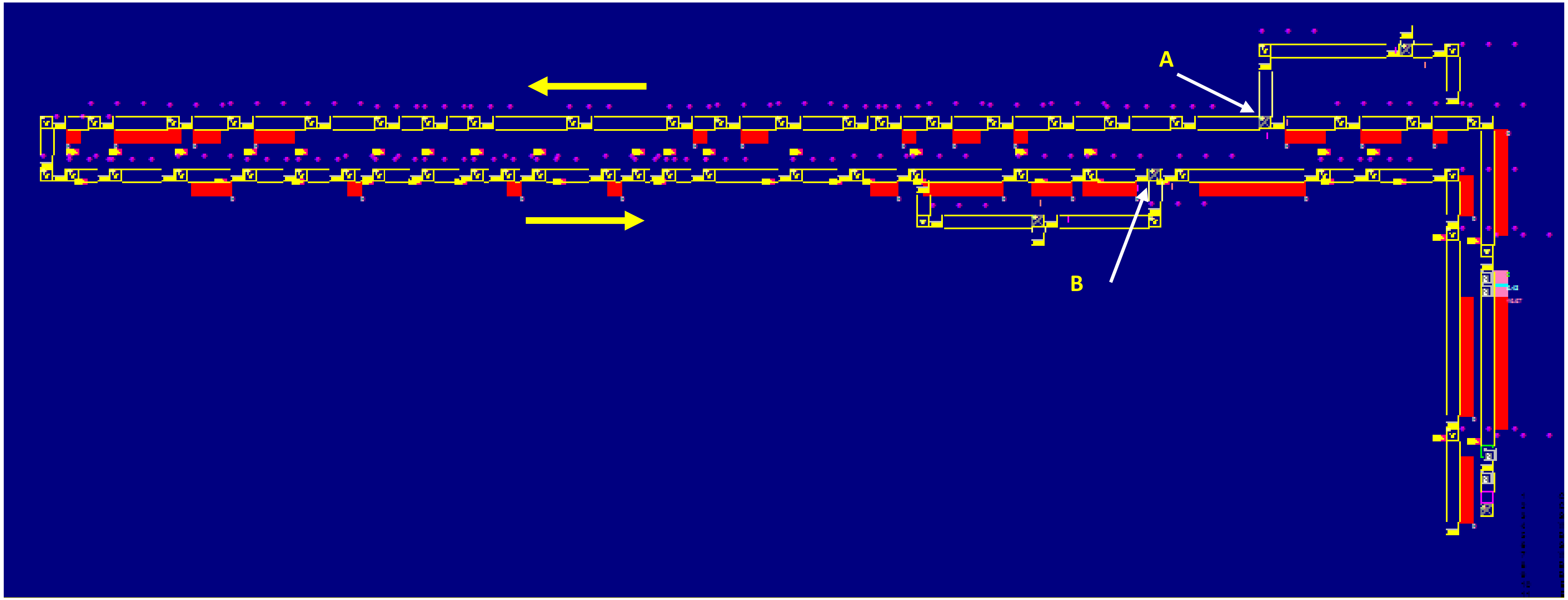


Figure 5.7: Position of Repair Loops on the Sigma XLoop Extension.

It is easiest to explain the differences in data available through use of Figure 5.7 showing the Sigma XLoop model built using FAST. Arrows A and B highlight the entry positions of the two repair bays built in this model: X1 and X2 respectively. These positions are known as data upload points as the information on the engines status is uploaded from the platen tag to a central computer at these locations. If the engine has failed any of the processes prior to this point, and therefore requires repair, the tag is read and the engine diverted in to the repair bay. Consider in particular the repair loop X2, labelled B, any one of the operations between A and B could have caused issue for the engine to require re-work. However, it is only at the point B rework can be undertaken. This effects the calculating and modelling of FTT, not to mention the disadvantage to the manufacturing efficiency itself.

It is important to consider only the number of good parts at the point of arrival, in the denominator, since parts that have already failed on an operation upstream bypass all stations until reaching a repair bay. Therefore, these parts cannot be failed again by another station and thus should not be factored into the FTT percentage. The data provided by plant does not consider the station level failures and instead compares only the number of engines that have entered the repair loop versus the number that have not. This calculation is undertaken for each repair loop on a daily basis comparing the daily buyoff score, at the end of the line, to the number of jobs carried out in the particular repair loop. The calculation used by Ford BEP is flawed for the purposes of simulation for the following reasons:

- There is no consideration for retests. The number of engines entering the repair loop may have failed and undergone a retest at a station level.
- Rework is not considered; the number of parts bought off, used in the denominator, are the total and does not consider how many of those parts themselves had previously undergone rework.
- Although the FTT percentage is calculated for the zone, the denominator uses the number of engines bought off at the end of the line and not those passing through the zone. Any stoppages between the repair loop and buy off will affect this.

The combination of all the above factors results in the FTT percentage for the zone being flawed. The limitation of the current data collection and analysis method was highlighted and the associated errors were described. There seems no impetus to change these calculations at BEP, possibly through concern that the true FTT may be different. It is to be expected that the true FTT will be lower than it is currently reported since as the number of stations with potential to fail a part increases the FTT quickly decreases.

Having introduced the actual data that is available from Ford Assembly Plants, and in particular BEP, it is necessary to consider how this can be modelled within DES effectively. The FAST interface allows a large amount of data to be modelled. Thus the modeller is limited by actual data rather than the DES. FAST allows an FTT percentage to be input for all automatic and manual stations as well as semi-automatic and hot/cold test cells. The number is simply entered within the specified box in the spreadsheet GUI – all calculations are then handled automatically within Witness. There are also further inputs that allow the modelling of retests, however this data is not available from BEP. The FTT percentage for each station is then read into a data file to allow transfer to Witness where it is held within a real variable for each station.

These FTT percentage variables are used within the DES to model the failures by comparing the percentage to pseudorandom numbers generated. If the pseudorandom number generated is greater than the FTT percentage then the engine is a fail. By default all parts, engines, within the model are created with an attribute known as 'Eject'. Initially Eject is set to 0, however, when the part 'fails' at a station the attribute is increased such that Eject=1. This attribute is used for various purposes throughout the model to base logic decisions on. With respect to modelling FTT it is used in two ways. The attribute is read at every automatic, semi-automatic and manual station and if it is 1, i.e. it is a failed engine, then the cycle time for that operation is reduced to zero such that the engine bypasses the station. Secondly, when the engine approaches the entry to a repair bay the Eject attribute is read and if it is 1 the engine will be diverted into the repair bay to be repaired. This accurately reflects the reality. A similar approach has been used when modelling directly within Witness although a more visual element has been added. This is demonstrated in Appendix C5.4 on the CD.

The constraint when modelling quality through FTT is not the limitation of DES or the method used but rather a lack of data. When data is limited the approach used has to be adjusted since data is not normally available on a station by station level, as required within FAST. If data is only available on a zone by zone basis dummy values are used with FAST such that the FTT percentages for all stations are initially set to 100%. The exception to this rule is the last operation before a repair loop which is set to the FTT percentage for that zone. Effectively, this final operation becomes the point where the Eject status is set on the engines based on the percentage provided from plant. With sufficient data it is advised that FTT percentage should be applied on each and every station along the line.

Two simple witness models have been compiled in order to illustrate: the desired approach where FTT is set on each machine, Figure 5.8 and the approach used under limited data conditions, Figure 5.9.

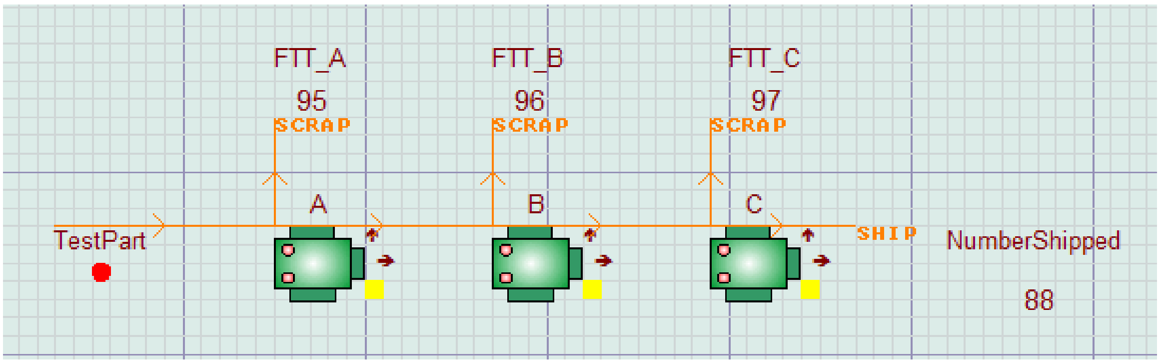


Figure 5.8: FTT applied on station level

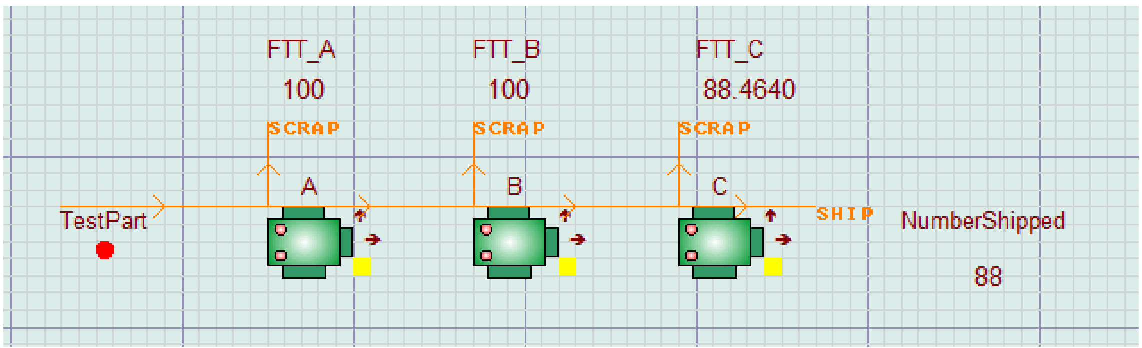


Figure 5.9: FTT applied on zonal level

The part 'TestPart' is constrained such that only 100 parts can be input into the model over the entire run and the simulation is run until all parts have entered the line. The three machines all have a 1 minute cycle time and there are no breakdowns or set ups modelled. On entering the machine the part is processed, for the 1 minute, and through the use of pseudorandom number streams a percentage of the parts pass, as per the relevant FTT for that machine displayed above it. FTT_A, FTT_B and FTT_C are variables containing the percentage FTT for the respective station A, B or C as displayed. The number shipped is recorded into a variable and displayed at the right of picture. Notice in Figure 5.9 that FTT_A and FTT_B are assigned 100% and FTT_C is applied a lower percentage than any other percentages in Figure 5.8. The pass/fail decision is considered using a random number against the percentage. Those that pass continue to the next machine whereas the failed engines are ejected from the model as scrap. This is a simplified version as with what is happening within reality and FAST models. The demonstration models were built to aid explanation; Witness was used since variables could easily be created to make the explanation more visual. This reinforces what was discussed in Chapter 4, namely that Witness is much more useful than FAST when conveying information through a model.

Figure 5.8 shows the station level approach whereby a FTT percentage is applied at each station, as discussed this requires data often not available from plant. The FTT percentages in Figure 5.8 are entirely fictional and chosen only to demonstrate the effect. Figure 5.9 shows the repair bay level approach, the FTT_C percentage has been calculated using the ‘rolled FTT’ calculations shown in Equations 5.3 and 5.4

$$n \times m_1 \rightarrow (n \times m_1) \times m_2 \rightarrow \cdots \rightarrow \big((n \times m_1) \times m_2\big) \times m_3 \ldots \times m_j \qquad Equation\ (5.3)$$

$$Overall\ (Rolled) FTT = \prod_{i=1}^{j} FTT_i \quad Equation\ (5.4)$$

Where:
n is the total number of parts entering the model
m_i is the i^{th} FTT percentage
j is the number of stations

It can be seen that within these two models there is no difference between the overall number of parts shipped when the FTT is applied at only the last operation (i.e. at the entry to the repair loop) or on individual stations. This is of course to be expected since the percentages were manipulated to match. The model does however need to run for a sufficient amount of time to allow the random numbers to tend towards the expected value. In larger models, where there is greater complexity and variability, the same experiment results are not necessarily to be seen. In reality, and in the assembly line models created, the following criteria also need to be considered which are not factored into the demonstration models.

- Machines are not 100% available and work at different cycle times
- The number of parts entering the line is not constrained to a maximum value, time is the constraint
- Failed parts bypass machines
- A failed part may have a greater probability of failing again
- Rework is undertaken on the line – in the demonstration model the failed parts are immediately scrapped from the model

The final point complicates the mathematical calculations when predicting the FTT on a zonal level in a full assembly line model. Only a percentage of the parts that reach the repair bay are scrapped, the remainder are reworked and re-enter the model. This is an additional source of variation. Further to this these parts are considered reworked and thus the numbers of good parts entering following machines are affected.

Even if the calculation could successfully be made to apply FTT on one station prior to a repair loop; it would still limit some of the potential inputs that could be modelled within the DES (or FAST). It is proposed that the following details are missed as a result of applying the FTT zonally:

- Incorrect modelling of retests: As failures are not triggered at each station it is not possible to carry out re-tests at the individual stations. In reality when a part fails in a machine, the machine can cycle a second or third time to test if it was a false negative. As retests are not modelled there may be an increase in the utilisation of the repair bay. The modelling of retests is a complex matter and is limited by the data available from assembly plants on which machines retest and for how many attempts. Operators at manual stations don’t follow a standard procedure for the amount of rework they can attempt in station.

- Failed engines do not bypass stations up to the repair bay: Since the FTT is applied on the last operation before entering the repair loop there is no chance of an engine failing an operation and bypassing all of the following stations. All engines immediately enter the repair loop when FTT is applied zonally. This means it will not cycle so it has the potential to pass through the line quickly until it reaches the repair bay.
- Incorrect group leader utilisation: The group leader is required to check any machine that has failed over 3 engines in succession. Until the group leader resets the machine, the machine is unable to work causing unscheduled downtime. The time for the group leader to respond is also variable. By placing the FTT all on one station prior to the repair bay, the probability of 3 engines failing in sequence is increased. Consider a fictional line with three operations prior to the next repair loop, for simplicity assume all three stations have an FTT of 95%. Ignoring external factors, the probability of any one of these individual stations having three failures in succession is $(1-0.95)^3$=0.0125% assuming each failure is independent of the previous. In reality this may not be true due to machines falling out of calibration but to proceed we make the assumption. If we roll the FTT of the three stations, as per Equation 5.4, the FTT on the final station would be 85.7% and the other two can be set to 100%. The probability of having three failures on this final machine would therefore be $(1-0.857)^3$=0.292%. The overall probability for any one of the three stations failing three engines in succession is less than the probability when the FTT is rolled on to one station.

Having discussed the various methods DES can be used to model FTT, attention is refocused on the data available and in particular the data available on the Sigma Xloop extension. In order to investigate the data a period of 5 days operational time was chosen. Since the Sigma line is high volume, in which each station works on around 10,000 engines per week, there was a huge amount of data recorded across even a short period. Thus, although it would be desirable to consider a longer period, a working week contains a large sample of data. The week was chosen based on discussion with plant management and was considered to reflect an 'average week' i.e. there were no new parts introduced or tested, no change in supplier, no introduction of new labour and generally no failures occurring more than considered normal.

The zonal FTT data was requested from the quality team at Bridgend and was provided in the form of daily reports. Appendix 5 shows a FTT daily report provided by the assembly plant for within the study period. Note some fields have been modified for purposes of commercial confidentiality. These daily reports were then merged to provide a weekly FTT percentage for each repair bay by calculating the mean across the 5 reports. A summary is included in Table 5.2 for the Sigma XLoop extension repair bays:

Table 5.2: FTT Percentages from Shift Reports

Loop Number	FTT Percentage (%)					
	Day 1	Day 2	Day 3	Day 4	Day 5	Mean
X1	99.12	98.70	96.39	99.12	99.05	98.48
X2	96.07	97.87	97.71	98.89	96.14	97.34

For comparison purposes, the station level data was manually retrieved from the Quality Management System (QMS) for the XLoop. The data was downloaded for the same study period of time that the plant reports were provided for. The downloaded data was then manually verified with assistance from plant staff and erroneous data eradicated. Stations reporting erroneous FTT were checked and then the data removed. Full details of the station level FTT percentages are included as Appendix 6 due to the amount of stations within this section of line. The data on a station level was then rolled using Equation 5.4 to give the rolled FTT percentage at the entry to both repair bays. For the X1 Loop the rolled FTT was 98.01% and for the X2 it was 93.50%. Comparing these values with Table 5.2 a discrepancy between the effective FTT percentages gained from the plant daily shift report and those gained from the data recording system on a station level. This discrepancy in FTT, especially in the X2 rolled FTT immediately confirmed there was an error in one of the methods. This reinforced the need to verify and validate any data provided as discussed in chapter 4. The large difference found in the X2 and not in X1 FTT is considered to be due to the way plant calculates the FTT using the number that have actually entered the repair loop rather than the true number of failures at each station. This is emphasised by the number of operations the FTT is rolled from and is best explained with a diagram. Referring to Figure 5.10 there are 2 operations between the repair loop X1 (B) and the previous bay (A) whereas between bays X1 (B) and X2 (C) there are over thirty. Note, the limited FAST graphical display is misleading as it shows all FAST inputs as an icon; even if they are not an operation. For example, between A and B there are six icons shown in the diagram, only two of which represent real operations.

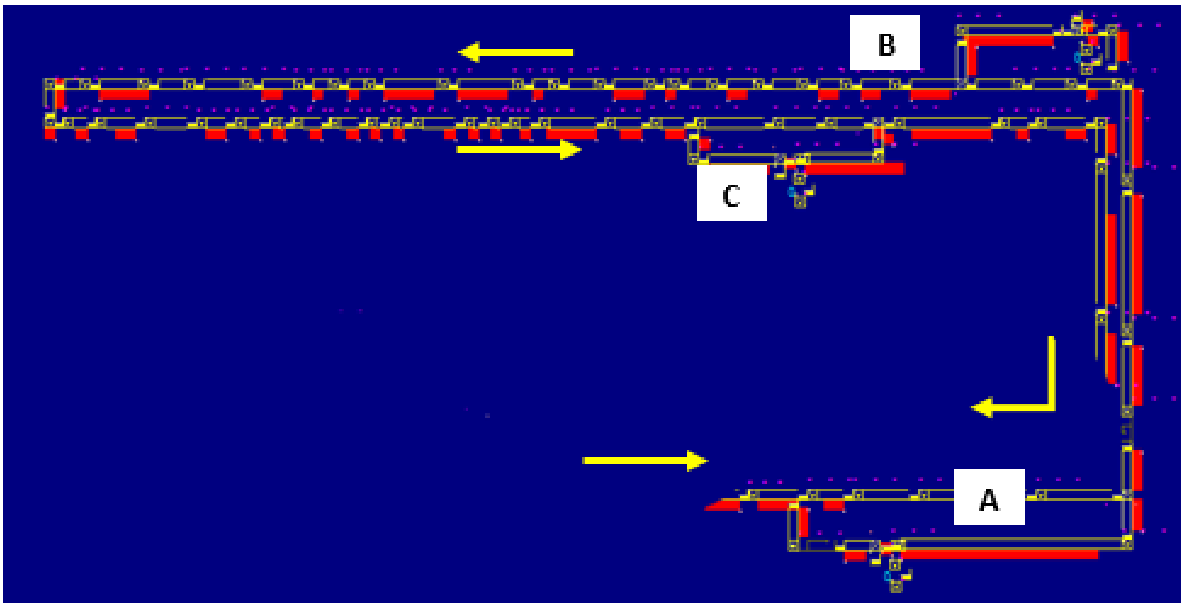


Figure 5.10: Position of repair bays

The reason for the difference can be accounted for in the drop off rate when many FTT percentages are rolled together. The plant method does not capture this effect. This reinforces the emphasis placed for the plant to reconsider their method of collecting FTT data. The question remains as to how best to deal with this situation. The plant firstly need to reconsider their approach to calculating FTT and can do so by correcting the data and calculations currently used. However, the majority of assembly lines are still going to be in a position where FTT data cannot be gathered on a station by station basis due to the limitations of the technology installed. It therefore becomes important to understand the effect of applying data on a zonal or station level within an assembly line model. Using the data collected from the Quality Management System (QMS) at BEP it is next considered if there is a difference, through use of DES, if the FTT is applied on a station level or if the FTT is rolled in to one percentage.

In order to investigate the two different approaches to FTT, two models were created from a modified version of the Sigma XLoop model. In order to ease and control the experimentation process, operations after the final repair bay have been excluded since their presence complicates matters. This is due to there being no repair bay between these final operations and the point of ship from the model, any failures after this point therefore cannot be repaired and are scrapped from the model. As indicated in Figure 5.11 the models have been shortened such that the last operation is a turntable following the X2 repair bay. All parts entering the XLoop model are good parts (not failed i.e. Eject = 0) since there is a repair bay immediately before the start of the XLoop model. Breakdowns and other variation remain unaffected by these changes and validation is not of concern for the scope of the comparative study.

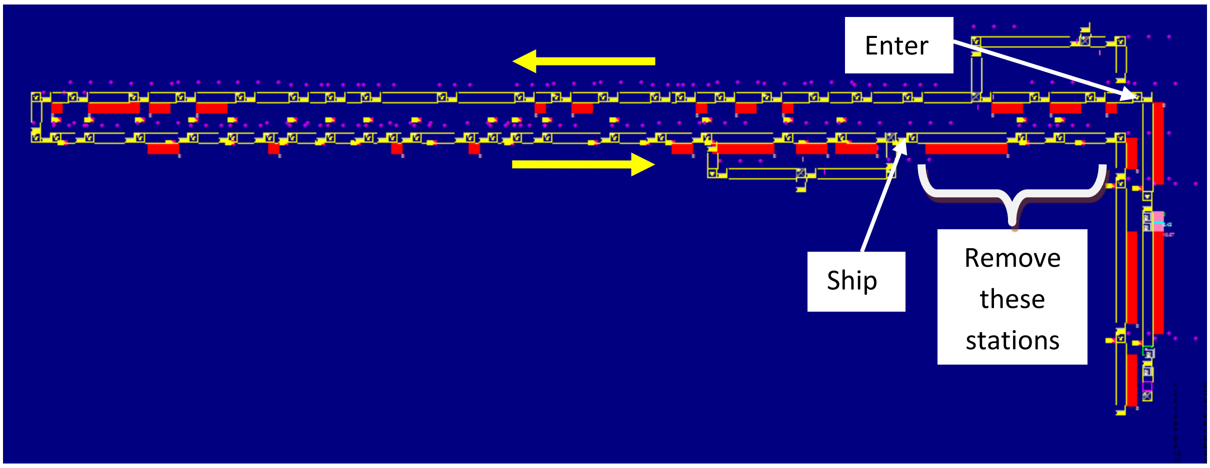


Figure 5.11: Modifications made to ease experimentation

Using the same convention as used in Chapter 5.2 we use: Model D to represent the **D**efault FAST method where FTT percentages are only applied at a zonal level due to limited data and Model E to represent the **E**xperimental method of applying FTT on a station by station approach. In order to control the external variables Model E was created as a direct copy of Model D and then only the FTT percentages modified. With the exception of FTT, this method ensures the same data is held within both models. Run Length analysis was conducted for both models using the method set out in Chapter 5.1. The thorough treatment is omitted however the graphs for Model D have been included for completeness as Figure 5.12 and Figure 5.13. Through this analysis a warm-up period and run-length was decided as 1 day and 7 days respectively, although shorter periods could have been chosen.

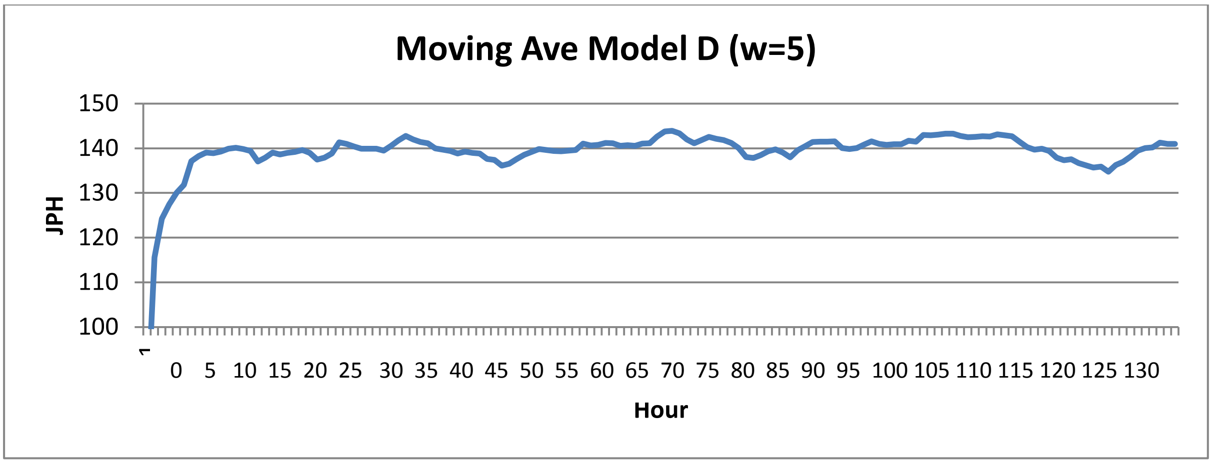


Figure 5.12: Welch's Analysis for Model D

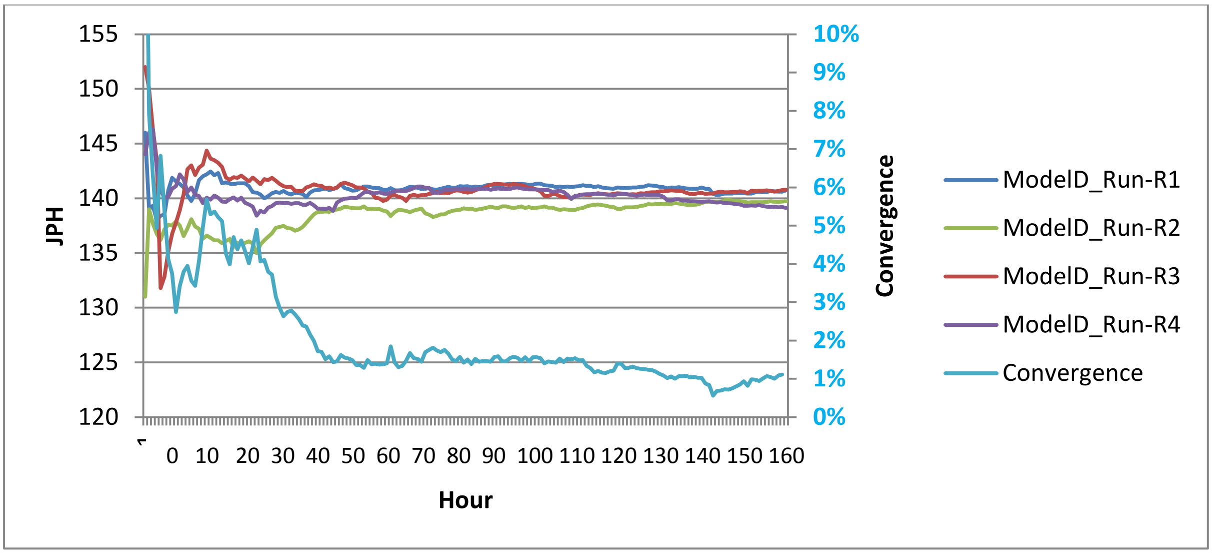


Figure 5.13: Sigma XLoop FTT test Cumulative Mean and Convergence Time Series

Model D and Model E were then run for the analysed period to produce the results given in Table 5.3. Using the method described in Robinson (2007) the paired t-confidence interval has been calculated. This is a valid approach to use since there are common random numbers in use reducing the variance. This can be seen since the sum of the individual variances $(0.65 + 0.78 = 1.43)$ is much greater than the variance of the differences (0.02) and therefore the variation has been reduced allowing the t-confidence interval to be calculated. The resulting intervals show that at 5% significance (α=0.05) the mean JPH of the two scenarios are different with Model E > Model D. The closeness of the figures to zero mean this is not a particularly substantial effect although there is some difference. The results shown in Table 5.13 confirm that the model is indeed sensitive to the different approaches used to model the FTT even when the same data is used in the two formats.

Table 5.3: Sigma XLoop zonal FTT summary of Results

Replication	Scenario 1 - Model D JPH	Scenario 2 - Model E JPH	JPH Difference (Sce1-Sce2)
1	140.72	141.63	-0.90
2	139.73	140.33	-0.60
3	140.77	141.71	-0.94
4	139.11	139.98	-0.88
Mean	140.08	140.91	-0.83
SD	0.81	0.89	0.16
Variance	0.65	0.78	0.02

Table 5.4: Paired-T Confidence Intervals

95% confidence intervals for the differences		
Lower		Upper
-1.08		-0.58

This contradicts the demonstration in Witness and confirms these models are oversimplified when compared to the real system. The effects of failed engines bypassing and retests are believed to have caused this. Further work is required to test the effects of each. Witness models of greater complexity than the previous demonstration models could be built, particularly for this purpose, such that the bypass and retest functions could be turned on or off.

The study has found that there is a difference in the FTT data achieved from the BEP assembly plant when requested on a zonal or station basis. Thus the initial aims of the study had to be manipulated since like for like data could not be gathered. This was highlighted as an inaccuracy in the calculations and methods currently used within BEP. The study proceeded to show that if the same data, reformatted, is applied on a station or zonal basis there is a slight difference in the mean JPH achieved from the model. Further work would be required to increase the applicability of this result to other assembly lines and manufacturing facilities.

Overall the work has highlighted the need for accurate data to be provided from industry. As technology adapts and more data becomes available DES can be applied in new ways that has important consequences on the modelling of metrics such as FTT. There is great emphasis on the modeller to understand the system since it is ultimately they who decide how to approach the modelling and the compromises required. The difference between ideal data and reality has been explored for the case of BEP.

5.5 Developing a Methodology to model the Variability of Operator Rotation

The following investigation is centred on the use of labour on assembly lines and in particular the monotony avoidance systems used within Ford engine plants. A new approach is developed to allow accurate modelling of operator rotation and its associated complexity. The method is then utilised in a comparative study to see the implications of using such a system.

In order to meet demand for engines, assembly lines are split up into many short tasks. This allows each station to be built up of less work content meaning each station cycle time is below or at the TAKT time for the line. The benefits of using such an approach are associated with an increase in efficiency since the jobs are built into many repeated tasks. Machines on an assembly line are unaffected by the repetition. However, humans can quickly become affected by the monotony of the task. On the Sigma engine assembly line each station undertakes the same task nearly twenty thousand times per week. The result on an operator involved with such a repetitive task has been seen to lead to a drop in motivation and resulting productivity. The modelling of human traits has previously been considered within simulation. Mason et al. (2005) undertook research to better understand the effects of Human Process Variation (HPV) on assembly lines using DES and confirmed that consideration of HPV was an important element when modelling. In the interests of health and safety and maintaining production efficiency, the operators on the line are able to move stations after a particular period of time, in BEP this is currently sixty minutes. Moving station allows the operator to undertake a different task which reduces the repetition. Operators are also given regular breaks three times per shift, these are taken as block breaks. The longest break is the lunch break which is usually 40 minutes but varies on the assembly line in question.

The Ford simulation software, FAST, does not currently model the effects of operators taking breaks. This can be justified as the main KPI of interest is the EOL JPH and modelling the break times will directly affect the EOL JPH achieved within the model since during the breaks no engines are bought off the line. As so many replications are run in the DES it is easier to remove hours containing breaks within the actual shift report data than in simulated data. This is such that the average EOL JPH the model aims to meet is only based on true scheduled uptime. This is a valid approach and saves considerable time when modelling. However, operator rotation occurs every hour and therefore the effect on JPH cannot be considered since there is no benchmark where rotation does not occur. This has motivated the question, which has been asked by plant management on several occasions, what is the true effect of operators rotating on the line?

Every one hour an alarm sounds within the plant triggering operators to move to a new station. The operator rotation is undertaken within teams and can be as simple as walking to the neighbouring station. However, there is some complexity at the edge of teams and also where specific grade operators are required at specific stations. From a simplistic viewpoint it may be considered that the jobs lost are $\frac{X}{60}$ th of the average EOL JPH, where x is the time taken to rotate. However, this is potentially oversimplified in reality. Only the operators stop working for the x minutes, automatic machinery continues to work throughout this process. If there is sufficient buffering the machines can continue to process parts and thus can have a temporary effect in continuing the movement of parts. Further to this the operators will not always take exactly x minutes to complete the rotation. There will be some level of variation to be considered, it is this variation this investigation aimed to address through the use of DES.

DES is a diverse tool that can be used to model effects, such as operator rotation, that cannot be considered in reality without consequence. The use of simulation lends itself well to this use as modelling can answer the questions and motivate future methodologies. Unfortunately, the standard Witness approach of using labour shifts does not capture the required level of detail for this task. FAST offers no control over these aspects whatsoever.

Shifts can be created within Witness including details of the working time, rest time and associated pattern over a repeating period for example a week. Witness deals with shifts through the interface shown in Figure 5.14.This information can be entered manually, although when working with complex shift patterns it is better to import them from a spreadsheet using the in-built function. Labour, machines and many more standard entities within Witness can be set to follow a shift pattern very simply. For the case of modelling operator rotation the shift would be set to the labour elements within Witness. Once the initial working time period has elapsed and the simulation time matches the first off-work interval start time then all of the labour elements disappear from the model view and are unable to work. This leaves machines allocated labour in the 'Cycle Wait Labour' state. When the break time has finished the labour team immediately reappear and start work.

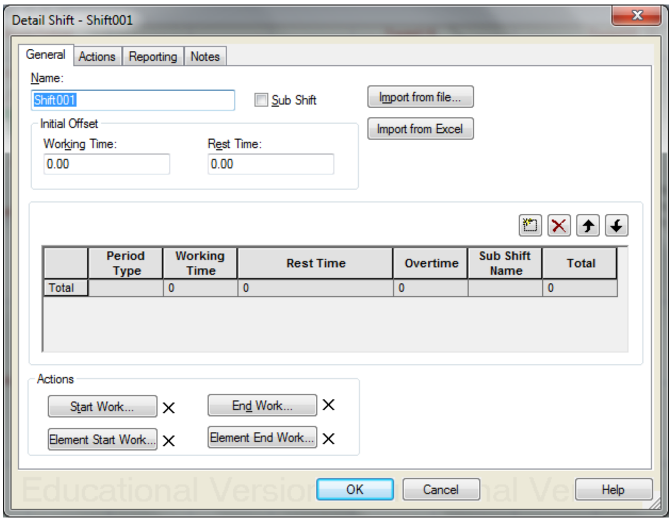


Figure 5.14 – Standard Witness Shift Inputs

In order to model the operators' availability on the line the natural choice is to consider the use of this shifts facility. This is a valid approach when considering the block breaks that operators take such as their lunch break and shorter rest breaks. The work and break intervals can be simply detailed in the witness interface and the model run. Witness then automatically handles the off-work time and monitors the statistics of the relevant labour and machines automatically. The problem with this approach when considering modelling the operator's monotony rotation is the time taken to rotate varies. It is therefore not viable to model the approach by using the witness standard shift function since operators will be assigned to work for the same length of time followed by a short rest time of exact and repeated durations. Since in reality operators are assigned the same block break intervals and are supervised, such that they are ready to work prior to the end of the rest time, the use of standard shifts for modelling the block breaks is valid without this variability.

In order to model the approach effectively it was firstly considered if the standard elements available in Witness could be modified such that variability could be added. The initial thought was to allocate a shift to the labour elements and use a distribution to represent the off time portion of the work. Being as operators rotate hourly on most lines it was considered that a 60 minute work time could be followed by a rest time of duration sampled from a standard Witness distribution. There were two issues discovered with this approach. Firstly the use of a distribution within Witness shifts is limited and the expression is evaluated to a constant and never re-sampled. This means the distribution entered does not have any variation at all. Secondly, the use of a non-standard interval between these breaks confuses the total run time. This is best explained with numbers, if the working time of sixty minutes is immediately followed by a rest time of, say, 2.5 minutes (evaluated from the distribution) a 1 day run would, as expected, contain less than twenty four full working periods due to the additional 'rest time' modelling the operator rotation. Thus the running time interval would need to be reduced to 57.5 minutes on this occasion to allow for the following rest time. As the rest time is initially evaluated from the selected distribution, albeit only once, it would need to be considered for each and every simulation run accordingly. Overall the use of the shift element does not allow for the variation associated with operator rotation to be modelled effectively.

It quickly became apparent that some alternative thinking would be required. The developed approach is perhaps abstract at first sight. However, it provides all of the functionality required to model the operator rotation, that is:

- Operators become unavailable at 60 minute intervals
- The time associated with rotating is variable
- Automatic machines are unaffected by the rotation
- The statistics are recorded correctly
- There is a suitable visual display

In order to meet these criteria a Witness machine element is used. To make sense of this it is best to consider the rotating of operators as a 'process' carried out by the team. The use of a machine is motivated by the DES elements standard ability to model a process in great detail. A 'dummy' machine is included in the model for the sole purpose of allowing the modelling of this rotation process. The use of the word dummy here is used to represent that the element is included for the purposes of simulation modelling only and does not represent a physical entity. When used in conjunction with a dummy part the arrival rate and variation can be modelled effectively as well as allowing a suitable visual output.

The details of the approach are best discussed through example by incorporating the approach into a model. The model chosen for this purpose was the AJ Head Assembly model which was built directly within Witness. Having the model built directly in Witness, without use of an interface such as FAST, allows complete freedom to demonstrate the effect. It should be noted, however, that a slightly modified approach could also be incorporated within FAST and used in other DES.

5.5.1 Overview of Operator Rotation modelling method

The method requires the insertion of two dummy elements; 1 machine and 1 part. The dummy machine models the actual rotation task but the dummy part triggers the event to occur. In order to capture this effect the part needs to be modified so that it arrives using an 'active' profile. This allows the part to directly push to the machine every 60 minutes, or at the user required interval. This approach would allow variation to be modelled in the interval between the rotations, if required, since the arrival interval could also be set to sample a distribution. As the part is a dummy part no interaction is required with any other process so the machine should push the part directly to being shipped out of the model. It is then necessary to model the rotation task time using the machine cycle time. The cycle time should be set as the time for which the operators are to be unavailable for. As required, in the initial problem statement, a distribution can be used to model the variation associated with the rotation task. The log normal distribution was picked to model the effect of the operators rotating since it can be created simply with the mean and standard deviation and still provide a suitable fit for modelling a task (Robinson, 2007). Further work could be undertaken to test the alternative distributions however.

The next step is important and should be considered carefully. Having configured the parts to arrive and the machine to process the part, labour now needs to be allocated to complete the task, the rotation. If the wrong labour is selected, such as the group leader, or not all the labour is selected the method will not work correctly. Using the dummy machines labour rule, it is necessary to allocate all of the operators that rotate, at the specified intervals, to that machine. This should be done using "AND" to link the different operators such that all operators are required to complete the task.

At this stage, although it may be initially considered as functional, further details are required to increase the effectiveness of this model and ensure it works under all conditions. It is necessary to activate the pre-empt function on the machine in order to model the 'drop tools' attitude when operators rotate. This feature can be modelled in great detail by using the standard functions of the machine. Firstly, set the priority of the dummy machine to be the highest in the whole model. This ensures when the labour are called they immediately leave their current station to tend to the higher priority task - rotating. Additional variation can be modelled including time penalty, which also can be inserted as a distribution. This models the fact that there may be an increase in the next cycle time undertaken by the operator due to getting ready to work and deciding the state where the job was left off in. The model will then work correctly and models the associated variation.

A step by step guide to building this approach has been developed and included as Appendix 7. Using Witness the visual display can be enhanced to aid verification and validation. In the AJ Head Assembly model the following further modifications to the display were made as shown in Figure 5.15:

- a string variable was added which was updated to give the state of the operators at any time
- icon of the dummy machine was changed to avoid confusing with non-dummy machines
- the icon of the operator moves from being at the station to near the dummy machine

The reader is invited to view the video included on the CD to see the true visual output.

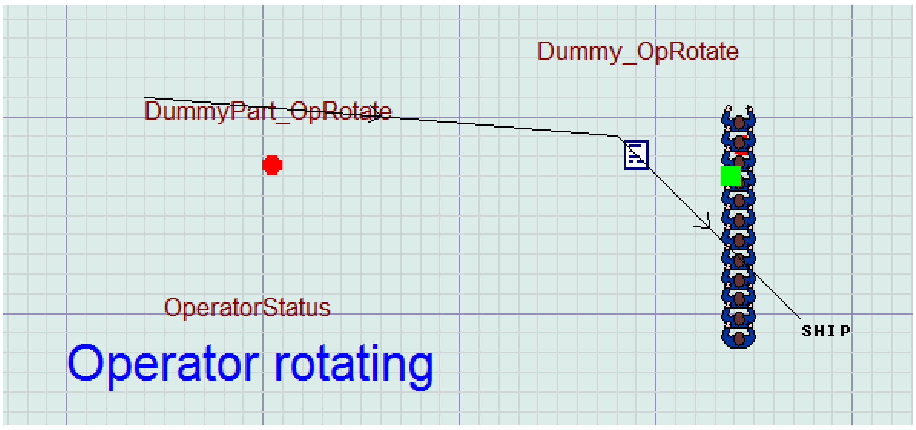


Figure 5.15: Visual Display for Operator Rotation

The method developed has undergone stringent testing through the validation and verification processes discussed in Chapter 4. Although it was known that the operator rotation contained a level of variation, there was a lack of data available at BEP. In order to input the parameters associated with the rotation, such as the 'cycle time' and the time penalty, a study was undertaken in conjunction with the industrial engineering team. Due to constraints on what can and can't be timed within Ford, limited data was collected and so the investigation was based on observation rather than time study. As a result of this study the following parameters have been built within the AJ Head Assembly Model:

Time to rotate (cycle time on the machine) = LogNormal(0.5, 1)

Time penalty = Triangle(0.1,0.15,0.25)

Allowance = 0

The collection of data and generation of distributions was not the primary purpose of this but rather to develop a means for modelling the rotation process. Further work is required to consider the various parameter inputs, those included above are just a sample for the AJ Head assembly line. The time to rotate was modelled with a log normal with mean 0.5 minutes and a standard deviation of 1. This was the best distribution that could be included with the limited data. However, lognormal is an approved distribution for modelling variation in the time to complete a task, the task in this case just so happens to be operator rotation (Robinson, 2007).

5.5.2 Limitations and benefits of the Operator Rotation Modelling Method

The methods developed for modelling operator rotation is valid for use to investigate the effect on JPH, however there are some limitations that should be considered when using this approach. These are considered below:

- Wrong state recorded on labour: During the operator rotation process the labour on the line is unavailable to work and would be considered non-scheduled downtime or miscellaneous. However, when using the approach developed within this chapter the labour are called to work on the dummy machine which models their rotation. This results in their state being recorded as working which is incorrect. However, the manual stations the operators have rotated from correctly record the statistics as waiting labour, therefore the utilisation of the labour is recorded in the machines representing the manual stations.
- Care must be taken when setting the pre-empt level and priority: When using a dummy machine to call labour at particular time intervals it is highly important that the pre-empt level is set such that the labour is pulled immediately away from their current task. If the pre-empt level is set too low then the method will fail. In order to ensure this method works the priority of the dummy machine must be set as the highest in the line and set the pre-empt level low. Visual animation inspection does however quickly spot errors in this method.
- The method is sensitive to the labour allocated: If the incorrect labour is selected or excluded from the labour rule of the dummy machine then the method will not work correctly. It is not possible to provide verification during the labour allocation process since it is at the modeller's discretion.
- All labour are pulled off the line for the same amount of time: Within the method described all of the labour are allocated to one dummy machine. This results in all labour being unavailable for the same amount of time due to cycle time and taking the same amount of time to get back to work (pre-empt). The method could be adapted to include an individual dummy machine per operator. This adds complexity but allows greater variation.

The developed method was used to model the process of operator rotation and the inherent variation associated with the task. The standard Witness method of using shifts was considered, however, it was found impossible to capture the variation required. The solution developed highlights the need for deep understanding both of the DES and its various elements, as well as the process in reality. The developed method is not limited to the modelling of operator rotation. The same approach can be used to model different shifts for different areas of the line and automatically switch shifts using the part trigger. This information is considered as future work.

An investigation using this developed methodology is now undertaken to quantify the effects of operator rotation on the AJ Assembly line to better understand the impact on manufacturing lines of operator rotation.

5.6 The Effects of Operator Rotation

As previously mentioned, the FAST software does not model the operator rotation on the line, thus Ford has never simulated operator rotation. The following investigation was conducted, using the developed approach within Witness, to quantify the effects on JPH of the operator rotation on the AJ Cylinder Head assembly line. The information was then provided back to Ford to inform of the effect of operator rotation, as well as allowing the simulation team to re-consider their current modelling approach in FAST. Two versions of the model were created. The models are named as per previous investigations, with Model D being the **D**efault (standard) current approach, without rotation and model E being the **E**xperimental model, containing the operator rotation. An overview of the experiment is included below; details on warm-up and run-length analysis are kept to a minimum to avoid repetition.

5.6.1 Warm-up and Run Length

Welch's analysis was undertaken to establish the minimum warm-up period for both Model D and Model E and was initially run for a period of seven days with four replications. A moving average of window five (w=5) was required to smooth the data sufficiently to allow inspection of the time series. Model E Welch Analysis is included in Figure 5.16.

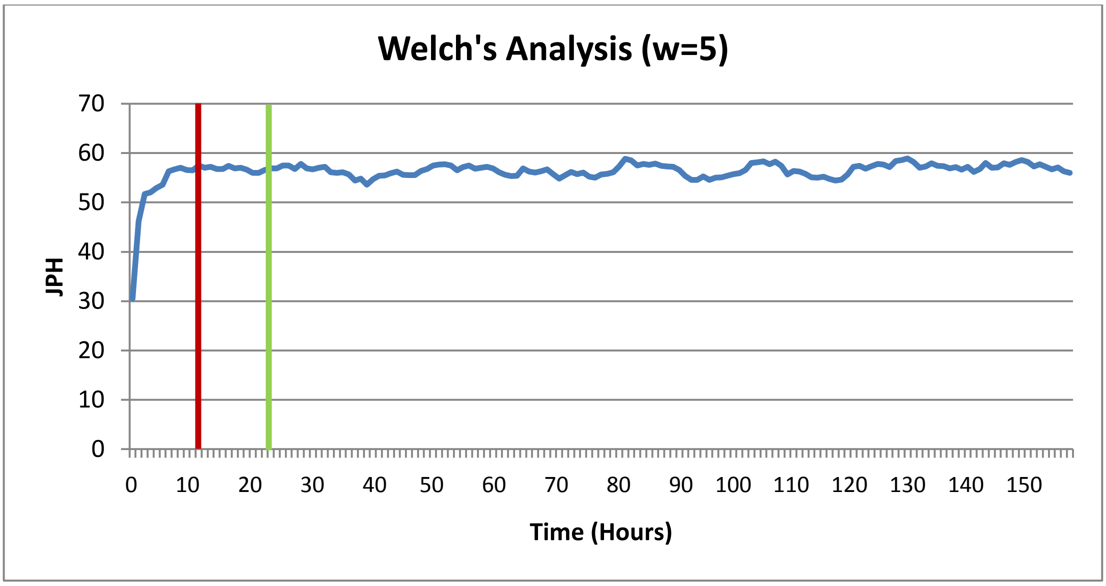


Figure 5.16: Welch's Warm-Up Analysis

The TSI in Figure 5.16 highlights a period of around 13 hours (indicated with the red line) as sufficient to remove the initial transient bias. However, as previously discussed, a longer period can be chosen if time permits and so the warm-up period was set at twenty-four hours for Model E (shown with the green line). The analysis was then repeated for Model D, which showed less variation highlighting a warm-up period of 5 hours being sufficient. This result would be expected since Model E samples a greater amount of random numbers to model the operator rotation. For ease of experimentation both models are run for the same 24 hour warm-up period.

Run length analysis was then carried out for Model E using four replications. The cumulative means for each replication were calculated and then plotted, as seen in Figure 5.17.

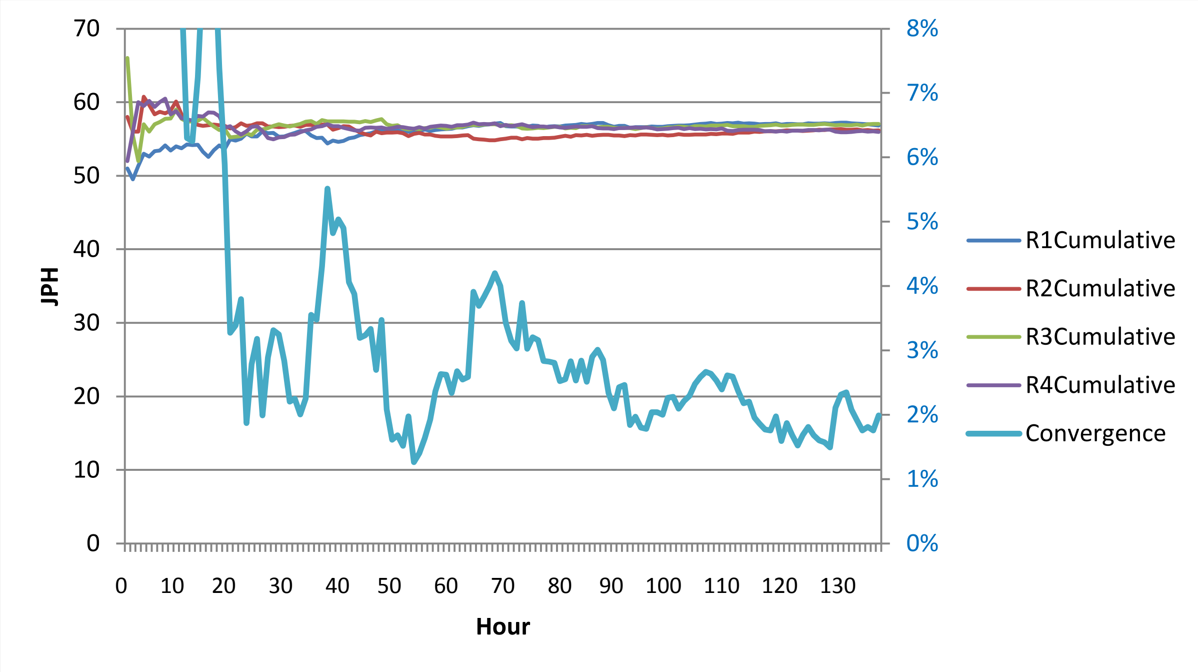


Figure 5.17: Cumulative Mean and Convergence Time Series

The time series is initially sporadic but settles below the recommended minimum level of convergence at around 48 hours. Figure 5.17 highlights the need for caution when undertaking run length analysis as although the graph initially drops below the 5% convergence limit, at around 20 hours, it does not remain there until after 48 hours. A run length of 10 days was decided to base further investigation on since the model is quick to run and the increased run length will increase confidence in the results. As an aside, although not required in this particular instance the 10 day run also satisfies the general rule of thumb recommended by Banks (2001) since it is ten times the initial warm-up period.

5.6.2 Operator Rotation Results

Model D and Model E were then both run for four replications for the 1 day warm-up plus the 10 day run. The JPH measure was used to track the performance of both of the models across the four replications. The replications were performed with common random numbers across Model D and Model E to allow the paired-t confidence intervals to be calculated, as outlined by Robinson (2007). In order to do this the results from all replications were imported into a spreadsheet, Table 5.5 provides a summary.

Table 5.5 Summary of Operator Rotation Simulation Results

Replication	Scenario 1 - Model D JPH	Scenario 2 - Model E JPH	JPH Difference (Sce1-Sce2)
1	57.89	57.19	0.70
2	56.70	56.41	0.29
3	57.18	56.63	0.55
4	57.22	56.42	0.80
Mean	57.25	56.66	0.59
SD	0.49	0.37	0.22
Variance	0.24	0.13	0.05

The data in Table 5.5 was then assessed to meet the conditions of reducing variation, required for the paired t-confidence interval test. Since, 0.24 + 0.13 > 0.05 the criteria is met as variance is reduced through the use of the common random numbers. The method outlined in (Robinson, 2007) was used and paired t-confidence intervals have been calculated at 95% confidence (α=0.05) as shown in Table 5.6.

Table 5.6: Paired-T Confidence Intervals

95% confidence intervals for the differences		
Lower		Upper
0.23		0.94

Since both intervals are greater than zero, the results show that at 95% confidence the mean EOL JPH, across the four replications, in Model D is greater than Model E. The small deviations from zero in Table 5.6 suggest that the change is small, although the result is still significant. Naturally, it would be anticipated for Model D to perform better than Model E as the operators are always available to work since there is no rotation modelled.

Through consideration of some novel values it is possible to gain better understanding and motivate future work. Considering the JPH in Model D the base value for EOL performance on this particular assembly line would be 57.25 JPH as in Table 5.5. The log normal distribution was populated with a mean of 1.0 minute, and as an assumption the operators were considered to take 1 minute to rotate. Using the calculation, previously discussed, when the operators rotate it may be surmised that the loss in JPH would be approximately

$\frac{1}{60}th$ of the EOL. Applying this would give an expected JPH of approximately 56.30 JPH. At 95% confidence the mean in Model E is greater than this. This may suggest that the machines on the line and the use of buffering reduces the expected loss through operator rotation. The figures used in the above paragraph are novel and further work is required to confirm.

The study set out to determine the losses associated with the monotony reduction systems in place by modelling the approach within DES. A better understanding of the requirements in modelling such variation is addressed. Whilst this study did not confirm an exact JPH loss that could be applied generally across all assembly lines, the work conducted shows that there is a loss in EOL JPH resulting from the rotation process.

Several limitations to this pilot study need to be acknowledged. The study has been conducted on just one assembly line at one assembly plant and thus there is a lack of generality. However, the development of the novel modelling approach provides a platform for further work to be conducted. Considerably more work will need to be done to determine the best distributions to represent operator rotation; limited availability of data will mean an empirical study is required. Once this is established, work can be conducted to consider the losses associated with the current rotation process as well as, through the use of DES, to consider the optimum rotation strategy.

5.7 Summary of the Chapter

The chapter has investigated three key areas of manufacturing simulation; buffer sizing, quality and variation; and operator rotation. Through individual studies examining different elements, the overall aim was to better understand the particular manufacturing systems and through this understanding advance the application of DES.

Section 5.3 highlighted the need for better communication between plant staff and simulation staff since it was confirmed that, in the case studied, there was no resulting inaccuracy from modelling buffer sizes on an accurate layout. Section 5.4 confirmed the need for better data provision since the toolset is capable of handling detailed inputs and there is sensitivity to the accuracy of data input. Finally Section 5.5 developed a novel approach to modelling the variation associated with operator rotation which motivates further study in this area.

Overall, although advances have been made in applying DES moving simulation closer to emulation through development of new and better modelling processes, there was little impact on the overall results achieved by the model. For example, the effect of differences in buffer capacities between those predicted in FAST and true buffer sizing did not affect the EOL JPH. Section 5.3 showed only a 1% difference between the two approaches developed for applying FTT when limited by data. In Section 5.5, the introduction of the operator rotation made only a 1.7% difference to the EOL JPH. Thus the study suggests that DES is already a capable tool for producing models in the scope of high level studies examining the EOL JPH. The lack of confidence displayed by plant side engineers due to the lack of detail examined when modelled at distance does not greatly affect the EOL JPH they consider. Thus doubt should not be placed in the modelling tool itself when modelling at this high level. Instead, the tool should be considered as relatively accurate for this use, and the attention turned to the inputs used by the modeller where inaccuracies can be formed by a lack of understanding of the manufacturing systems, and therefore incorrect data input, or indeed insufficient and inaccurate data from facilities provided.

The current study has only examined a small selection of the inputs and modelling approaches possible within DES. Further work might explore different aspects of assembly line simulation. It is also required to examine the subtleties lost through the different approaches to further address the plant side engineers' concerns over the level of detail modelled.

Chapter 6

A Case Study: Optimisation of an Engine Assembly Line using DES

6.1 Introduction to Case Study

The following study is included to convey the successful application of DES within BEP on the AJ assembly line resulting in increased efficiency. The inclusion of the case study is to outline the application of the knowledge and methodologies developed and used throughout this thesis. Due to the sensitivity of information some data and graphics have been concealed. Work within this chapter concentrates on the application of DES and the resulting improvements; thus details of model building are kept to a minimum.

6.2 The Model

As discussed in Section 3.4, the AJ assembly line is a batch model assembly line building V-configuration engines for Jaguar Land Rover. The engine assembly process is particularly complex and the line runs a 'bow-tie' cycle time distribution with a, relative to the other lines at BEP, low hourly throughput. (The 'bow-tie' distribution of cycle times is a Ford Motor theory, not seen in literature, whereby the majority of zones are run at cycle times below TAKT and constrained by one zone, the 'knot', which is run at TAKT time. The use of the distribution is motivated by the push and pull interactions from zones adjacent to the knot.) The engines are built on platens and several reject spurs are used, as opposed to repair loops. This allows defective engines, and their platen, to be removed from the line for repair at a separate location. All of these details were modelled. The POSmon and QMS systems allowed access to data for breakdown information, and therefore enabled the creation of empirical breakdown and repair distributions, and the capture of FTT data on a station level. Details of stock handling and cycle times at each station were gained from a reviewed work standard document and a layout used to orientate and sequence the operations correctly. Buffer capacities were confirmed through examination of the actual assembly line since there is some complication through the use of pre-stops and kit boxes in different areas of the line. A time study was undertaken to confirm the automatic operation cycle times and processes.

FAST was utilised to facilitate a quick model build and there was sufficient data available to populate the FAST interface. To compensate for the loss of visual output, during the model development process smaller Witness models were built to capture specific areas of detail. The initial goal of the project was to assess if there was room for improvement on the assembly line to meet an anticipated rise in demand for the product. Thus, there was motivation among all involved reflected in the amount and accuracy of data gathered. Throughout the data gathering and model building period, which were completed in parallel, the engineering staff were engaged through several meetings used to convey the modelling process being used and the data input.

6.3 Verification and Validation

Although limited through the use of FAST, the model was verified whilst building the line and a full study undertaken on completion, as described in Chapter 4. Having verified the model was working as expected as per the inputs in FAST, the model was then validated and plant staff involved. Conceptual validation of the repair spurs was undertaken and refined the modelling approach used. Animation inspection and believability were assessed with the assistance of process engineers and maintenance staff who confirmed, after minor tweaks, the model was believable.

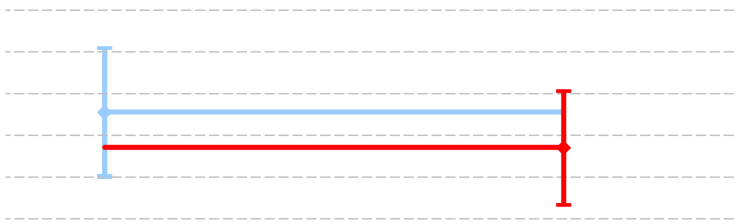


Figure 6.1: Ford H-Chart

The Ford H-Chart was used to visually convey the JPH data comparatively to the real data achieved through a series of shift reports. Figure 6.1, shows the actual data, red, versus the model output data, blue. Confidence intervals were plotted and the measure explained to the team. Using the H-Chart, the line was validated through the use of EOL JPH and was considered representative of the real system.

6.4 Warm-Up and Run Length Analysis

The method used for undertaking warm-up and run length analysis is almost identical to that considered in Ssection 5.2; to avoid repetition details are kept to a minimum. However, using the batch method described elsewhere (Robinson, 2007) the model was analysed over periods of 480 minutes, to coincide with the length of a production shift on the AJ assembly line. The average JPH over each of these periods was considered, as opposed to the total shift throughput, to allow comparative measure to be made against Ford shift reports. The x-axis in Figure 6.2 is therefore modified to reflect the number of shifts rather than the number of hours. Four replications were used and the model was initially run for a period of 50 shifts with no warm-up period set.

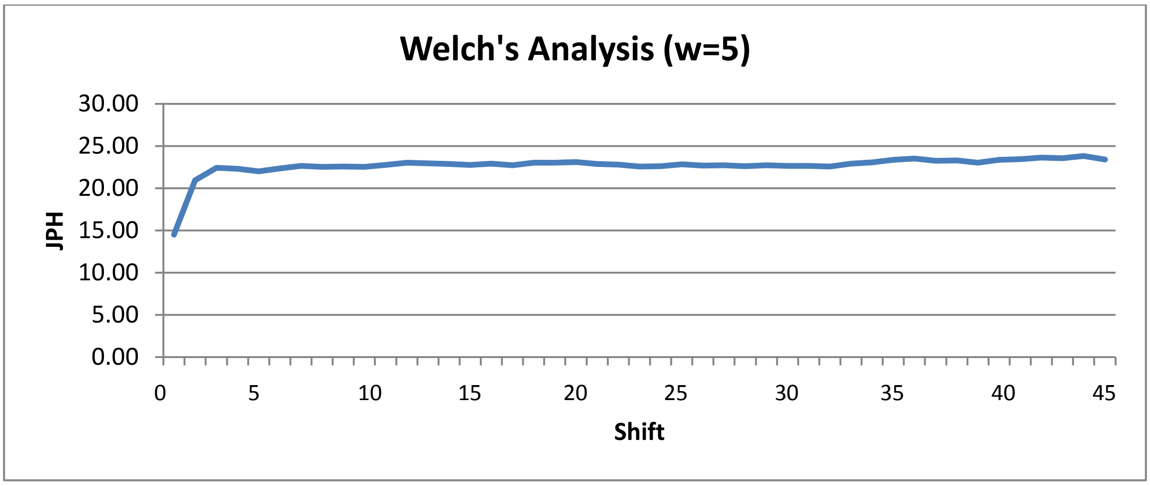


Figure 6.2: Welch's Analysis for Warm-up Period

Analysis of Figure 6.2 highlighted the requirement of a warm-up period of 3 shifts of simulation time. In order to determine the minimum run length the model was run for 500 shifts with a 3 shift (1 day) warm-up used. The resulting time series of cumulative means and convergence is included as Figure 6.3.

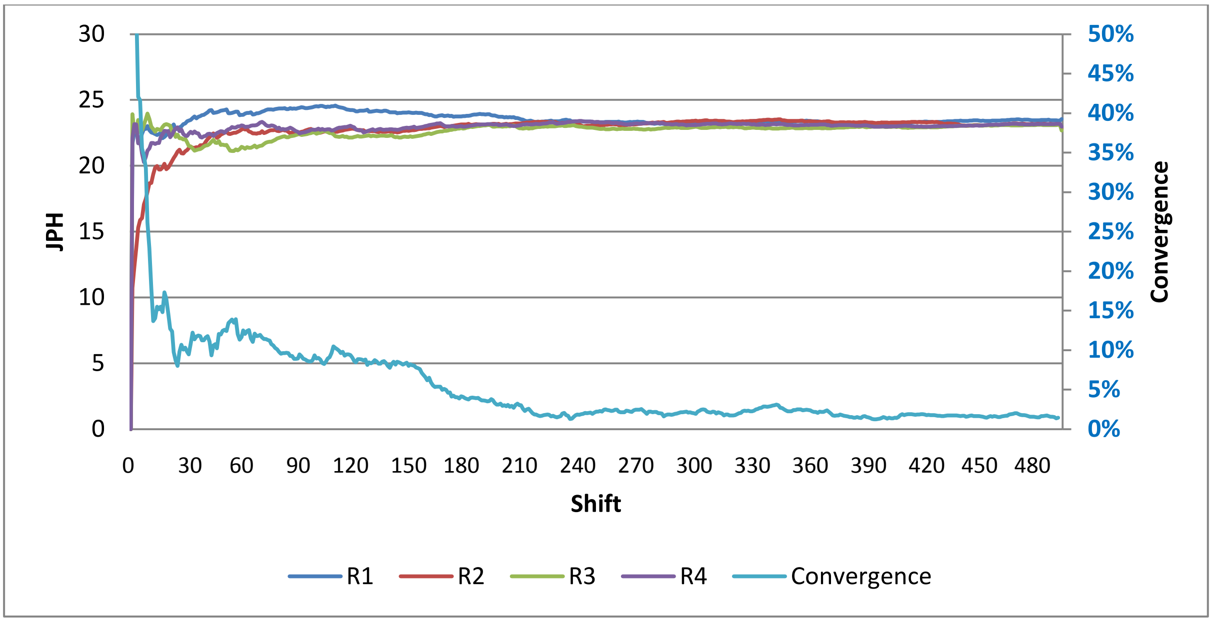


Figure 6.3: Cumulative mean time series and convergence plot

Inspection of the time series showed the graph to be initially very sporadic before settling to a level of convergence below 5% at around 210 shifts. A period of 300 shifts, or equivalently 100 days simulation time, was decided upon for further experimentation as the convergence settled below the five percent level and all four replications were similar.

6.5 Experimentation

Having built the verified and validated model and assessed the minimum warm-up and run length required, experimentation was undertaken in several key areas. A description of the experimentation process and results follow.

6.5.1 Identification of Bottleneck

With the aim of locating the bottleneck operation the interaction of elements on the line was examined through analysis of the statistics of all the operations on the line. This output data was then combined and a stacked column chart created with categories for the states related to the performance and statistics generated by Witness such as the utilisation (cycling), broken down, blocked and idle times (Leporis and Kralova, 2012). Figure 6.4, displays two extracts from the chart and a key to the relevant states. For the purposes of confidentiality the highlighted operations are labelled A and B.

Although there are numerous methodologies recognized in the literature for assessing bottleneck processes using simulation, the method used in the case study was the Utilisation Method. The method employed was to analyse the stacked chart to identify the operation with greatest active time (Roser et al, 2003), that is: “when an item is performing a function” (Faget et al, 2005). This is a highly respected method for identifying bottlenecks, as discussed in the above references, and allows quick and simple identification of the bottleneck operation. Operation B was analysed as the item which had the most active time, since it showed the tallest bar excluding the *inactive* states; blocked, purple, and starved, yellow. In particular Operation B showed over twelve percent of its total state recorded as broken down. By contrast, Operation A showed a very low utilisation on the station, or the least active state, with less than ten percent cycling state.

Figure 6.4: Operation State Data

Due to the method used within Witness, the broken down state captures not only machine failures but also operator breakdowns or overcycles. Thus, further analysis was required to understand the reason for the low availability of operation B, as reflected in the simulation output. Analysis was undertaken on the input data, and in particular the POSMon data used to generate the breakdown distribution sampled. This examination highlighted that there was a high frequency of operator overcycles on the station.

In order to understand if the data was erroneous, or indeed there was an issue with the availability of operators at the station, advice was sought from the Industrial Engineering (IE) team at BEP. The data was found to be correct and after brief investigation of Operation B it was established that there was a large complexity in the tasks undertaken at the station. This validated the model output and confirmed operation B as a bottleneck process.

6.5.2 Breaking the Bottleneck

Having identified the bottleneck operation within the model and validated its presence in reality, attention turned to solving the issues raised. Based upon the confidence gained in the model in correctly identifying the problems this process continued with the author working closely with the plant IE team. A system was implemented whereby the team proposed methods on how to best 'break the bottleneck' using MODAPTS analysis and the proposed changes were simulated. The identification of Operation A, through simulation, as being the least utilised operation was exploited when forming the proposals. The final solution involved introducing an additional station near Operation B which allowed the complex work undertaken at B to be balanced across several stations. A series of simulations were used to finely adjust the balance of work across the stations. The additional operator required for this station came from eliminating the work content at Operation A and transferring the operator across to the new team.

The result of breaking the bottleneck, initially present at Operation B, was a 2% to 4% increase in simulated EOL JPH. The range of values given is a result of the difficulty in predicting the effect on the resulting level of overcycles when dealing with untested work content at the new stations. The range covers the most conservative estimate, where by the frequent operator overcycles are left in place at station B, and the more realistic, using a similarly loaded operation with similar work content to model the new stations breakdowns.

6.5.3 Platen Experimentation

The modelling approach was validated for platen experimentation during the model building stage. Consideration of the number of platens removed with engines at spurs and delayed awaiting repair was already imbedded within the model. Platen experimentation was conducted for a range of values, 90 to 290, centred on the initial number of 190 platens in use. The model was run for 4 random numbers across the range of platen quantities in 5 platen intervals. Figure 6.5, depicts the analysed model output.

The analysis showed the AJ models' lack of sensitivity to the number of platens across a large range of values. Model validation was revisited for an extreme number of platens and animation investigation was repeated to confirm the results. Having confirmed the approach used was correct, the results were presented and it was recommended that 170 platens were used in total, although around 140 to 200 platens would not significantly impact the JPH achieved, as indicated with vertical green lines on Figure 6.5.

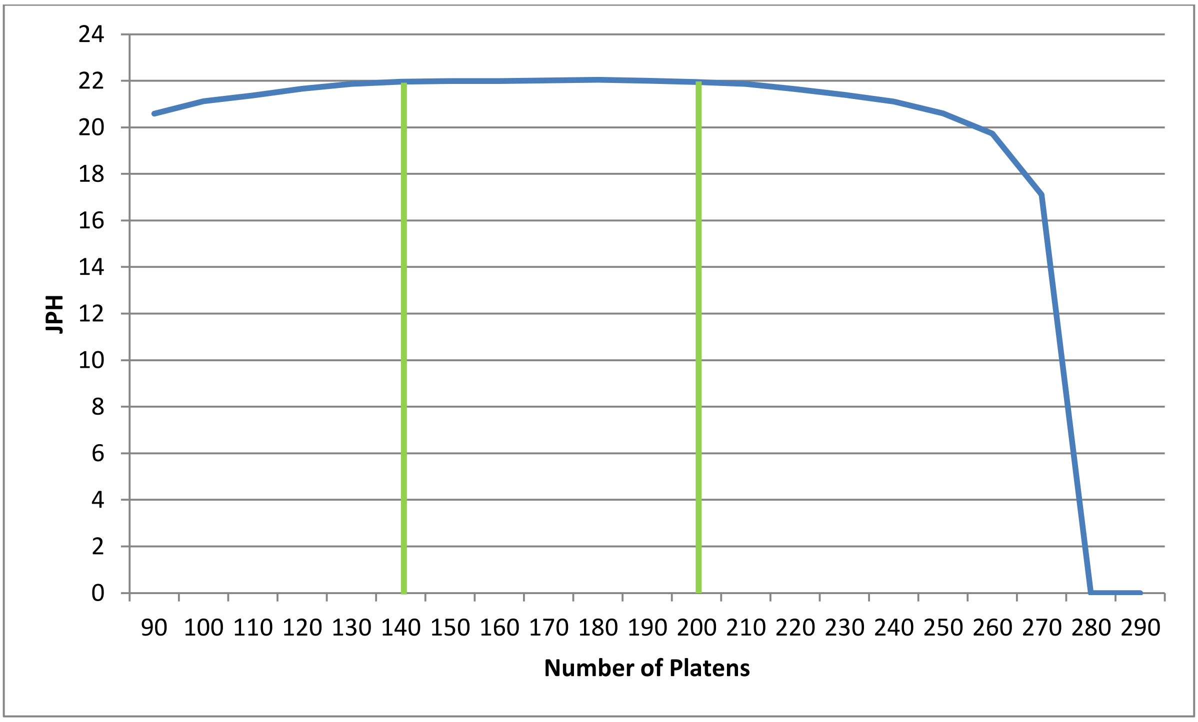


Figure 6.5: Platen Experimentation Results

As a further point to note, as the number of platens increases passed 210 the JPH achieved starts to decrease until finally reaching zero at approximately 280 platens. This is a result of the way the platens are fed into FAST and there being a finite number of pitches on the assembly line. Once the 280 mark is reached the assembly line is grid locked and engines cannot flow around the line to be assembled; hence the JPH reaches zero.

6.6 Summary

On proposing the changes to address the bottleneck to the AJ production team at BEP there was no real knowledge of the frequency of over cycle occurrences initially at Operation B. Where there was acknowledgement of the difficulty of the tasks undertaken, the effect on the overall system was unknown or hugely underestimated. DES quantified the effects of operator overcycles and their impact on an assembly line process. The simulation also provided a tool to ensure the solution proposed addressed the problem effectively. The identification of an underutilised operation allowed the implementation costs to be massively reduced as well as increase the average utilisation of the assembly line. The simulated changes were implemented to the line resulting in an increased efficiency for very low implementation costs.

The AJ assembly line production manager was surprised by the result gained from the platen experimentation and stated how an order had been placed for an additional number of platens. When prompted as to how the decision was made to purchase further platens it was explained that a comparison of the number of pitches to number of platens had been made and it was anticipated that the additional platens would increase the EOL JPH. The results included in Figure 6.5 were used to explain how the investment in additional platens was anticipated to show no benefit and, in fact, may actually hinder production. Further, it was recommended that where platens were broken it may be worth reconsidering immediate replacement. As a result, the order for additional platens was cancelled.

6.7 Overall Conclusions

The work outlined within this case study is a result of the knowledge gained through research and developed through previous chapters of this dissertation. The success of the application of DES provides further evidence of its usefulness as a tool for both finding bottlenecks and solving them in real life manufacturing systems. The assessment of required platen quantity identifies gaps in understanding that can be addressed through DES.

For DES projects to be successful it is important to link back from model to reality to gain confidence and credibility from a team of engineers. Further, as exemplified in this study, there is great importance of building a team of different skill sets. Being near the facility and its staff increases the effectiveness of this process as rapport is built and credibility can be gained in both the DES tool and modeller.

Chapter 7

Overall Impact of Research in Ford BEP

The work undertaken during the course of the research in Ford has had benefits beyond those discussed in the course of this thesis. Many of the detailed tasks undertaken and errors corrected within the software and processes have not been discussed within the body of the dissertation. The following section is included to highlight the importance of the work included in this dissertation and the impact created at Ford BEP in particular. The information included is applicable primarily to Ford although it is valuable for considering the usefulness of DES in this setting and in particular through placement of simulation personnel close to manufacturing systems.

7.1 Work Completed

In order to address these initial objectives the project was split into several smaller more manageable elements, some of which have not been included within this dissertation.

Project 1 – Manning Requirements for Module 3 Head Buy Off

Project 2 – Update and Verify AJ Assembly Model

Project 3 – Optimise AJ Assembly Model

Project 4 – Sigma Familiarisation and Data Collection Head Assembly

Project 5 – Model AJ Head Assembly Line

Project 6 – Model Sigma Assembly

Project 7 – Optimise Sigma Assembly

This is only a snapshot of the total work carried out since many of these projects were split in to sub projects in order to meet the objectives within the original plan. Work has also been conducted on machining lines and even abstract projects including a car park facility.

The main focus was developing an approach to apply discrete event simulation effectively within the BEP and in particular transfer the knowledge at the University into industry. As the project progressed it quickly became apparent that the problem with simulation was not limited to within BEP. An opportunity presented itself that this project could make changes not within only Ford BEP but could benefit the simulation in the whole of Ford Europe. Although the project was initially designed to utilise the Witness software as the sole DES, it was discussed and the project broadened to include use of the Ford Simulation Package, FAST.

7.2 Benefits to Ford

In order to discuss the benefits of the work at BEP this section is split into three categories.

7.2.1 Application of DES and FAST

Through line side placement a better understanding of the use of Witness within Ford was gained and developed. This knowledge was combined into a training manual as previously discussed providing long term ability to improve the application of DES. The work in chapter 4 also improved the process of verification and validation through use of the models created. Through this better understanding the following discrepancies between simulation and reality were discovered and corrected:

- First Time Through
  The modelling of FTT has been improved by considering the reality and in particular the group leaders involvement. The group leaders function involving FTT is now correctly modelled.
- Operator Over-Cycle's
  Through consideration and study of the role of operators off standard work such as stock handling the modelling approach has been adjusted. Again the group leaders involvement is now correctly modelled.
- MTTR and MTBF
  The Ford simulation POSMon data analysis tool is extremely powerful and analyses a lot of data greatly reducing the time to produce empirical distributions for breakdowns. Through verification of the tool an error was highlighted which meant incorrect records were being read. The correction of this provided more confidence in the resulting system analysis.
- Cold Test and Hot Test Modules
  Through verification an error in the FAST interface was highlighted and corrected such that cold and hot engine retest behaviour has been adjusted to work as required

- Buffer Modelling

  The Ford FAST system requires a buffer to be used as default even if there is not a buffer in reality. Through models created directly in Witness the error was demonstrated and the correct logic provided to fix the error.

These enhancements were communicated to the Ford Simulation team at Dunton for wider distribution within Ford Europe.

7.2.2 Understanding of Simulation

Utilising the capabilities of Witness and FAST a general improvement in the understanding of simulation has been transferred to many of the stakeholders who are regularly involved with simulation projects. Using the Witness interface directly in face to face sessions means that many of the operators on the assembly line have been shown the simulation tool and its power for predicting changes. Through this platform an understanding of their personal impact on the manufacturing line has been conveyed.

DES has been used and has become an integral part of the constraint meetings held on Sigma Assembly Line. The modelling of subtle changes has been completed prior to the raising of funding such that the best decisions can be made or the impact of changes assessed. Through these meetings both FAST and Witness has been used as a result some of the charts used in simulation presentations have been clarified.

Witness has been used ‘live’ on occasions to showcase the various impacts of changing parameters within DES. The use of the visual display has highlighted the impact of such changes highlighting to plant staff the need for accurate data as well as the true capability of DES as a tool.

7.2.3 Efficiency and Decision making

As discussed in Chapter 6, there has been an increase in production efficiency made through the use of DES in Ford BEP. This represents an increase in JPH of around 3%, which equates to an increase in turnover of approximately $15m per annum, for only very minor implementation cost. This change has only been made possible through the knowledge gained through the study of all the elements in this dissertation.

Other efficiencies have been made through the use of DES including work on manning strategies for crank and head machining lines, as well as an increased awareness of the workings of complex systems including leak test machinery and counterbalanced elevators.

7.3 Contributions to knowledge at Ford

The following contributions to knowledge at Ford have been made:

- Provision of a documented account of the issues surrounding verification and validation through research in the form of formal presentations. Being plant side and gaining knowledge highlighted the need for better communication which is not always easy due to the complexity of assembly lines and simulation.
- The review of literature identified many standard approaches for verification and validation. The use of sensitivity analysis led to discovery of major errors embedded in the logic behind the custom FAST interface. This highlights the risk of becoming too reliant on an interface such as FAST as validation was relaxed within Ford.
- Through building numerous models directly into Witness and indirectly through FAST a comparison of the advantages has been compiled. Something that has not been possible to analyse previously due to lack of access to the Ford software.
- All of the knowledge gained and research undertaken have formed a training manual in Witness that will be used for Ford (helping those using FAST gain a better understanding of the details of Witness DES) and students or staff at academic institutes.

Work has been rolled out to Dagenham engine plant and throughout Ford Europe in the methods of improving the accuracy of simulation.

7.4 Summary of Chapter

There has been an emphasis on increasing awareness of simulation within the host company; as a result, simulation is no longer a tool that is so misunderstood. The work undertaken at Ford BEP has provided both short and long term benefits. Increased efficiency on the AJ assembly line has resulted in short term annual savings for the company. The models created as part of this project have been embedded within Ford for future use representing a midterm benefit. Perhaps most importantly, the long term benefit is the knowledge embedded within Ford and methods implemented providing a framework for effective modelling of all future assembly lines. Through increased scope of the project these benefits are not limited to BEP but any Ford plant that chooses to utilise the FAST toolset.

Chapter 8

Conclusions

The aim of the research undertaken was to use the DES tool to provide better understanding of assembly line simulation and in turn of manufacturing processes and data themselves in order to advance the application of Discrete Event Simulation within the automotive industry. The dissertation is structured to develop this understanding of assembly lines, and the significance on DES, by building on knowledge through a chapter by chapter approach. Chapter 2 discussed the current application of DES within Ford and, through research, informed the reader of the requirement for structure when modelling complex systems; as exemplified by the FAST interface within Ford. Chapter 3 improved the understanding of automotive assembly lines and applied this knowledge to improve the concepts used in DES modelling as exemplified through the creation of several complex assembly line models. The pitfalls within current simulation were examined, within Chapter 4, and contributing factors to a lack of confidence portrayed by plant side engineers were addressed. The validation and verification methods have been researched through literature review and applied within Ford Europe. The resulting impact was the identification of major errors within the current DES system and steps were taken to fix these as well as developing a method for better future verification. In order to address some of the issues with believability and credibility of assembly line modelling, an in depth study and investigation in three key areas was undertaken within Chapter 5. Finally, Chapter 6 combines all of the knowledge gained and, through application, increased production efficiency by 3% which was made possible only through this body of research.

The key findings through each of the stages within this dissertation can be identified and combined to highlight the wider benefits to the work undertaken. The section is split to discuss the findings related to the DES and, later, the reasons for the scepticism towards simulation.

8.1 DES as a tool

The dissertation has focussed around the use of two methods for building models, the use of DES directly and the use of an interface, FAST, to manipulate the DES without user intervention. Thus, findings are made relative to the individual tool where necessary.

The results of this study highlight the importance of matching the simulation tool to the requirements of the project being undertaken. Both Witness and the FAST interface have proven to be useful in different circumstances and therefore a manufacturer should not limit the choice to one particular method or software. The use of an interface, such as FAST, and the availability to make use of predefined jigsaw pieces allows quick model building which is important when modelling systems that continually change. Assembly lines fit in to this category and represent good reason to make use of such an interface. However, utilising the DES directly is better where detail is required to be modelled outside of the framework offered by a restricted interface. Naturally, use of the DES directly requires greater skill and knowledge of simulation as a whole in order to achieve accurate simulation. However, the benefits include: greater visual output and flexibility to increase the detail modelled.

The building of lines at distance has been shown to be a valid approach when using a structured interface as there is a reduction in knowledge of the assembly lines required for accuracy. However, the results of this investigation have shown the requirement for accurate data inputs that cannot be validated when modelling away from the manufacturing process. For this reason the interface, FAST, has been to shown to act as a data driven system that allows little flexibility or creativity.

There was a need highlighted for better consideration to the Validation and Verification (V + V) of simulation models. The finding of two major (as in Appendix 3 and 4), and several less significant, errors within a short placement highlights a lack of attention placed on V+V. It has been demonstrated that the use of an interface has implications on V+V through the over reliance on a familiar system and the presence of hidden workings of the system, which can also cause difficulty. The credibility of models is also to be considered when implementing an interface for repeated simulation; do it right and you build confidence in one model, however, do it wrong and you effect the results of current, future and past models and therefore the believability of simulation as a whole.

Overall, the work within this thesis highlights that both tools are capable of simulating assembly lines but the choice of which one to use should not be routine. The greater understanding and flexibility offered within Witness should be taken advantage of, but increased caution is required with V+V. The use of an interface reduces the time to build lines but is so data driven that it requires almost factual information to get right. The constraint within this investigation was often not the DES tool chosen but rather the data available and the flexibility to adjust approaches to suit.

8.2 Scepticism of Simulation

There is no doubt that simulation, and in particular DES, is a powerful tool that when utilised correctly can provide solutions to complex problems. However, when faced with the decision to use the tool within industry there remains a level of scepticism. Thus, when given the opportunity to undertake an appropriate project the use of DES is either rejected or full buy in is not provided. The reasons believed to contribute to this scepticism are discussed in this section.

There is some overlap between the knowledge of simulation personnel and those involved with manufacturing engineering. Engineers are generally trained in many of the areas, such as mathematics and the associated statistics, that involvement and understanding of a simulation project requires. Of course, there are gaps between the two professions such that the engineer could never fully understand all aspects of simulation without great time investment and effort. Likewise the modeller will not have the same level of knowledge of production systems as manufacturing engineers. It is proposed that this is not necessarily a problem.

Engineers do not fully understand many aspects of projects they can be involved in, yet this gap in knowledge does not seem to affect their involvement or motivation. For example, consider a machine which requires reprogramming to address an issue. A manufacturing engineer may not understand how to conduct this task themselves yet are happy to appoint a professional to undertake the work, whilst providing a description of the task and managing the project through good communication. This example draws many similarities to within a simulation project.

It is proposed that there is no need for the manufacturing engineer to fully understand the task that is being completed by the simulation engineer. Instead through communication of key information the manufacturing engineer can be involved with and manage the project remotely without understanding the exact detailed approach used by the modeller. Providing the resulting simulation model is correct and validated for its purpose there should be no concern regarding the use of the model. However, there are also some key differences to the example addressed. Firstly, the result of successfully reprogramming a physical machine is obvious and expected, and, secondly, the machine can be tested by the manufacturing engineer themselves. Finally, when fixing a machine there is often a framework provided to undertake the repair or there is only one route suitable to fix the machine.

When considering a simulation, the output is clearly not as obvious as a machine correctly processing parts. Thus, communication is required in a format that is obvious and applicable and most importantly understandable to the manufacturing engineer - as discussed in Chapter 4. The need for good communication here is emphasised as the manufacturing engineer does not have the skill set to actually assess the simulation created themself. Unlike repairing the machine, the creation of a simulation project is not always one dimensional and the modeller has freedom to proceed how they see fit. Due to the limitations, albeit reduced through advances in DES over recent years, there are assumptions required which need to be carefully considered. As a result there is requirement for a competent simulation expert to be appointed and responsibility lies with this person to convey important details to the manufacturing engineers for validation. Where there is no choice in who undertakes the project confidence may be impacted. Perhaps the most difficult aspect when considering the results of a simulation project is the outcomes of the simulation project do not always match what is expected by the manufacturing engineer. As the tool is still relatively new it is apparent that engineers do not place complete trust in the work that goes against 'gut feeling'. It is believed by the author that as the number of successful simulation projects increase and changes implemented resulting from these the confidence in DES tool will increase.

Simulation is currently in a position where it is ready to be used but until success has been demonstrated to all stakeholders the scepticism will continue to hinder the effectiveness of simulation in manufacturing changes. Overall the simulation modeller needs to understand the entities within the simulation package and have awareness of their application to real manufacturing systems. The manufacturing engineers need to convey their knowledge of the real systems to the modeller, with sufficient detail, understanding their applicability within DES. If this is achieved, through strong communication, then this study has shown that DES is a capable tool when used to increase assembly line productivity.

8.3 Contributions to Knowledge

The research and developments made possible through this work has made the following contributions to knowledge:

- A better understanding of the applicability of the various DES elements in a manufacturing specific context in relation to real life manufacturing systems
- Valuable information has been disseminated within Cardiff University for learning DES and supporting a better understanding of manufacturing systems in both teaching and research.
- An added capability to the use of Witness through novel approaches used within the manufacturing environment.
- Consideration of the way in which collaboration with plant engineers at the model building stage can overcome the pessimism expressed towards simulation and in particular DES
- An enhanced framework for verification and validation when working with DES and complex systems

Although the current study is based on a small number of assembly lines and limited within one particular automotive plant these lines are complicated and do contain examples of many different elements that one would expect to find in many other modern manufacturing facilities. This means that the findings are widely applicable and the information provided can make a useful contribution to the understanding of systems across many manufacturing applications. Despite its focus and bias towards the FAST system within the host company the issues surrounding the use of a database model building interface are common to other similar systems. The developments made possible through research in this dissertation will serve as a base for further work in the area of manufacturing simulation and the areas associated with use of DES.

8.4 Future work

This research has identified many questions in need of further investigation.

Firstly, more work is required internally within Ford to broach the lack of confidence in the data sources used within simulation. It is believed that this will directly motivate future work to understand the changes when belief is gained in the simulation toolset.

Considerably more work will need to be undertaken to confirm the applicability of the work contained within this dissertation to general manufacturing outside of engine assembly lines. Achieving this goal requires access to different manufacturing systems and real world data.

Further research might explore the psychology of the engineers towards the perceptions of simulation as a tool. Although this is outside of the scope of discussion, a change of mentality is required to motivate staff towards the use of simulation as a tool. It is planned by the author to continue work in this area and understand the change to perception of simulation through the undertaking of this placement and creation of this dissertation.

Further investigation and experimentation into the communication between modeller and engineer is strongly recommended. Creation of a standard document to record data included and assumptions made may ease this communication such that engineers understand, and can therefore approve, what the model is based on. This may be especially relevant when the data input, and assumptions made, differ from what is considered normal.

Overall, steps are required to continue the learning and understanding of DES within a manufacturing context such that the tool can continue to move towards emulation. The following topics have been considered by the author for future work:

- Assessment of the minimum quality of data required for accurate simulation by considering the sensitivity to the various inputs within an assembly model
- Examination of different cycle time and buffer distributions across manufacturing lines to minimise effects of operator variability and unscheduled machine downtime during the line design phase

- Better understanding and modelling of repairs and rework on assembly lines to utilise DES as a tool to understand the best approach for real assembly line systems

References

B. W. Hollocks. 2006. Forty Years of Discrete-Event Simulation: A Personal Reflection. *The Journal of the Operational Research Society*, Vol. 57, No. 12. Dec, 2006, pp. 1383-1399

Baines, T. Mason, S. Siebers, P. Ladbrook, J. 2004. Humans: the missing link in manufacturing simulation? *Journal of Simulation Modelling Practice and Theory 12* pp. 515–526

Bangsow, S. 2012. *Use Cases of Discrete Event Simulation, appliance and Research* (Ed.) 2012 Springer Books.

Banks, J. 2007. *Handbook of Simulation: Principles, Methodology, Advances, Applications and Practice*. Published online Dec 2007. John Wiley & Sons, Inc.

Banks, J. Carson, J. Nelson, B. Nicol, D. 2001. *Discrete-Event System Simulation.* (3rd Edition) Prentice Hall. 2001.

BBC, 2013. *Ford Bridgend's £24m new engine production secures jobs*
Available at: http://www.bbc.co.uk/news/uk-wales-south-east-wales-22244777
[Accessed: 3 November 2013]

WalesOnline, 2014. *Car giant Ford appoint a new plant manager at Bridgend*
Available at: http://www.walesonline.co.uk/business/business-news/car-giant-ford-appoint-new-7072972
[Accessed: 20 June 2014]

Box, G Lunceno, A. Quinones, M. 2009 as cited in Leal, F 2011. *Statistical control by monitoring and adjustment*. 2 ed. New Jersey: John Wiley & Sons.

Dewson, 2006. *Simulation Philosophies within Ford Motor Company*. MRes Dissertation, Cranfield University.

Diamond (2010) as cited in "*The Application of Discrete-Event Simulation in Production: A case study in Volvo CE.*" 2012. Norouzilame, F (Mälardalen University, School of Innovation, Design and Engineering)

Faget, P. Eriksson, U, Herrman, F. 2005. Applying Discrete Event Simulation and an Automated Bottleneck Analysis as An Aid To Detect Running Production Constraints. *Proceedings of the 2005 Winter Simulation Conference.* Dec 2005. Orlando, USA.

Ford UK, 2010. *News from Ford*
Available at: http://www.ford.co.uk/experience-ford/AboutFord/ News/CompanyNews/2010/ PrinceofWales
[Accessed: 20 August 2013]

Ford UK, 2013. *Ford Bridgend earns multi-million pound investment to produce new 1.5-litre EcoBoost Engin.*
Available at: http://www.ford.co.uk/FordFleet/NewsAndReviews/FordForBusiness/2013/May/Multi-Million-Pound-Investment
[Accessed: 3 November 2013]

Ford, H. Crowther, S. 2005. *My Life and Work*. [EBook #7213] [First posted on March 27, 2003] Edition: 10

Gass, S. I. 1984. Documenting a computer-based model. *Interfaces*, 1984, Vol.14, pp.84-93

Greasley, A. 2004. *Simulation Modelling for Business*. 2nd Edition. Ashgate Publishing, Ltd

Johansson, B. Jain, S. Montoya-Torres, J . Hugan, J. Yücesan, E., eds. Verification and Validation of Simulation Models. *Proceedings of the 2010 Winter Simulation Conference.* December 2010. Baltimore, USA.

Ladbrook, J. and Januszczak, A. 2001. Fords PowerTrain Operations – Changing the Simulation Environment. *UKSIM 2001, Conference of the United Kingdom Simulation Society.* Cambridge UK, March 2001. pp.81-87

Lanner Forms Partnership, 2013. Design Products & Applications.
Available at: http://www.dpaonthenet.net/article/56408/Lanner-forms-partnership-with-Virtalis.aspx
[Accessed November 2013]

Lanner, *Ford Engine Assembly Line Simulation Tool Case Study*.
Available at: http://www.lanner.com
[Accessed: June 2013]

IT Director, *Lanner Group partners with Frontline to provide simulation modelling solutions to UK health sector*
Available at: http://www.it-director.com/enterprise/public_sector/news_release.php?rel=19696
[Accessed: June 2014]

Lanner, About *Lanner*
Available at: http://www.lanner.com/en/about-us.cfm
[Accessed: October 2013]

Law, A. 2010. Statistical Analysis Of Simulation Output Data: The Practical State Of The Art. *Proceedings of the 2010 Winter Simulation Conference.* December 2010. Baltimore, USA.

Law, A.M. 2007. *Simulation Modelling and Analysis*. 4rd Edition. McGraw-Hill, New York.

Leal, F. Costa, R. Montevechi, J. Almeida, D. Pequi, F. 2011. A Practical Guide For Operational Validation Of Discrete Simulation Models. *Journal of Pesquisa Operacional* .Vol.31 no.1 Rio de Janeiro Jan./Apr. 2011.

Lu, R.F., Qiao, G. and McLean, C. 2003. NIST XML Simulation Interface Specification at Boeing: A Case Study. *Proceedings of the 2003 Winter Simulation Conference,* Vol. 2, pp. 1230 – 1237. December 2003. New Orleans, USA.

Mahajan, P. Ingalls, R. 2004. Evaluation Of Methods Used To Detect Warm-Up Period In Steady State Simulation. *Proceedings of the 2004 Winter Simulation Conference.* December 2004. Washington, USA.

Mason, S. Baines, T. Kay, J. Ladbrook, J. 2005. Improving the design process for factories: Modeling Human performance variation. *Journal of Manufacturing Systems.* Volume 24, Issue 1, 2005, Pages 47–54.

Mason, S. Baines, T. Kay, J. Ladbrook, J. 2005. Improving the Design Process for Factories: Modelling Human Performance Variation. *Journal of Manufacturing Systems* Vol. 25/ No.1 2005.

Mohsen, J. Eldabib, T. Naseera, N. Lampros, K. Stergioulasa. 2010. Simulation in manufacturing and business: A review. *European journal of operational research*, Volume 203, Issue 1, 16 May 2010, Pages 1–13.

Page, E. & Opper, J. 1999 Observations on the Complexity of Composable Simulation. *Procedings of the 1999 Winter Simulation Conference*. Phoenix, AZ. pp. 553-560.

Pavol Semanco, David Marton. 2013. Simulation Tools Evaluation using Theoretical Manufacturing Model. *Acta Polytechnica Hungarica* Vol. 10, No. 2, 2013.

Robinson, S. 1994. *Successful Simulation: A Practical Approach to Simulation Projects*. UK: McGraw-Hill Book Company Europe.

Robinson, S. 2007. *Simulation the Practice of Model Development and Use*. UK: John Wiley and Sons.

Robinson, S. A. 2002. Statistical Process Control Approach for Estimating the Warm-Up Period. *Proceedings of the 2002 Winter Simulation Conference.* December 2002. San Diego, USA.

Roser, C. Nakano, M. Tanaka M. 2001. A Practical Bottleneck Detection Method. *Proceedings of the 2001 Winter Simulation Conference.* December 2001. Arlington, USA.

Roser, C. Nakano, M. Tanaka, M. 2003. A Practical Bottleneck Detection Method. *Proceedings of the 2003 Winter Simulation Conference.* December 2003. New Orleans, USA.

Sargent, R. 2010. Verification and Validation of Simulation Models. *Proceedings of the 2010 Winter Simulation Conference.* December 2010. Baltimore, USA.

Sargent, R. G. Charnes, M. Morricem D. Brunner, T. Swain, J. 1996. Verifying and validating simulation models. *Proceedings of the 1996 Winter Simulation Conference*. December 1996. Coronado, USA.

Sargent, R. G. ed Tew, J. Manivannan, D. Sadowski, A. Seila, A. 1994. Verification and validation of simulation models. *Proceedings of 1994 Winter Simulation.* December 1994. Orlando, USA.

Soroka, A. Bennett, A. Prickett, P. Higgins, M. 2013. Industrial Perceptions of Manufacturing Operations Simulation. *Proceedings of the 2014 International Conference on Sustainable Design and Manufacturing.* April 2014. Cardiff, UK.

Swabey, P. 2013. *Simulated manufacturing at Ford Motor Company*.
Available at: http://www.information-age.com/technology/information- management/ 123457369/ simulated-manufacturing-at-ford-motor-company
[Accessed: November 2013]

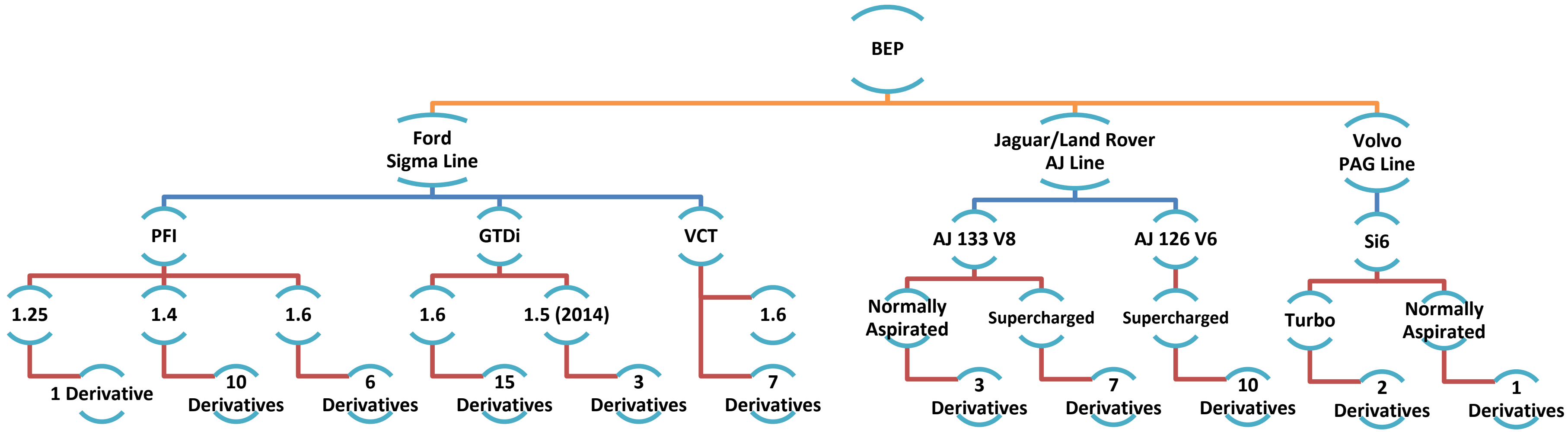
Timmiss, A. *Using Varying Levels of Simulation Detail to Model One System. Nissan Motor Manufacturing LTD*.
Available at: anthony.timmiss@nissan-nmuk.com

Ulgen, O. and Gunal, A. (2007) *Simulation in the Automobile Industry, in Handbook of Simulation: Principles, Methodology, Advances, Applications, and Practice* (ed J. Banks). Hoboken, NJ, USA John Wiley & Sons, Inc.

Winnell, A. and Ladbrook, J. 2003. Towards Composable Simulation: Supporting the Design Of Engine Assembly Lines. *Proceedings 17th European Simulation Multiconference*, June 2003. Delft, Netherlands: SCS-European Publishing House.

Appendices

Appendix 1 - Engines Created at Ford BEP (2013/2014)



Appendix 2 – Witness Instruction Manual for Manufacturing

User Guide for Manufacturing

Michael Higgins

December 2013

Version 1.3

Introduction

Please note, the contents page has been removed from this version but is present on the CD attached.

The aim of this user guide is to provide an insight, through worked examples, into the functionality of the WITNESS 12 simulation software from the Lanner group. By no means is this intended to replace a full user guide with appropriate training, instead the methods developed and used daily in developing simulation are explained. Emphasis is placed on the use in a manufacturing context; however the material presented should allow the key skills to be transferred to many other areas.

The guide will begin with an outline of using the WITNESS 12 interface and an overview of the basic tools available to construct a model. The more advanced tools are then discussed including functions and attributes as well as routing parts through the model using tracks and vehicles.

For the avoidance of confusion I will refer to labour in the way approached throughout Witness – that is with the American spelling labor.

1. Interface

This guide is based on, and includes screenshots from, the WITNESS 12 Manufacturing Performance Edition version of the software. Following the continued success of the Witness package, Lanner group redesigned the interface releasing Witness 12 in 2012 . Major changes include the ability to float windows outside of the main screen as standard in Windows 7, the general quality of icons and toolbars have also been improved. Figure 3 and Figure 4 show the two most recent Witness versions; PWE 3 and Witness 12. Overall subtle refinements have been made that will not cause confusion to users who are familiar with previous versions of the software. Thus, it is hoped that this guide will remain semi-applicable to users of previous versions of Witness 12.

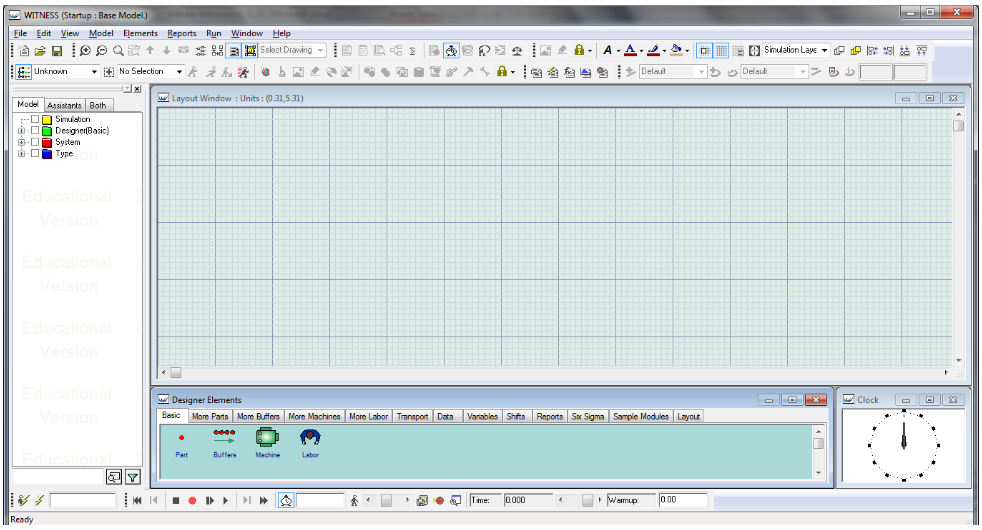


Figure 3 – Witness PWE 3

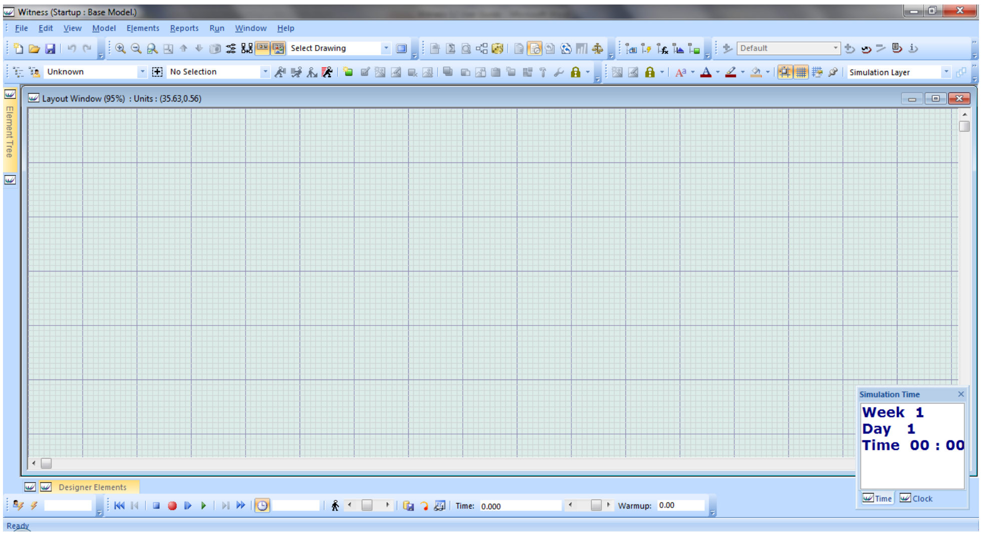


Figure 4 – Witness 12.
Notice how the interface is now cleaner allowing a large view of the main 'Layout Window'

There are several parts to the main window allowing the user to access a variety of options without the need to frequently access drop down menus from the toolbar. Figure 5 shows the Witness 12 user interface fully expanded to include the hidden windows.

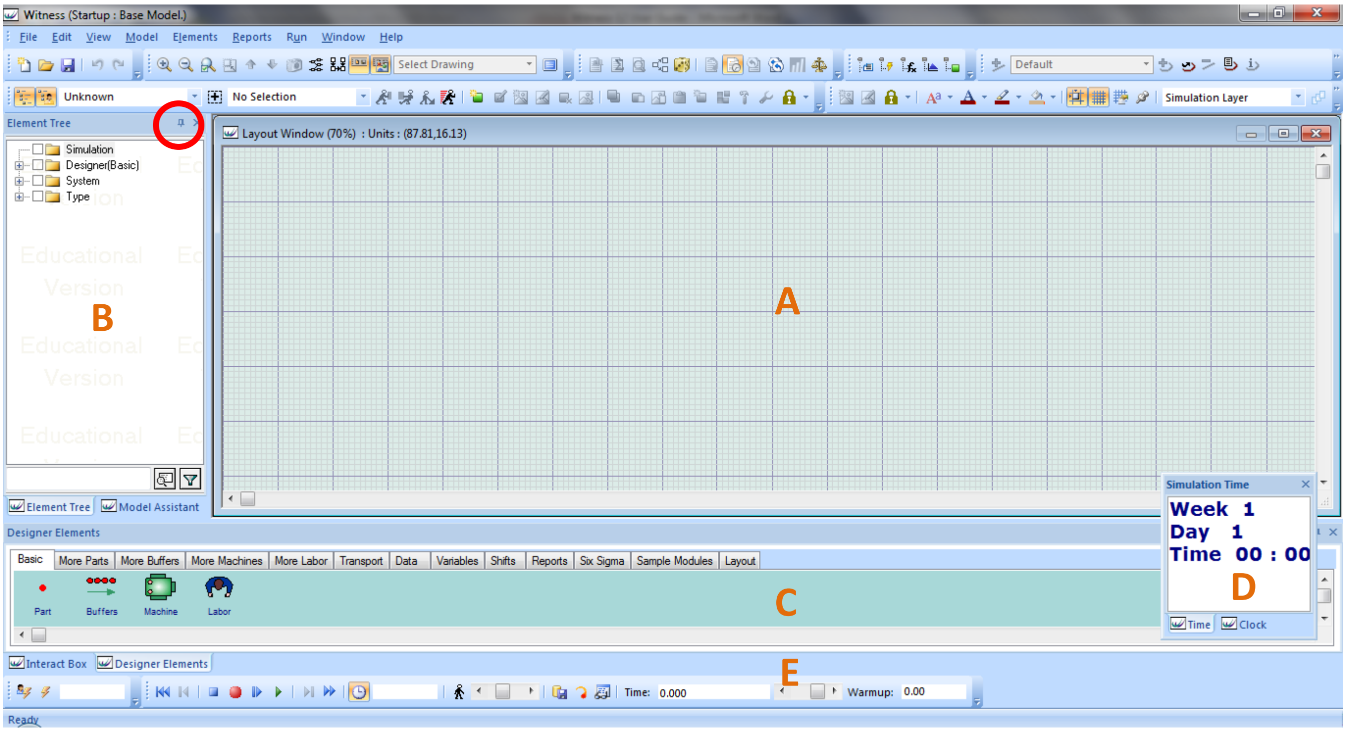


Figure 5 – Image outlining the different windows expanded
Layout window (A), Element Tree (B), Designer Elements (C), Simulation Time (D) and Simulation Toolbar (E)

Note, your screen may not look like this upon opening the Witness 12 software; this is due to the windows B and C being automatically hidden. In order to open these simply highlight the window such that it slides open and then press the pin (as circled in red above) to lock in position.

2.1 The Layout Window (A)

This is the area where the simulation is created and the visual representation laid out. It can be used to carry out several important tasks including defining elements, connecting elements and defining part flow and altering elements visual representation. When a simulation is running, the Layout Window provides a 2D visual representation of parts flowing through the various elements.

It is often useful to have more than one layout window open. This allows the user to have different parts of a simulation magnified in different layout windows. The ability to zoom in to key areas of the simulation further improves the visual representation of a model – this is especially useful on large models.

2.2 The Element Tree (B)

The element tree found on the left side of the window provides two main uses. Firstly, under the 'model' tab, found at the bottom of the element pane, there is an overall summary of all the elements that are part of the model. This provides the user a list of all elements included in the simulation in alphabetical order, there is also a search bar at the bottom of the pane which allows filtering by name and/or type of element. This is especially useful for locating and editing elements that do not include a visual representation in the layout window (A). Secondly, under the 'Model assistants' tab there is a list of commonly used functions within simulations. This includes formulae and distributions that can be inserted at various points within a simulation. Right clicking on a formula or distribution and selecting 'insert with prompt' or 'insert with wizard' provides the user with a step by step walkthrough in providing the necessary information for that item to be inserted.

2.3 Designer Elements (C)

The designer element window spans across the lower part of the screen, again if this is not visible it might be hidden using the auto hide pin. On opening a new witness window the designer element window may be empty. In order to fully initialise the Witness designer elements for manufacturing the startup.mod file can be used. This will load a whole range of pre-defined designer elements including labor, buffers, transport, variables and much more.

It is possible to add new elements to any witness file designer elements pane. This is useful as new/customised elements can then simply be found in the designer elements pane and easily inserted multiple times. In order to do this select the element which you would like to add and just click the 'Create Designer Element' found in the top tool bar.

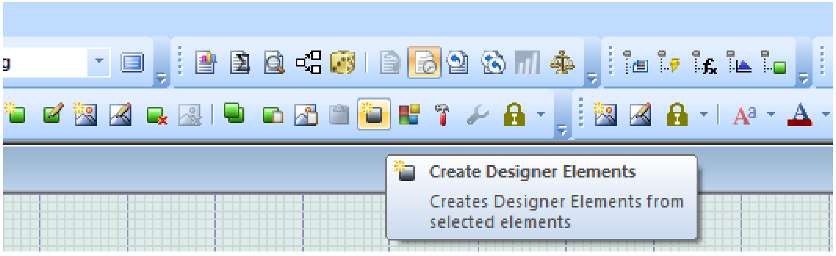


Figure 6 – Create Designer Element Icon

2.4 Simulation Time (D)

The simulation clock can be displayed in either analogue or digital format. The purpose of the clock is to give a visual representation of the time passing. It should be noted that the time units used could be seconds, minutes or indeed a variety of other units. This is defined by the user in the units used to define cycle times on machines etc.

Note: The clock may appear to slow down and speed up during the course of a simulation. This occurs due to the simulation speed varying depending on what calculations the computer is processing at that time.

2.5 Simulation Tool Bar (E)

The controls for running the simulation are located along the bottom of the main witness window. A set of intuitive icons are used, similar to that found on a DVD player remote control, that allow you to start, speed up, slow down, pause and stop a simulation. This is shown below as Figure 7.

Figure 7 – Simulation tool bar controls.
From left to right; rewind model to start, rewind, stop simulation, autosave, step, play, fast forward and batch.

The simulation run time is also set at the simulation toolbar (A), see Figure 8. This is the time that the model will stop at if you press play and leave the model. How long the simulation is run for is dependent on the size of the model and the amount of variability within a model. It is important to ensure the clock is highlighted to the left of the simulation time box, otherwise the simulation will run indefinitely ignoring the simulation time input. There is also an opportunity to input a warm up time (D). With a warm up time entered the model will run normally with the exception that all statistics and records will be reset to zero once the warm up time is reached. It is essential to warm up a large model for sufficient time such that buffers are filled and the line is in a natural running state. Note, if you wish to run a model for 60 minutes with a 10 minute warm up time it should be noted that the simulation statistics will be based on only 50 minutes simulation time as the 60 minutes simulation time includes the 10 minute warm-up and is not in addition.

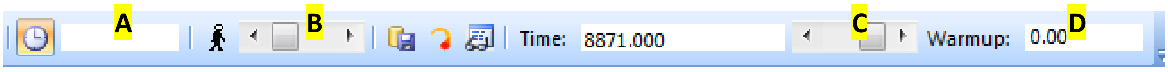


Figure 8 – Simulation tool bar
A – Simulation run time, B – Labor walk speed, C –Slow Motion Speed, D – Warm Up Time

Labor walk speed (B) and slow motion speed (C) control the speed the model runs at. It is useful to slow down the model using B to view labor walking from station to station and verify appropriately. Option C is normally left in its default position, full to the right, which means the model runs as fast as possible.

2.6 Other

The top of the main witness window houses the tool bar that provides a number of other functions. The most commonly used icons are those used to define the flow of parts through the model - that is the push and pull options for elements. There is also the ability to force breakdowns/repairs on machines and a whole host of other tools available to customise the appearance of the layout window.

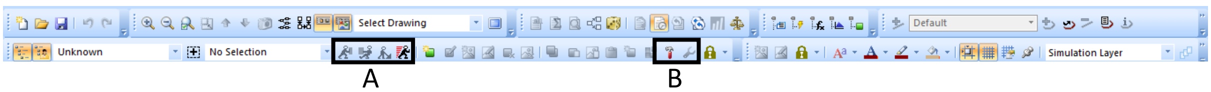


Figure 9 – Main toolbar found at the top of the screen. Highlighted are the push and pull tools (A) and manual triggering of faults and repairs (B).

As of Witness 12 release the ability to work with multiple monitors has been included. It is possible to pull the Element Tree (B), Designer Elements (C) and Simulation Time (D) outside of the main interface and onto a second monitor. This frees up the layout window to fill the first screen giving a better view of the elements. If a second screen is not available then it is possible to still maintain a clean view of the layout window by auto-hiding the windows B, C and D. To do this each of the windows can be 'auto-hidden' by clicking on the pin, shown in **Figure 10** below, on each respective window.

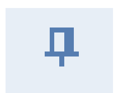


Figure 10 – Pin used to toggle auto hide and lock windows

2. Basic Elements

The word element has been used throughout the previous two chapters without explanation. In essence, element is a term used to describe any building block part of the simulation. It is useful to think of elements like pieces of Lego™ all of different colours and sizes but sitting on top of each other to assemble a larger item – the simulation.
Inserting a element within witness can be done in one of two ways, firstly a element can be selected from the designer element bar using a mouse left click and then left clicking again within the layout window. This method is most useful for inputting machines, parts, buffers and any other element with a visual icon. The second method is used for inputting elements such as attributes which do not have a visual icon. To insert such items, right click within the layout window and select define. A window will appear allowing selection of the element type using the dropdown box found at the bottom.
The witness software provides a huge amount of flexibility in the elements that can be used to construct simulations, however, in order to create a basic simulation there are essentially five different types of elements that can be used.

2.1 Parts

Parts are the most important part of a simulation. Without parts there is nothing to process through the simulation and there will nothing for witness to process. Parts represent the physical parts that move through the various operations, in a manufacturing capacity this could be a piece of rough casting that is input to the line, transported along tracks, into buffers, machined and then finally output as a finished product.

Parts can be modelled in two different ways, constrained with inter-arrival times (active profile) or with an infinite stock (passive profile). Modelling a part with a passive profile means parts can be continually pulled into the model without limit. In most cases it is unrealistic to model a part as being constantly available and thus an active profile is usually assigned to the part. This active profile outlines the time a part will first arrive in the model and then the time between arrivals; this can be represented with a distribution or a constant time difference. The maximum number of parts that can enter the model can also be set in an active profile.

Parts, like all elements, can be represented with a 2D or 3D graphic, this can be selected from one of the standard icons or a custom image can be included, and can contain either variable or fixed attributes that can be called in to the simulation.

2.2 Buffers

Witness allows a great deal of accuracy to be captured in the modelling of buffers. Buffer elements are used to store part elements throughout the simulation. In a manufacturing context this could be a pallet or perhaps a reserved, empty, pitch on a production line. There are two main types of buffers within witness; part buffers - where the buffers provide a graphics representation of each part within that buffer - and count buffers – where the buffer simply provides a count of the total number of parts within that buffer irrespective of the type of part.

There is no difference in the way the parts are stored within the buffer, however the display differs as per above. The part buffers are generally good for small models with different parts flowing through the model as it is possible to view the different part types within the buffer. For larger models the count buffer is more practical as the part buffer will display each part within the buffer taking up a huge amount of space in the layout window. Figure 11, below, displays the two different types of buffer.

It is often useful to include a delay on the buffer. This is the ability to assign a minimum, or maximum, amount of time a part has to remain in the buffer. It is also possible to model the buffer logic on a 'first in – last out' or 'first in – first out' method or pull particular parts from the buffer dependent on a parts attribute. These options are easily configured within the detail window of a buffer.

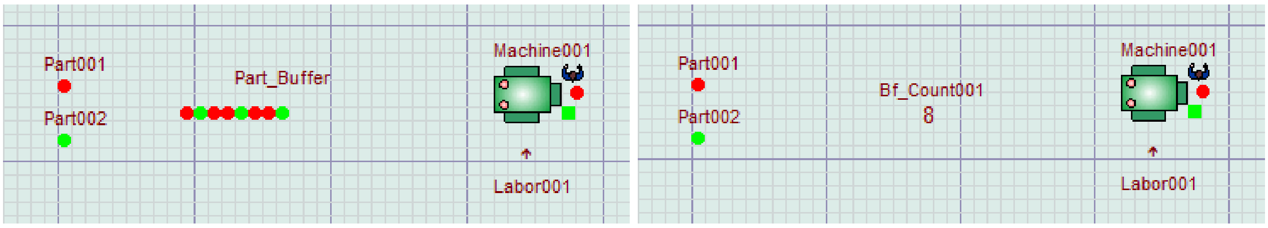


Figure 11 – Left; part buffer and right; count buffer

2.3 Machines

These are the elements that carry out all operations on parts within the simulation. Any time an operation is carried out on a part, whether it carried out automatically, semi-automatically or by a labor element it is necessary to represent the operation with a machine. The default icon for the machine element is a green machine perhaps giving the false impression that it is only possible to model automatic operations. However, within witness it is necessary to model manual operations using a machine element with labor assigned to it.

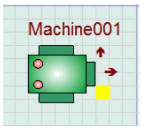


Figure 12 – Default machine icon. Not to be mistaken as purely automatic operations.

There are several 'type' of machines within the detail box for a machine, these allow easy customisation to represent a machine that is processing a single part or perhaps a batch of parts or perhaps carries out multiple cycles. Set up and break downs of machines can also be accurately modelled within this detail box,.

2.4 Labor

It is worth re-noting that the American spelling of labor is used within the witness package and for simplicity the American spelling is used throughout this guide.

Labor can be used for a variety of purposes within the simulations. Most frequently labor is assigned to machines to simulate manual operations. However, the labor element can be used to transfer parts along paths, set up machines and repair broken down elements. As in reality, once a labor element is assigned to a task it will become busy for that period and thus unable to take on other work. Care should therefore be taken when assigning labor to multiple tasks.

Labor elements can easily be assigned to work particular shift patterns which determine how long they work for and when/for how long their breaks are.

2.5 Transport

There are three main types of transport available to represent the flow of parts from element to element. These take the form of conveyors, tracks & vehicles, paths and overhead gantry networks.

2.5.1 Conveyors

Conveyor elements are used to transport parts on a continually moving track. We can further split conveyors into two types, continuous conveyors and standard conveyors. The major advantage of using continuous conveyors is seen when there are parts of various sizes to be transferred allowing parameters such as the size of the part and minimum spacing between parts to be modelled. Both the standard and continuous conveyors require input of the length, speed of travel and capacity of the conveyor.

2.5.2 Tracks and Vehicles

Tracks and vehicles are used together to represent platens carrying parts around a moving track. Parts can be loaded and unloaded to and from the track in to other elements such as machines or buffers, the method for doing this is examined in detail later in section 6.

Vehicles are what carry the parts around the tracks, unlike a conveyor a track is unable to carry a part unless there is a vehicle available and in the correct location. There are several parameters that can be input to the detail page of a vehicle, most of which are self explanatory. The most important settings are the maximum speed of the vehicle, the quantity of the vehicle available and the number of parts each vehicle can carry – it's capacity. It is worth noting that modelling a vehicle with a capacity greater than 1 adds extra complication when it comes to loading and unloading parts. Issues can arise with access to the required part and machines not being able to extract the desired part from the vehicle.

Tracks provide the means for the vehicles to travel around on and it is important detail is paid to inputting the right details to each track. The track limits the maximum speed of vehicles, for example if a vehicle has a maximum speed of 10 m min^{-1} but the track has a maximum speed of 5 m min^{-1} then the vehicle speed will be set to 5 m min^{-1}. It is also necessary to input the capacity, physical length, display length and where to load and unload parts. The physical length is the actual length of the track generally input as metres. The display length, however, is input to show the number of points on the track that Witness can display a vehicle – purely for display purposes. The capacity of a track cannot exceed the display length.

2.5.3 Paths

Paths are used to connect elements and can be used to ensure labor elements follow a particular route. For example, this could be a path from a labor elements rest position through to a machine it is operating on. Other examples include setting a path for parts to follow to a buffer, machine or many other elements. Within the path details box a time for the part to transverse a path can be entered, as well as it's displays length and most importantly, the start and end points for the path. It is not necessary to assign a path joining each part element to a machine, or similar, as by default Witness will automatically assign a pseudopath which is a direct path from A to B. If required, this option can be disabled within the model dropdown menu at the top tool bar by following **Model -> Options**. Once pseudopaths are disabled part elements can only move to its destination if there is a path assigned for them to follow.

It is worth noting that if labor is to follow the path then it must be added to the 'From' box in the path details box.

2.5.4 PF Transport (Overhead Gantries)

Witness also has standard elements used to represent overhead gantries named PF Transport.

Pf transport works in a very similar way to tracks and vehicles but work in set networks of sections that must be pre assigned before the carriers can move on them. A major difference between tracks and PF transport is that encountered when loading/unloading parts. Unlike tracks and vehicles parts can only be loaded to a PF network at a PF Station.

Note

It is possible to design the conveyors, tracks, paths and gantries to represent reality. It is relatively simple to do this in 2D and is a matter of simply inserting the elements in straight sections and then manipulating them in to shape. This done through selection of the track and using the CTRL key or SHIFT key depending if the line is to be shaped into an arc or ninety degree angle. This is how the logo on the front cover was created and it is a good exercise to become familiar with the design capabilities of Witness.

3. Detailing an Element

In order to give the accuracy within a simulation it is necessary to fill in particular details to the element, many of these are outlined in section 3 above, and include details such as the speed of a conveyor or the capacity of a buffer.

In order to edit the elements details we must first insert the element in to the model using one of the two methods outlined in section 3. Once part of the model one can simply double left click on an element icon within the layout window in order to open the elements detail window. If there is no icon, the element to be detailed must be located within the element tree, right clicked and the detail option selected from the dropdown menu.

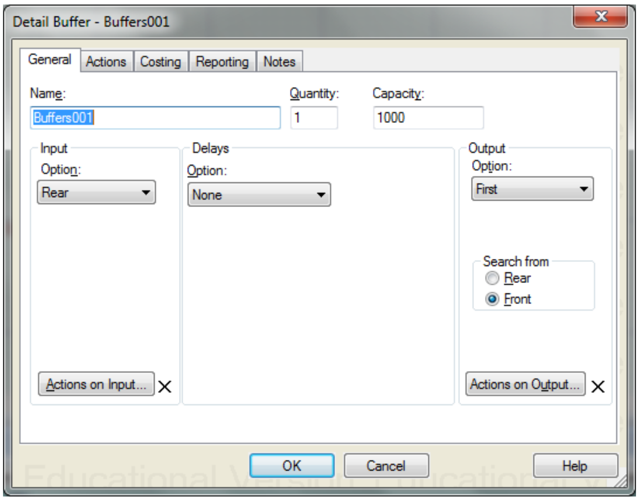


Figure 13 – Detail window for a buffer

Witness simplifies the process of inputting details in to elements as the detail window which will open contains only the relevant fields for that particular element. Further to this, selecting options from dropdown menus will either hide or reveal more options relevant to that particular element. For example figure 11 shows the standard buffer detail window once opened, once an option from the delay option dropdown box is selected more fields will be revealed as shown in figure 12.

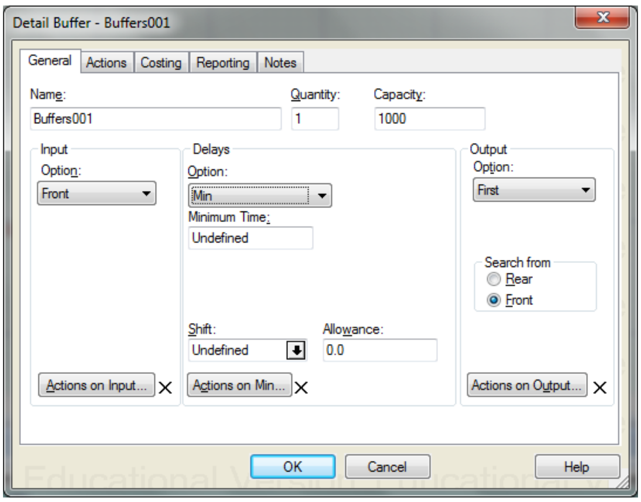


Figure 14 – Buffer detail window with delay option selected. Notice how more fields appear.

Within the detail window several options are available across several different tabs located at the top of the window for a buffer these include General, Actions, Costing, Reporting and Notes as seen in figures 11 and 12. These tabs will vary depending on the element however; the notes tab is found in all elements and allows the user to add notes about that element within a large text box. The reporting and costing tabs are also found in most, but not all, elements and allows the user access to advanced options on the way the element records within the model and to put a track on the resources an element is using – these options are not explored within this manual.

The main tab requiring attention is the 'general' tab which is where the main information about the element is entered. Tracks have two additional important tabs allowing the setting of loading and unloading paths. The populating of the details pages will be covered in more detail in later sections through the use of a case study.

Finally, within the details tab we can set 'actions' to occur under particular circumstances. These include carrying out a particular action when a part leaves or enters an element or when a machine starts cycling. In order to edit such actions we find a selection of buttons within the detail page, on clicking these 'action on ...' buttons a window titled 'Edit actions' will open with a space to enter text. This allows the entry of particular tasks or actions to carry out under that particular action on circumstance.

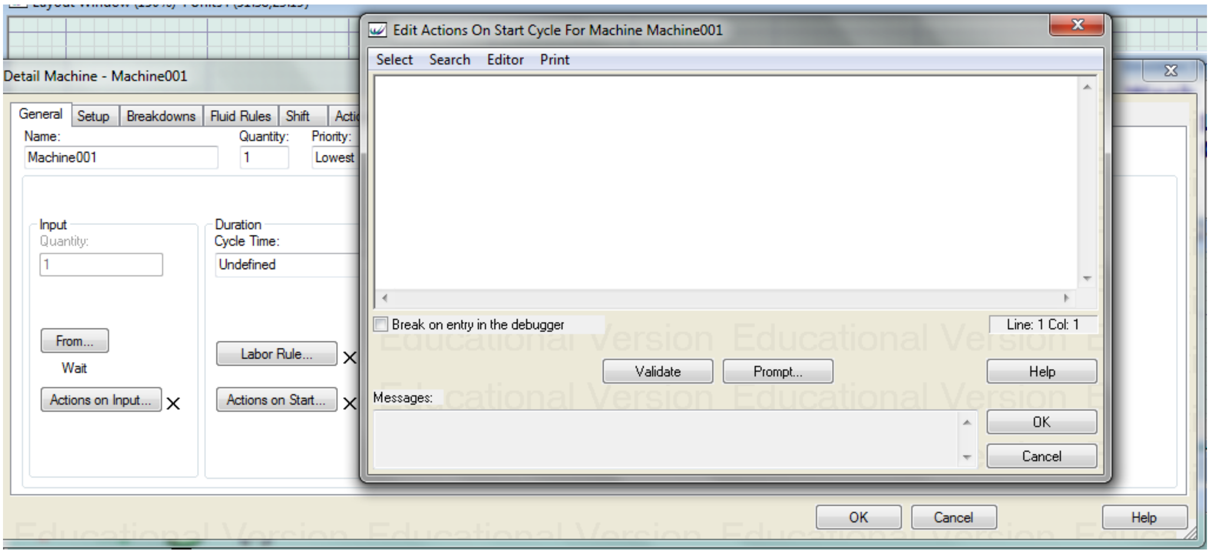


Figure 15 – Actions on box for a machine. All action on windows have the same appearance.

4. Part Flow

It is incredibly simple to set up the flow of parts through a witness simulation and is done so using just a couple of mouse clicks. The buttons to set up these rules are found at the top of the witness interface, position highlighted in Figure 7 (A) and detailed in Figure 14 below.

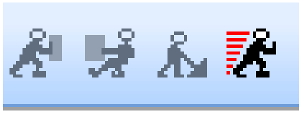


Figure 16 – From left to right; push rule, pull rule, allocate labor and quick rule.

To set up most part flows through the simulation it is necessary to use only the push and pull rules.

5.1 Push and Pull Rules

In order to move a part from one element to another, e.g. end of a conveyor to a machine, it is necessary to connect the elements with a push or pull rule. A push rule is used to push the part from A to B where as a pull rule pulls the part to B from A. Generally it does not matter which rule is used as they effectively do the same thing. It is worth noting however that it is best practice to pull parts where possible from a computing efficiency stand point. This is due to when the push rule is chosen the Witness software has to continually try to push the part at every time increment. The effect is negligible on one occasion but can make a difference when building large complex lines.

5.2 Connecting A to B

Not all types of elements in a simulation will connect directly with push and pull rules. Witness will only allow us to select the appropriate rules by only highlighting the suitable options. For example, in the above image, figure 14, only the quick rule is highlighted and therefore available use.

When using buffer elements it will quickly become obvious that only certain rules can be used. That is, it is not possible to pull parts in to a buffer or push them out. Instead it is necessary to use the element before to push parts in to the buffer and the element after the buffer to pull parts out. Figure 15 shows the route of a part through a buffer. Pushing parts from a buffer to a buffer proves a little more difficult and requires a dummy operation in the middle of the two buffers to push and pull appropriately. A conveyor is often used in this circumstance.

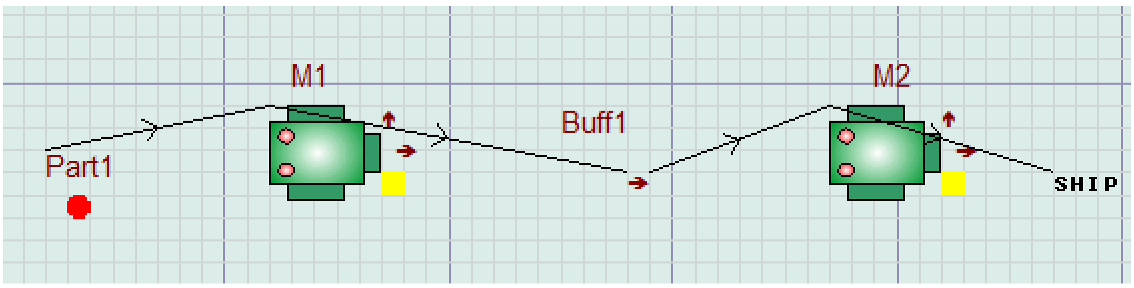


Figure 17 – Notice how the Part1 is pushed into Buff1 by M1 and pulled out again by M2

It is possible to view the route a part takes through the model by highlighting the elements of interest (click and drag) and pressing the Element Flow button as shown in figure 16.

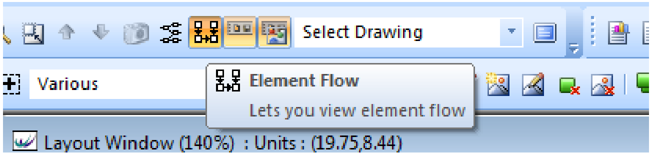


Figure 18- Element Flow button used to show path parts take through simulation

Machine elements are much simpler to use as they allow parts to be pushed in to them when empty and be pulled out of them when the cycle is complete. The same is true for conveyors.

Tracks can only push vehicles to tracks, not pull them from a previous track. Tracks can however pull and push parts on to the vehicles that are on those tracks.

5.3 Making Connections

The simplest way to connect elements within witness is to use the 2D representation of elements within the layout window.

1) Select the element that you wish to pull parts from or push parts to
2) Press the pull or push rule button on the toolbar
   (A window will appear)
3) Select the element which the part is to be pulled from or pushed to

The elements are now connected.

It is also possible to use the details page for a particular element and manually enter code. To do this open the details page for a particular element and under the general tab locate the 'From...' and 'To...' buttons. Clicking on the from button is the equivalent to setting a pull rule, the to button is the same as using the push rule. A text window will open allowing you to enter the rule; the template for such code is outlined below. It is worth noting that by default the 'wait' rule will be found within the window and should be deleted before entering the code.

#	*Pull Rules*	*Push Rules*
A	Pull from Machine1	Push to Machine1
B	Pull PartA from Machine1	Push PartA to Machine1
C	Pull from Machine1, Machine2	Push to Machine1, Machine2

Rule A will pull/push *any* part that resides in the operation. Rule B is an extension of A but with the constraint that only PartA will be pulled/pushed. Rule C will pull/push from/to Machine1 when available, however, when blocked or starved the part will be pulled/pushed from/to Machine2. Witness uses logical code and it is possible to use variations on above for example combining B and C gives:

Pull PartA from Machine1, PartB from Machine2

stating to pull partA from Machine1 unless it is not available in which case pull partB from Machine2. Once the code is entered click the validate option to ensure the code is valid – witness will actually validate the rule automatically when clicking Ok to save the code.

5.4 Other connection options

Wait – This is the default rule for push/pull and essentially just tells the element to wait until a part is pushed into it.

Pull from world – If a part has a passive part profile, i.e. parts are always available, then the pull from world command pulls a part from outside of the model.

Push to ship – Once a part has been processed through the model as required we can use the push to ship command to send that part out of the model.

Sequence/Next and Sequence/Wait – The sequence/next command is used to push or pull parts in a particular sequence such as pulling two of PartA then three of PartB. This will then cycle in this sequence. The syntax for such a rule is:

SEQUENCE/NEXT PartA Buffer1#(2), PartB Buffer2#(3)

If PartA is not available it will move on to the next part, PartB.

The Sequence/Wait is very similar to the /Next rule with the exception that when it cannot pull from/push to the first source it will wait until the part becomes available. This is useful for assembly operations when parts need to be pulled from off-line stock sources and when not available it is necessary to wait before the engine can be released.

Percent – The percent rule is used to push or pull a particular percentage of parts. This is useful for pulling a particular amount of a certain parts in to the model to simulate a particular batch mixture. The syntax for which is below:

PERCENT/1 PartA from World 25, PartB from World 75

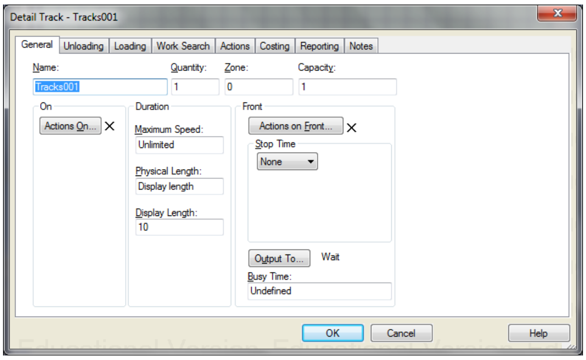
The above code would be used as a pull rule and is pulling the PartA and PartB into the model in the ratio 1:3 ratio (25% PartA, 75% PartB)

It is recommended that the Model A simulation (Page 165) is now built in order to reinforce the ideas studied this far.

5. Loading and Unloading Vehicles on Tracks

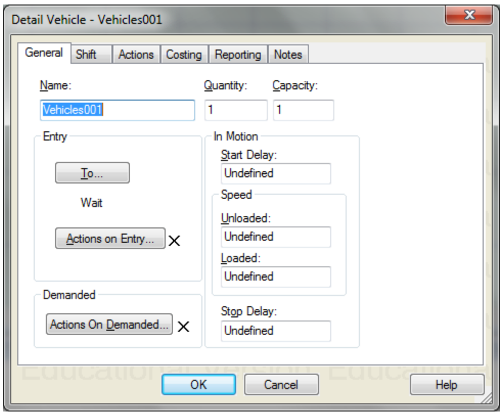
It is often desirable to use vehicles and tracks when modelling assembly facilities. This is especially applicable when modelling assembly lines which are constrained by a particular number of pallets/platens. It is possible to model a palletised line without using tracks and vehicles by constraining the number of parts that can be in the model at any one time. Buffers can be used with a minimum delay time to represent a transfer time between operations. Although this is a valid approach there is some complication introduced such as if there are two types of pallets in use or if there are different transfer speeds in different areas of the line. Tracks and vehicles can be used to ease this process.

<u>Setting up the Track</u>



The track can be used to constrain the number of vehicles that can be used on it. Thus it is recommended that the model is built in short sections of track connecting operations. This way the capacity can be set as the amount of parts that can be held between the two operations – representing buffering. The maximum speed of the track can be set such that if there is a speed limit on a particular track the set limit cannot be exceeded. When modelling the platinised assembly line it is most useful to set the maximum seed of the track to be the constraint i.e. the vehicles are capable of travelling faster than the limit of the track. This allows different sections of line to have different speed limits. The display length and physical length have been discussed before.

<u>Setting up the Vehicle</u>



The vehicle speed it set for two conditions, a speed for when the vehicle does not have a part loaded and a separate speed for when a part is loaded. This is often required and useful if vehicles have to travel slower when loaded due to health and safety. The entry point is entered and should be a track, when the model is run the quantity of vehicles, also input by the user, are all pushed in to this location. There is also the ability to model a start-up and stop delay although this is not necessary for the circumstance of platinised assembly lines. Different vehicles can be used on the same track if required.

Having inserted the tracks and platens it is now necessary to model the machines working on the parts. In reality on assembly lines parts can be worked on whilst still on the platen however when modelling with tracks and vehicles this is not possible. Instead within the model the vehicle, platen, has to be stopped at the end of the track and the part offloaded to a machine. The vehicle is then held at the end of the track until the machine has completed cycling at which point the part can be loaded back on to the same platen and continue along. In order to avoid confusion over buffer sizing it is recommended that a track of capacity 1 is input for the purposes of holding the platen whilst the machine is being worked on.

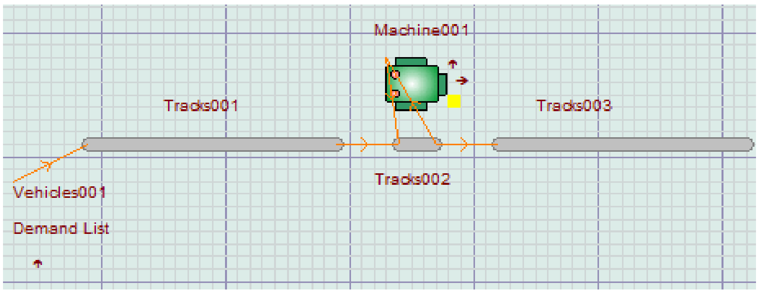


Figure 19 – Desired flow for elements via a machine. The part is offloaded from the vehicle to the machine and then reloaded

The settings for where the track pushes a vehicle to can be incredibly simple, or incredibly complex. NOTE: if the track has loading or unloading enabled but nowhere to unload to or load from, it will not push the vehicle to the next piece of track.

There are three options for outputting vehicles to tracks:

1) Push: this pushes vehicles to the track(s) that have been selected. In this mode, it will always push a vehicle to the first track on the list unless it is full, in which case it will try the next track.

2) Sequence: this pushes vehicles to the tracks that have been selected in sequence, with the number that has been selected (eg #(5) outputs 5 to that track) being pushed before moving on. There are two sequence modes, 'wait' and 'next'. In 'wait' mode, if the track that a vehicle is being sent to is full, the simulation will wait until there is space. In 'next' mode, if the track is full, the simulation will move on to the next track in the list until it finds one with space.

3) Percent: this setting pushes out to the selected tracks, and gives a certain percent of the vehicles to each depending on the percentages set in the 'push to' box.

These options can either be used straight and be carried out whenever a vehicle reaches the end of a track (and is loaded if necessary), or can be used in conjunction with IF statements.

Having connected all of the tracks it is necessary to now input the points where we want parts to be unloaded and loaded from – in the case of assembly lines this will be at all machines. There are a number of options that can be set in relation to loading and unloading vehicles. The two main areas that need to be considered are under what conditions the vehicle unloads and loads parts, and how many are loaded or unloaded.

The default setting for how many items to unload or load is 'all', which, as you would expect, unloads all parts the vehicle was carrying, or loads as many it can carry. If the vehicle capacity is set at 1, this is not an issue. However, if it can carry more than 1 part at a time, the simulations can become confused if it cannot unload both at once, and then cannot reload an item and continue. If this becomes an issue, you need to unload parts into dummy buffers, process them from there, then reload them onto the vehicle. Luckily in the case of assembly lines a platen generally only holds one part.

The two main options for setting the conditions for loading/unloading are 'always' and 'if'. 'Always' mode means that the vehicle will always load or unload, while 'if' mode will only load or unload if the conditions entered are met.

The two most common phrases used with the 'if' statement for loading and unloading are ISTATE and NPARTS. The ISTATE phrase lets you specify loading or unloading depending on the state of a machine, buffer or track (eg busy, empty, broken down), while NPARTS lets you specify conditions based on the number of parts present in a machine or buffer. The codes for each are:

ISTATE (elementname) = x

NPARTS (elementname) = x

The only other phrase needed to be used in the 'if' statement is for if you are using variables that are set after operations have been completed. In this case you can use the code

Variablename = x

If this is the case, if the variable value does not match, then no unloading or loading will be carried out.

NOTE: If unloading or loading are enabled, even if there is no specified input or output, the vehicle will attempt to carry this out at the end of the track, and will not proceed to the next track because it cannot.

When modelling platens on an assembly line we make use of the load all and unload all since the capacity is always set as 1 on the vehicles (platen). The transfer mode should be ‘Always’ in most cases where the part is to be processed by the machine. If the assembly line is a mixed model line then it may be necessary to use the transfer mode ‘if’ and make a decision on if the part requires processing. An alternative use is by leaving the transfer mode as ‘always’ but changing the cycle time on the machine to 0 if the particular part is not processed at that machine.

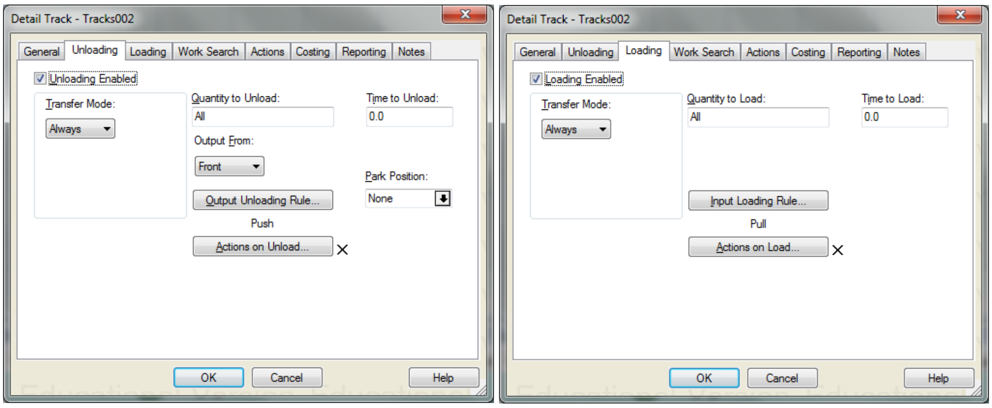


Figure 20 – Unloading and loading rules set on the track in front of the machine

An example of this process is discussed in Model B2.

6. Logic, Variables and Attributes

7.1 If rules

The basis of an IF statement is as follows:

IF this happens

Do this

ELSE IF this happens

Do this

Otherwise

Do this

End of IF statement

NOTE: You don't need to have the ELSE IF part of the statement; this is only used if you are wanting to carry out actions depending on multiple conditions.

When in use, the IF statement would look like this:

IF Variable = 1

Push to track1

ELSEIF Variable = 2

Push to track2

ELSE

Wait

END

There are a couple of important points when programming an IF statement. The first is that ELSEIF is written as one word with no space, otherwise an error will occur. Also, when typing the statement, you do not need to use the correct case of letter, as Witness will correct these when you confirm the statement. Finally, the Wait term can be substituted with an exclamation mark.

7.2 Variables

Variables are incredibly useful within a simulation. They can be used for a wide variety of things, from counting how many parts a machine has processed, to stating whether a buffer is full, or triggering/re-routing events.

There are a number of different types of variable that can be set: real (a decimal number), integer (a whole number), name (name of a Witness element) or string (text string). It is important to ensure you define your variable type correctly, or else it will cause errors later in the simulation.

For almost all uses of variables, the code used to set them is the same. The code used is generally set in something like an actions on entry or actions on front for a track, or actions on finish for a machine. The code should look something like:

```
IF PartAttribute = 1 or 2
ExampleVariable= 1
ElseIf PartAttribute = 3
ExampleVariable= 2
Else
ExampleVariable = 0
EndIf
```

In this example, the machine/track/buffer would look at the part, and examine the attribute called PartAttribute. If this attribute was 1 or 2 the variable ExampleVariable would be set to 1, if the attribute was 3 the variable would be set to 2, and if the attribute was anything else, the variable would be set to 0.

The only changes that would be made to look for other things would be to change the IF and ELSEIF lines to look at the desired element or attribute.

Counting Operations

Another use for variables is to count the number of operations that a machine has completed, or the number of items that have entered a buffer, etc. The below example would make a variable called VariableCount keep track of how many parts had been processed by a machine called MachineExample. This would be set in the actions on complete option of MachineExample.

```
VariableCount = VariableCount + 1
```

This command says that on the completion of a machine cycle, the system would add one on to the current value of VariableCount.

7.3 - Attributes: Fixed and Variable

Attributes are one of the many elements in Witness that do not usually have a graphical icon that is displayed within a simulation. Attributes within a simulation either allow you to set specific values that can be called up at certain points, or they can be variable, and changed at different points in the simulation. They are often used to make logic based decisions. These two types are called Fixed and Variable attributes.

There are a number of reasons for using attributes within a simulation. They can be used in conjunction with Functions to set different cycle times for different parts in the same machine; they can be used to denote whether or not a part has been through a certain process; or to assign labor in specific circumstances.

For both types of attribute, you must first create the actual attribute element before you can set any values for it. To create an attribute, right click in the main window, select 'define', and then select 'attribute' from the dropdown list. When entering a name, Witness will not allow you to create the element if another with that name already exists. Finally, select an attribute type: integer (whole number), real (decimal number), name (name of a Witness element) or string (text). Then, press 'create' and the attribute will be created. Once the attribute has been created, you can reference it in the simulation.

Fixed Attributes

Fixed attributes can only be set within the part options dialog box, and will then apply to all instances of that part within the simulation. To set fixed attributes, open the part dialog box, go to the attributes tab, and check 'Fixed attributes'.

Now you can enter the fixed attributes for that part which will apply globally. Let's imagine that you have two different parts passing through the same machine, and you have created an attribute called MachiningTime. In one part's fixed attribute list, you would enter MachiningTime = 0.5, and in the other part's list you would enter MachiningTime = 0.8. This allows us to set the machine cycle time to MachiningTime. Once the simulation was started, every time the first part entered the machine it would take 0.5 units of time to process, while every instance of the second part would take 0.8 units of time.

NOTE: As well as setting these fixed attributes, you also need to assign functions, and set a machine to look for these functions in order to set machine cycle times using attributes. This is further explained in the Function section.

As another example, let's imagine that we want to be able to check which of two parts is on a track passing through a definition point. By creating an attribute called PartReference we can set this. By entering PartReference = 1 in one part's attributes and PartReference = 2 in the other, we can use a variable to display this attribute as the part passes the definition point, and therefore know which part is present. (Further details of how to set a variable to check this are set out in the Variables section.)

Variable Attributes

Fixed attributes are set in a different way to fixed attributes, and it should be noted that the two cannot both be used on the same part. Variable attributes can change as a part progresses through a simulation, and therefore separate instances of the same part can have different values for the same attribute because of where they are in the simulation.

The easiest way to set variable attributes is through the various 'actions on' options in a simulation. These are available in a huge range of places, a few of which include: part actions on create, part actions on leave, machine actions on load, machine actions on finish, track actions on enter, tracks actions on front, etc.

Commonly, you may want to set a variable attribute as a part enters the simulation. Let's consider a simulation where there is an attribute called HasPartBeenMachined. In the 'actions on create' option in the part dialog box, you can enter HasPartBeenMachined = 0. Now, whenever a new instance of the part is created, it will carry a value of 0 for that attribute. Then, in order to know whether or not that part has passed through a machine, in the 'actions on finish' option in the machine dialog box, you can enter HasPartBeenMachined = 1. Now, if machined parts and un-machined parts are passing along the same piece of track, you can check what value they have for this attribute, and know whether or not they have been machined. More importantly, you can then use this for part routing using IF statements such as in the example below.

IF HasPartBeenMachined = 1

Push to Track_BypassMachine

Else

Push to Track_FeedsMachine

EndIf

In this example, if the part had been machined, it would be pushed to the track that bypasses the machine. If it has not been machined, it gets pushed to the machine.

To examine this in greater detail, check out the example file VariableAttributes.mod.

Fixed and Variable Attributes

Since fixed and variable attributes cannot be used in the same simulation, if you want to set different cycle times for different parts and also set attributes to say that parts have been processed by certain machines, you need to find a way round this issue.

The easiest way to get round this problem is that instead of inserting the various cycle times in the Fixed Attributes section, you can put them in the 'Actions On Create' area of the part. This will give the same results as putting the times in the Fixed Attributes area, but with the difference that these numbers could be changed at some point in your simulation. This means you need to be careful not to reference and change any of these numbers at any point during the simulation.

7.2 Functions

Functions in Witness allow the modeller to carry out a number of different actions. One of the main uses of functions is to make use of variable cycle times in machines based on fixed part attributes. This is necessary because you cannot call an attribute and directly apply it to something like a cycle time. You can also use a function to pull values from a variable and apply them to a process.

Applying a Function

To apply a function, first you need to right click, select define, and create the new function. Note that you do not set options such as function type at this stage; these are set after the function has been created.

Once the function detail dialog box opens, you can set items such as parameters, and the function type, choosing from real, integer, etc.

NOTE: Once you have created and detailed the function and it is in use, you WILL NOT be able to alter the function type or parameters.

Once you have entered these details, you can enter the Function Body by clicking on the actions button. Here you can tell the function to return details of a part attribute by using the code below:

```
RETURN ExampleAttribute
```

With this code, whenever the function is called, it will return the value of that attribute in the part being examined.

To make use of this function, you can then enter it so that a machine cycle time will be based on the values returned by the function. Open a machine dialog box, and in the cycle time box code resembling that below would be entered:

```
ExampleFunction ()
```

With this code, whenever a part enters that machine, the machine will call the function ExampleFunction and return whatever value it is set to lookup. Note that the brackets are needed after the function name or else Witness will show an error message.

7. Breakdowns and Modelling Variability

Distributions are useful within models for allowing items to make use of varying values, whether for machine cycle times, or for breakdown or setup times.

There are two different types of distribution that can be used within a simulation: continuous and discrete. For each of these two types, integers, real numbers or names can be used.

The difference between continuous and discrete distributions is that a discrete distribution will select only the values that have been entered, while a continuous distribution will select values that lie between the entered values at the given frequency.

Once you've defined the type of distribution, you can enter the data to be used. This can either be done manually, or can be imported from a file. If entering data manually, you enter a value in the Value box, the weighting of that value in the Weight box, and then click Add. Repeat these steps for all data points you wish to enter.

If you wish to update a value, select it, update the figures, and then click Update. To delete a value, select it, and then click Remove.

Once you have entered all desired data points, you can click Ok, and then reference the distribution in the desired element within the simulation.

There are many standard distributions that can be input in to models with the assistance of the distribution input wizard. In order to insert a standard distribution navigate to the model assistant in the element tree and then navigate to distributions, right click on the preferred distribution and select 'Insert with Distribution Wizard'. Witness will then assist the modeller with the selection of parameters by displaying information and a profile of the distribution.

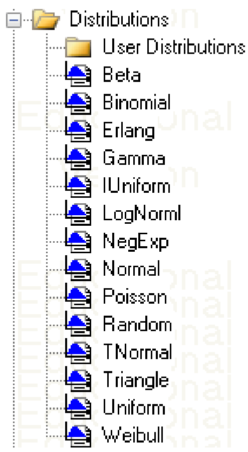


Figure 21 – Distributions available as standard within Witness.

It is also possible to use an external spread sheet to generate distributions as required by the user. This data can then be saved in a standard file known as a .DST file which can be opened by Witness.

8. Statistics and Analysis of Results

The statistics report is accessed for an element by selecting it in the element tree, right clicking on it and selecting the 'Statistics' option. This will open the statistics report for the element. An example of a statistics report for a buffer and a few tracks is included below.

WITNESS

Buffer Statistics Report by On Shift Time

Name	Buff1
Min Time	10.00
Min	0
Avg Delay Time	
Avg Time	816.29
Avg Delay Count	
Max Time	1665.00
Avg Size	16.66
Now In	34
Max	34
Total Out	1
Total In	35

WITNESS

Track Statistics Report by On Shift Time

Name	% Empty	% Busy	% Blocked	No. On
Tracks001	86.59	13.41	0.00	23
Tracks003	26.24	73.76	0.00	23
Tracks002	87.17	12.83	0.00	23

It is not possible to discuss all of the statistics and analysis that can be done on a model output. Instead the reader is recommended to review the following books and chapters. Robinson (2007) provides explanation through the use of case studies, whereas Law (2007) takes a more mathematical approach to explaination.

Robinson, S. 2007. *Simulation the Practice of Model Development and Use*. UK: John Wiley and Sons.

Law, A.M. 2007. *Simulation Modelling and Analysis*. 4rd Edition. McGraw-Hill, New York.

9. Other Features

9.1 Display

One of the strengths of creating and running simulations in Witness itself, as opposed to using Scenario Manager (discussed later) or an Excel interface, is that you can visually check that the simulation is running correctly and study particular phenomena, and what effect changes make to it.

If you use designer elements to create your simulation, then there will already be both 2D and 3D icons selected, which can then be updated as you desire. If you right-click on an element and select 'update graphic', then there are several 2D icons that you can choose from. If you then click the '3D' button on the right, you can change the shape and colour of the 3D icon.

If you want to resize a 2D graphic, then you can select it, hold Control, and drag one of the corner tabs to resize it.

You can also change what level an item is on, which takes effect in 3D view. This means you can stagger the heights of overhead conveyors and other elements to make it appear more realistic.

The big advantage of using the 3D view is that you can see parts while they are on a vehicle, something that is lacking from the 2D view. This means you can check that parts are being routed and unloaded/loaded correctly as they make their way through the model.

NOTE: Running 3D visuals is far slower than running the 2D equivalent (unsurprisingly). Also, if you run the 3D view then close it and go back to the 2D view, it will run far slower than previously. To rectify this, save and close Witness, then reopen your model and it will work fine.

Model A – Building your first model

In order to reinforce the basic model building principles within witness 12 a simple simulation will now be built. The aim of this model is to represent a single part being processed in a semi-automatic machine, by an operator, where the part is machined and then shipped. A buffer is used to store the parts before being loaded to the line – this buffer could be a pallet of parts for example.

Assumptions

For simplicity we will assume there is no transport required between the buffer and machine. We assume the machine cannot work without the labor elements presence, as required for a semi-automatic machine. There is a flow of parts entering the model at regular intervals. The parts arrive at a slower rate than the cycle time of the machine reflecting a bottle neck operation before the start of this model.

Elements Required

- 1 Part
- 1 Buffer
- 1 Labor Element
- 1 Machine

Details

The machine cycle time is 45 seconds and can only process one part at a time.

The buffer can hold a maximum of 60 parts at any one time.

The labor element is always available.

One part enters the model every 50 seconds.

Building the Model

1. Start by inserting all of the elements in to the layout window
   - Using the designer elements window click on the an element and then click anywhere in the layout window to place in that position
2. Arrange the elements in the correct order. Part -> Buffer -> Machine (with labor element)
   - Left click and drag to move elements within the layout window
   - Figure shows an example layout

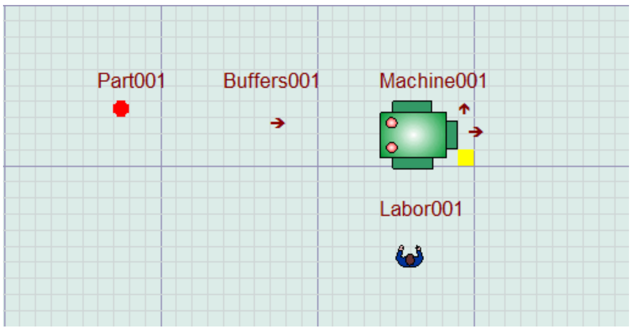
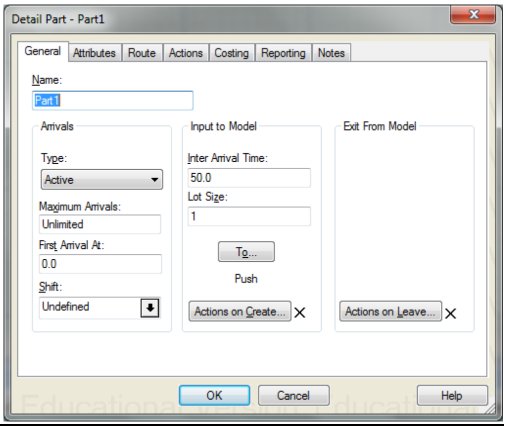
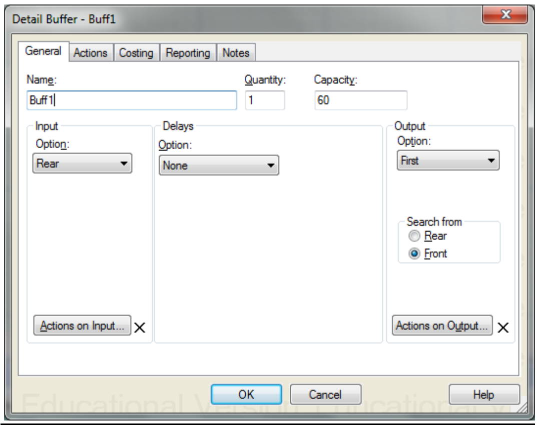


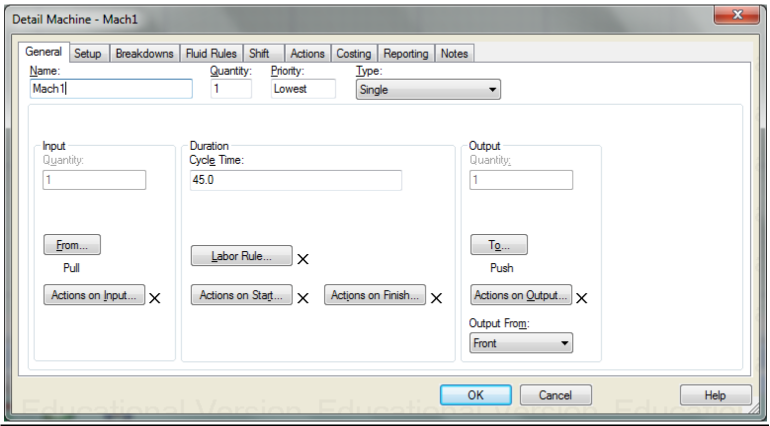
Figure 22 – View of layout window at stage 2

Note: it is not essential for elements to be arranged as shown providing they're connected correctly.

3. Detail the part, buffer and machine using the detail window
   - Double click an element within the layout window to open the details window
   - The part will need its arrival type set to active with an inter arrival time of 50 seconds
   - The buffer will need its capacity changed to 60
   - The machine will need its cycle time set to 45 seconds







- It is good practice to rename elements to identifiable but short names. For this case study the elements are named Part1, Buff1, Mach1 and Man1.

4. Assign the push and pull rules
   - The part will need to push itself into the buffer

- The machine will pull the part out of the buffer into itself
- The machine will then push to ship. Highlight the machine and click the 'push' button at the toolbar, simply select ship to push the part to ship, as highlighted in figure 21.

Figure 23 – Push window for Mach1

- Click the Element Flow button on the top toolbar to show the flow of parts through the model which should be as below in Figure 22.

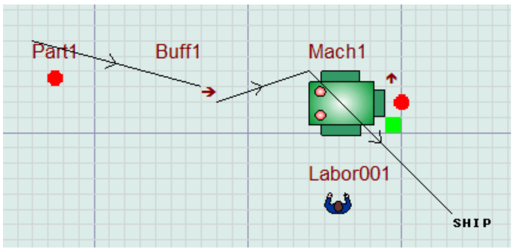


Figure 24 – Element Flow Diagram for Model A

5. Assign the labor element to the machine
   - Open the detail box for Mach1
   - Click Labor Rule to open the labor rule window
   - In order to assign the labor element we need to point witness to the desired operator, in this case Man1. Simply type Man1 in the text box and click ok.

6. Run the model
   - Set the run time to 10 minutes

   *Warning! Entering 10 in the simulation time refers to 10 seconds or 10 minutes depending on if the cycle time for Mach1 and inter-arrival time of Part1 are entered in seconds or minutes. Consistency is required throughout! In this model the elements are detailed in seconds thus the simulation time will need to be set as 600 seconds or simply 10 * 60 which witness will calculate automatically.*
   - We do not require a warm up on this model as the parts will be pulled in to the machine quicker than they enter the buffer.
   - Press the play icon to watch the model run in normal speed
   - Press the batch icon (fast forward) to run the model quickly with no graphical output.

7. Analyse the results

   - Right click on an element in the layout window and select 'Statistics' to open the report on that particular element. Verify that the below predictions are correct:

- Within 10 minutes 12 parts (=10*60/50) should have entered the model and the 13th just arrived

- These 12 parts will have entered and left the buffer
- The machine will have processed all 12 parts as it is constrained by the part inter-arrival time when entering the model and not the machine cycle time
- The machine will spend 10% of its time starved as the parts arrive every 50 seconds whilst the machine cycles every 45 seconds. This leaves 5 seconds with the machine starved per part.

There are a couple of measures found in the statistics window of a part which are not so easy to calculate arithmetically. These include the work in progress (W.I.P), average work in progress (Avg W.I.P) and the average time to process a part (Ave. Time).

Name	Part1
No. Entered	13
No. Shipped	12
No. Scrapped	0
No. Assembled	0
No. Rejected	0
W.I.P.	1
Avg W.I.P.	0.90
Avg Time	41.54
Sigma Rating	6.00

Figure 25 – Statistics window for the part in model A1

Figure 253 shows the WIP to be less than one reflecting the fact that the machine is processing parts quicker than the parts are entering the model and thus the machine spends time starving.

Model B – Modelling Transport

This case study will adapt model A to introduce a piece of transport to get parts from the buffer into the machine. Two models will be created, firstly with a conveyor belt system and then, secondly, a system of vehicles and tracks.

Model B1 – Conveyor Belt System

For simplicity the parts are considered to be entered on to the conveyor directly without the use of any platen. Thus there is no return conveyor required and will be modelled much like a conveyor belt in a supermarket – i.e. parts are removed from a buffer (trolley), entered at one end of the conveyor, processed by the till operator (semi auto operation) and shipped out of the model. The conveyor will be modelled as being indexed fixed i.e. every part has a fixed position on the conveyor and if the conveyor becomes blocked the whole conveyor comes to a stop and the parts remain the same distant apart. An indexed queuing conveyor however would allow parts to slide up the conveyor in the event of a blockage.

1. If starting from scratch, insert a part, buffer, machine and labor element in to the layout window and follow the instructions in Model A to construct a model.
   If starting from Model A, we will need to only insert a single piece of conveyor from the transport tab found in the Designer Elements window.
2. Position the parts in the layout window
   Click and drag to select the part and buffer in the dotted selection window, release and re-left-click on the part/buffer and drag to the side to make room for the conveyor. Drag the conveyor into place.

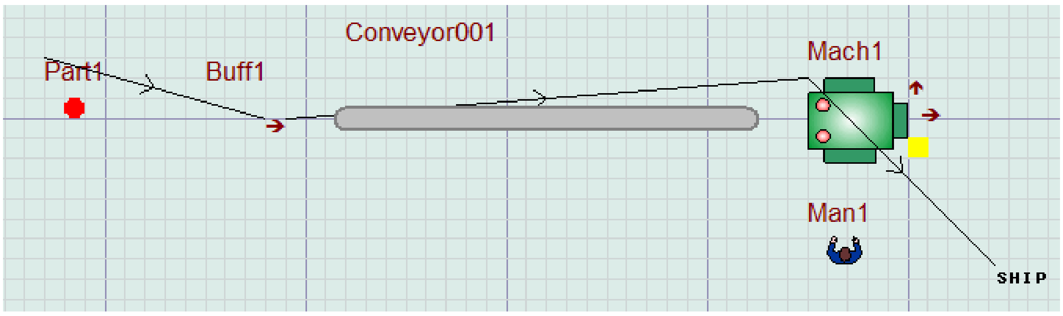


Figure 26 – Model at stage 2, notice how Buff1 is connected to Mach1 neglecting the conveyor.

3. Set up the Push/Pull rules to include the conveyor
   The conveyor will need to pull parts out of the buffer and the machine will need to pull parts from the end of the conveyor. It will be necessary to delete the current pull rule from Mach1 where it pulls from the buffer.
4. Detail the conveyor
   Change the Type to Indexed Fixed, length to say 5 parts and set the index time to say 10 seconds. This will mean it will take each part 5 *10 seconds to reach the end of the conveyor (assuming there is no blockages or breakdowns) since it will take 10 seconds to move from position 1 to 2, 10 seconds from 2 to 3 etc.
5. Run the model
   Set the run time to 10 minutes (600 seconds) and play through the model.

<u>Analysis</u>

Because the transfer time through the conveyor is equal to the inter-arrival time of the part and the Mach1 cycle time is less than this the conveyor will not become blocked. A part will be shipped every 50 seconds.
The introduction of the conveyor does however slow down the time between a part arriving in the model and transferring into the machine by 50 seconds. This results in a drop of the parts being processed – in model A in 10 minutes 12 parts were processed whereas in model B1 only 11 parts were processed by Mach1.
This is where a warm up period is sometimes useful, introducing a 60 second warm up means there will be sufficient time for the first part to have entered the model and transferred into the machine. Run this model again with a 60 second warm up time to confirm (don't forget to increase the run time by 60 seconds too!). You will notice there is now 12 parts processed within the 10 minute run.

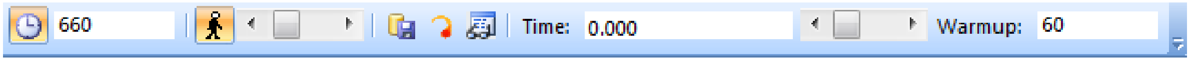


Figure 27 – Run toolbar set up with a warm-up.

The introduction of the conveyor (at the current transfer rate) has caused the average time to process the part by 89.8% as well as doubling the average W.I.P.

Name	Part1
No. Entered	13
No. Shipped	11
No. Scrapped	0
No. Assembled	0
No. Rejected	0
W.I.P.	2
Avg W.I.P.	1.90
Avg Time	78.85
Sigma Rating	6.00

Figure 28 – Statistics window of Part as in Model A

<u>Model B2 – Tracks and Vehicles</u>

It is now of interest to assess the effect of using tracks with platens circulating around it. Modelling the loading and unloading of platens in witness is done with a series of push and pull rules or modification of the details window. Unlike Model B1 this model will require a return route for the platens to circulate around after the part has been bought off. This walkthrough will begin with modifying model A.

1. Insert two tracks and a vehicle
   Insert two pieces of track and a vehicle from the transport tab found in the Designer Elements window.
2. Position the parts in the layout window
   Click and drag to select the part and buffer in the dotted selection window, release and re-left-click on the part/buffer and drag to the side to make room for the two tracks. Drag the tracks into place. Insert the vehicle anywhere within the layout window.

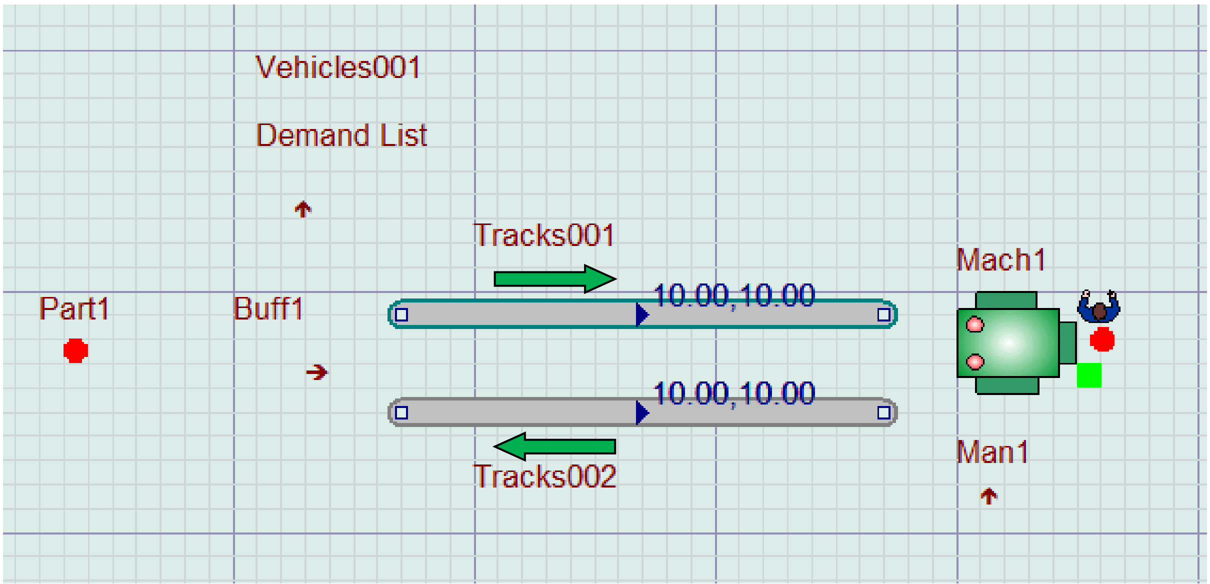


Figure 29 – Insert two tracks and a vehicle element and rearrange. Green arrows show the direction of desired flow

3. Re-orientate the return track for the platens
   The bottom track will return the platens to load further parts from the buffer. Figure 297 shows the proposed directions for the tracks with green arrows.
   To flip a track:
   a. Right click the track
   b. Select update graphic
   c. Click reverse

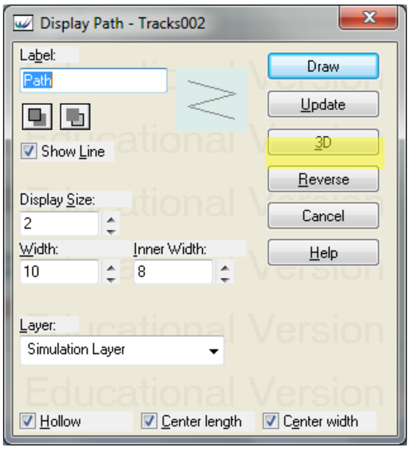
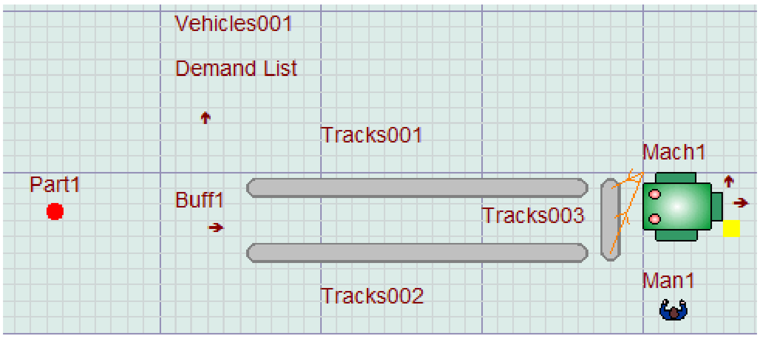
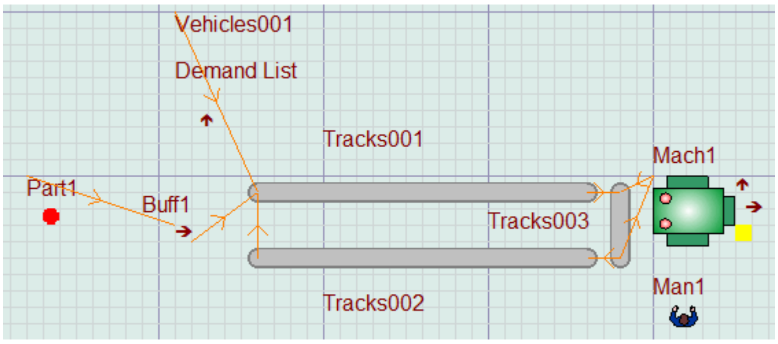


Figure 30 – Reversing a track. Display dialogue for a track

4. In order to model the off loading and loading of the part from the platen to the machine a third track is required as shown below. This track should then be set up to always offload and load parts to Mach1.

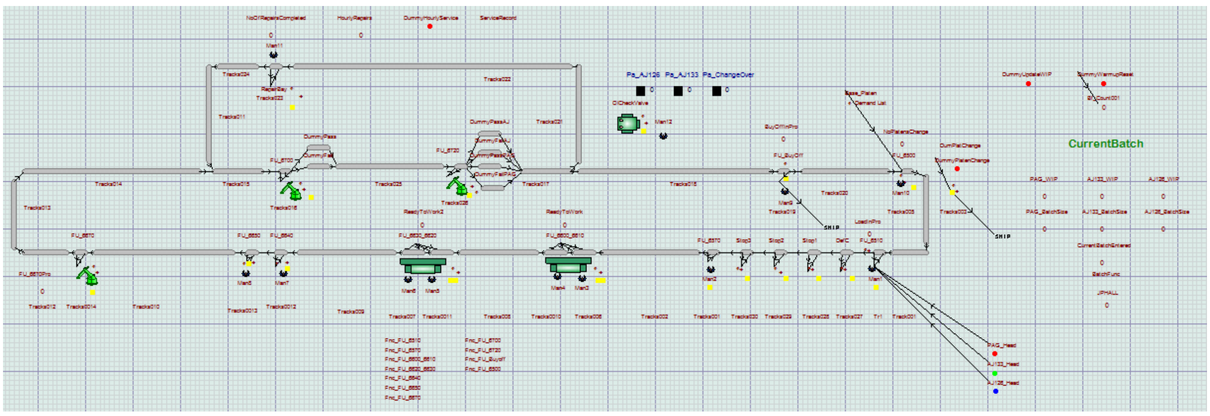


5. Connect all of the tracks in to a loop either by using the visual output rule or the ‘output to’ rule within the tracks element details. Set the details for the part and tracks and vehicles.
6. The numbers are not important as this example is to illustrate the approach used to create the model only. However, normally the vehicle speed should be set higher than any track and constrained down using the maximum speed within the track details. This allows different speeds on different tracks. The unloaded and loaded speeds should be set in the detail page of the vehicle.
7. A track should be selected to load the part from the buffer by using the transfer mode ‘always’.
8. The vehicles should be pushed to the track which will then circulate around the tracks. The model should look as followed. Notice element flow is activated to see the connectivity between entities.



9. The capacities of the tracks should now be set to match that of the buffering on the line and the quantity of vehicles increased to match the number of platens.

This is a basic example of how the use of vehicles and tracks can be set in a model of an assembly line using a platinised loop. This method can be used to build far more complicated lines such as the following model based on a real assembly line.



Effect of FTT calling group leader

1.0 Overview

- It was noted that wherever the FTT was applied, in the FAST Sigma Assembly Model, the utilisation of that operation increased.
- Initially thought to be due to the rework carried out by the test machine within repair loops where the FTT was applied.
- Moved the FTT application from within repair loop to an operation outside of the loop.
- Sensitivity analysis was then carried out. Changing the cycle time had no effect on utilisation of the operation - an unexpected result.
- Sensitivity analysis was then conducted on the FTT % applied which had a drastic effect – again, an unexpected result.
- FTT was causing change in utilisation due to the group leader being called to acknowledge every failure. This was wrongly being recorded in the utilisation of the machine as opposed to in a custom stat 'waiting auxiliary' and a misconception as to what happens in reality.

2 - Sensitivity Analysis Experimentation

For the purposes of this overview, the sensitivity analysis and results included will focus around 1 specific operation. The operation chosen, OpX70, was initially wrongly identified to be the bottleneck due to the error discussed within this case study. The experiments undertaken were carried out by making the changes outlined to the particular parameter in question using the FAST interface, running the model and gathering the results. The sensitivity analysis experimentation carried out surrounds the application of FTT.

Experiment 2.1 - 95.89% FTT on X70, 100% FTT on X80

Table 3

OP Name	Off Shift(%)	Cycling State(%)	Starved Primary(%)	Waiting Aux(%)	Blocked Primary(%)	Setup(%)
OPI_XOPX70	0.00	90.66	7.78	0.00	0.38	0.00
OPI_XOPX80	0.00	58.51	37.89	0.00	3.37	0.00

Experiment 2.2 - 100% FTT on X70, 95.89% FTT on X80

Table 4

OP Name	Off Shift(%)	Cycling State(%)	Starved Primary(%)	Waiting Aux(%)	Blocked Primary(%)	Setup(%)
OPI_XOPX70	0.00	64.93	8.63	0.00	26.41	0.00
OPI_XOPX80	0.00	87.91	5.85	0.00	4.87	0.00

2.3 Analysis of Results

As can be seen from comparing the results in Table 1 and Table 2, FTT was having a significant effect on opx70 cycling state. Changing the FTT percentage on the X70 operation, whilst holding all other factors constant, led to an approx. 25% increase in cycling state.

The apparent issue was investigated and identified. The code embedded in Witness through the use of FAST contained two issues. One was conceptual and the second was due to an error in the code. There was a misconception and the remote modelling team believed that in reality the group leader had to attend to every FTT failure that occurred. Thus the behaviour of the model had been adapted to match. This was an incorrect assumption. Further to this, the waiting auxiliary state remained at 0% in both models yet the cycle time was being changed. This identified an error in the states being used to record this process.

Experiment 3 – Corrected Model Results

The code within FAST was corrected such that the time waiting for the group leader was recorded within the 'Waiting Aux. State'. The conceptual error was left unfixed to allow comparison to the results in Table 1. The experiment was then repeated and the results are included for completion in Table 3.

Experiment 3.1 – 95.89% FTT on X70, 100% FTT on X80

Table 5

OP Name	Off Shift(%)	Cycling State(%)	Starved Primary(%)	Waiting Aux(%)	Blocked Primary(%)
OPI_XOPX70	0.00	66.39	7.74	25.19	0.44

It can be seen by comparing Table 1 and Table 3 that the cycling state has dropped from 91% to 66%. In turn, the Waiting Aux state increased by 25%. This experiment confirmed that the recording of states within FAST had been corrected.

The misconception regarding the attendance of the group leader at all FTT failures was also later corrected although the results are omitted.

Conclusion

The experiment outlined within this, brief, case study highlights the importance for thorough verification and validation to be undertaken. Two errors were identified through the use of sensitivity analysis on cycle time and FTT:

1. The incorrect recording of machine states
2. The incorrect assumption regarding group leader duties

The impact of these two errors on model performance was substantial. When corrected, the model JPH increased by around 11% and a change in the location of the bottleneck was observed. Due to the use of the FAST interface these issues were not isolated to one model, but all models created utilising the FAST tool.

Overall this case study highlights the method of sensitivity analysis as a quick method of verification which can highlight errors in both operational and conceptual modelling.

Appendix 4 – A Case Study: Verification via False Cycle Time

The following case study includes an overview of the method outlined within the body of this thesis in Chapter 4. The aim of this method is to identify issues within individual elements, their attraction and the resulting statistics. In particular attention is focussed on the resulting graphs and conclusions that can be drawn. A case study for AJ assembly line is reviewed. The aim of the methodology is to reduce the complexity of assembly line output as displayed in the below, initial, graph:

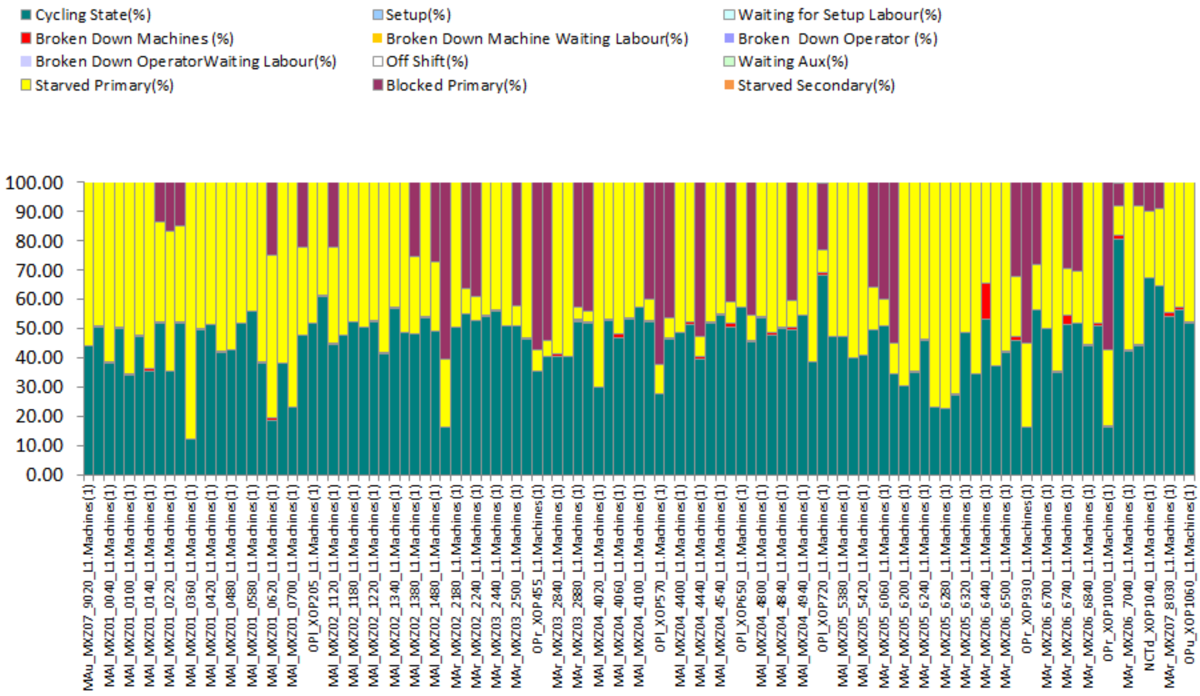


Figure 1- Standard Model Output. Stacked bar chart.

A4.1 Building the Model

When building the model it is recommended that attention is firstly dedicated to the correct positioning of elements within the correct order. Effort should be made to ensure capacities and buffer lengths are input as required. The input of cycle times should be delayed until later in the process. In particular the following parameters are suggested on the first pass of building the model:

FAST Input	Input Effected	Required Change
Buffer Length	Unaffected	No Change. Build as layout.
Sequence of Machines	Unaffected	No Change. Build as layout.

Number of derivate	Batch input	Reduce to 1
Variation	Quality, breakdowns & stock handling	Disable

In the case of the AJ assembly line the model was adapted such that one derivative, Engine 1, was selected as it was the majority. This detail is not essential as any could be selected as long as set to 100%, this removes the variation caused through different derivative having different cycle times at different stations.

Standard		**Adapted**	
ENGINE TITLE	BATCH (%)	ENGINE TITLE	BATCH (%)
3,	34	3,	0
2,	22	2,	0
1,	44	1,	100

Having built the model and verified the buffer sizing is correct attention turns to application of false cycle times. It is required that the model is built with flat cycle times to remove interaction between different elements and standardise the utilisation of all machines on the line. For the case of the AJ assembly line an overview of the standard cycle times is included below:

Team Number	**Section Balance Time (Seconds)**	**Number of Manual Op's**	**Number of Automatic Op's**
1	91	8	2
2	91	11	2
3	91	11	0
4	91	7	2
5a	91	4	2
5b	91	6	0
6	91	7	2
7a	99	7	1
7b	99	5	0
8a	103	4	0
8b	103	7	0
9	99	9	3
10	99	0	1 + 2 SA cold test stations
11	91	2	1

This cycle time was then flattened such that all the automatic and manual operations within Teams 1 to 11 all run at 91 seconds as the false TAKT time. When using this approach the cycle time of line automation, such as turntables, should be flattened to a percentage, say 50%, of the TAKT time, in the case of this example they were adjusted to 9 seconds representing 10% of the standard TAKT time. This allows the interaction between operations and line automation to be later inspected. The Cold Test cells were addressed separately due to there being 2 stations in parallel. This allows the cycle time of each cold test cell to be twice the TAKT time and maintain the required production rate. The ratio of the different stages of the test was maintained.

	Before	**91 Second Model**
Run Time (mins)	1.937	1.7820
Rig Time (mins)	0.818	0.7520
Derig Time (mins)	0.543	0.5000
TOTAL (mins)	**3.298**	**3.034**

A similar approach should be used on all operations undertaken in parallel.

Overall the following changes were made to the AJ assembly line:

FAST Input	**Input Effected**	**Required Change**
Buffer Length	Unaffected	No Change. Build as layout.
Automatic Stations	Cycle Time	Set to TAKT time
Manual Stations	Cycle Time	Set to TAKT time
Cold Test and Hot test Stations	Cycle Time	Set to TAKT time.
Parallel operations doing the same job	Cycle Time	Balance appropriately* across the number of operations.
Line Automation e.g. Turntables	Cycle Time	Set to 50% of TAKT time
Sequence of Machines	Unaffected	No Change. Build as layout.
Variation	Quality, breakdowns & stock handling	Disable

* Here it is suggested if there are N identical machines completing the same job in parallel then each machine can run at N*TAKT and overall meet the same TAKT capacity.

A4.2 Results

The model was then run. It is not necessary to conduct the model for run length analysis since there is no variation and the model is deterministic in nature. As a result the model used in this study was run for 3 days to give sufficient data output. The following stacked bar charts were then created:

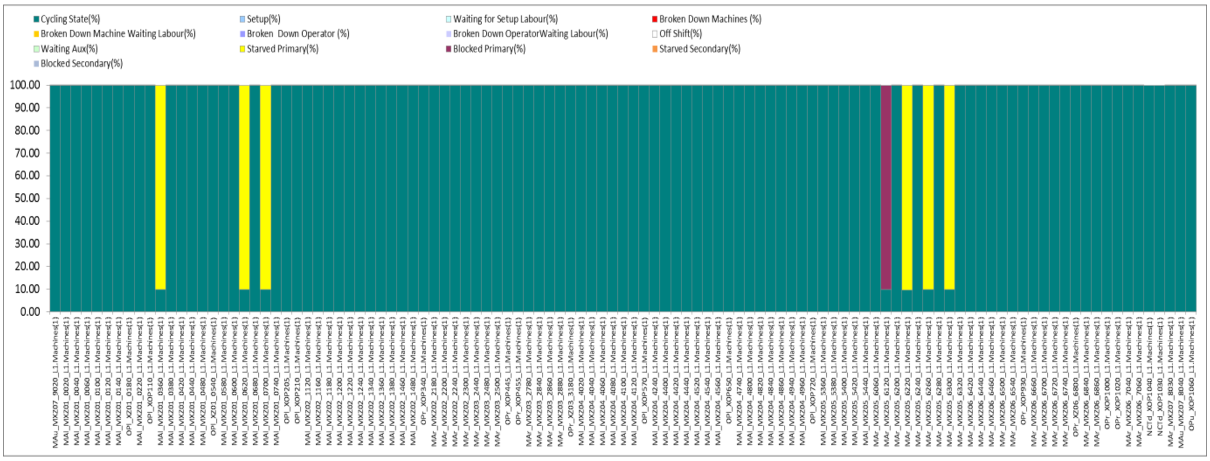


Figure 2: Graph highlighting minimised complexity through use of method

The x axis contains the names of all the operations on the assembly line and the y axis contains the cumulative percentage of each stacked state. The details are too small to visualise however it is the shape of the graph we are most interested in at this stage. Green displays the cycling percentage, yellow the percentage idle and the purple represents the blocked state.

As can be seen from the stacked chart the majority of the operations show 100% utilisation with a uniform shape. This is to be expected since all automatic and manual operations, which make up the majority of the line, are all cycling at exactly 91 seconds. Thus, no machine can work faster than another and therefore they all cycle 100% of the time. There are seven operations which deviate from this 100% cycling state and all represent the automation, such as the turntables. As would be expected, the percentage cycling of these automation operations are indeed not more than 10% cycling state.

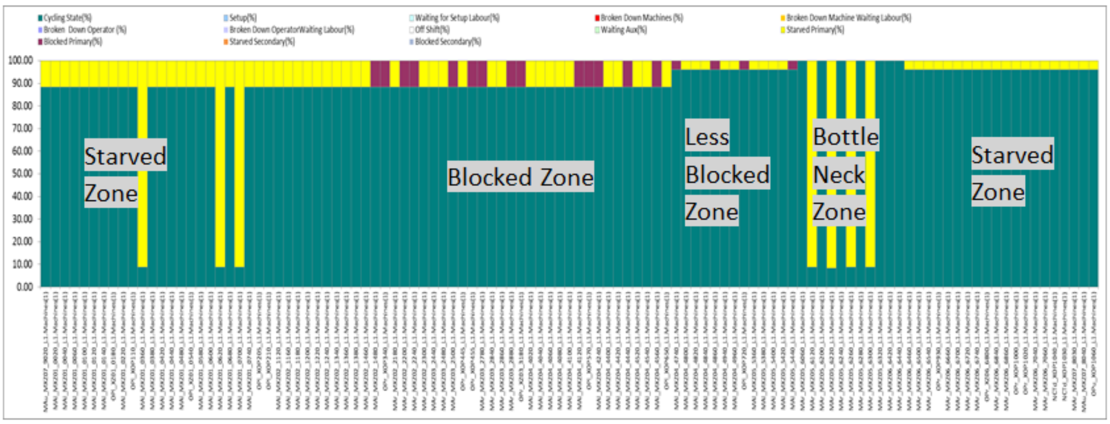
However, the purpose of adding the complexity of these automation operations at lower TAKT time allows for the interaction to be investigated. The approach outlined highlighted an error in the results as displayed in the stacked chart. The group of operations to the right of the graph, above, show an error in the recording of the states on the lower utilised automation operations. The low cycle time entered on the automation followed by the increased cycle time on the automatic and manual operations causes a predictable operation which can be considered conceptually. We would not expect the operations to show the statistics displayed. In particular the large amount of starved combined with zero blocked highlighted an error regarding the output rule of the machine and an operational error contained with the FAST model.

A4.3 Conclusion

The approach described provides an overview of the method of applying false TAKT times as a means of validation. The example highlighted in this case study identified an issue with the states recorded on results which had implications in the bottleneck analysis of all models built with FAST. Previous uses of this approach have identified errors where there variation was not being removed as expected and push and pull rules.

Overall, although further work is required to develop its application, the method has been and can be used for validating complex assembly lines by reducing the inherent complexity. The visual method used and minimal time investment required allows this method to be used during verification and validation stage of all models created.

Future work, includes adapting the method for considering optimum cycle time distributions and gaining better understanding of the interaction between designed bottleneck zones as introduced in the below graph:



Appendix 5 – Daily FTT Report Example

	14/10/2013					Target
SIGMA ASSY F.T.T. DAILY				81.39%		82%
No Of Engines Tested				2623		
Repair Loops		F.T.T.				
	EVENTS		Target	Better/ Worse		
Loop 1	57	97.83%	97.00%	B	+	0.83%
Loop 2	63	97.60%	97.20%	B	+	0.40%
Loop 3	24	99.09%	96.50%	B	+	2.59%
Loop 5	73	97.22%	96.70%	B	+	0.52%
Loop 6	9	99.66%	98.20%	B	+	1.46%
Loop 7	23	99.12%	96.20%	B	+	2.92%
Loop 8	103	96.07%	96.20%	W	-	0.13%
Loop 9	19	99.28%	96.20%	B	+	3.08%
Loop 10	25	99.05%	99.50%	W	-	0.45%
Loop 11	98	96.26%	98.80%	W	-	2.54%
Loop 12	39	98.51%	98.80%	W	-	0.29%
Total Assy Events	533					

Appendix 6 – Station Level First Time Through from QMS

(Operation names removed for confidentiality)

Number	Zone	Description	Quality	RTY
X05E	7	Main Side - Xloop Elevator	100	1.000
X05L	7	Main Side - Xloop Lowerator	100	1.000
X30	7	Auto	99.71	0.9971
X40	7	Auto	98.3	0.9830
		****ENTER REPAIR BAY X1****	98.01	0.9801
X60	7	Fit	99.93	0.9993
X60S	7	STAND_BY STATION	100	1.000
X70	7	Rundown	99.31	0.9931
X80	7	Apply	100	1.000
X85	7	Fit Bolts	100	1.000
X90	7	Fit Bolts	100	1.000
X110	7	Rundown	99.11	0.9911
X130	7	Fit	100	1.000
X135	7	Snug	99.99	0.9999
X140	7	Torque	99.99	0.9999
X150	7	Gun Start	99.98	0.9998
X155	7	Fit	99.97	0.9997
X160	7	Rundown	99.3	0.9930
X170	7	Rundown	99.35	0.9935
X190	7	Auto	99.92	0.9992
X200	8	Torque	99.99	0.9999
X210	8	Fit	99.99	0.9999
X250	8	Rundown	99.79	0.9979
X230	8	Fit	99.96	0.9996
X240	8	Fit	99.98	0.9998
X270	8	Fit	99.98	0.9998
X280	8	Assemble	99.99	0.9999
X290	8	Assemble	99.98	0.9998
X310	8	Fit	99.93	0.9993
X330	9	Fit	99.96	0.9996
X360	9	Assemble	99.95	0.9995
X370	9	Fit	100	1.000
X380	9	Lube	99.82	0.9982
X220	9	Feed	99.27	0.9927
X400	9	Rundown	98.6	0.9860
X410	9	Rundown	99.27	0.9927
		****ENTER REPAIR BAY X2****	93.5	0.9350
X435	9	Assemble	99.99	
X440	10	Fit	99.98	
XX10e	10	Xloop side - Xloop Elevator	100	
XX10l	10	Main Side - Xloop Lowerator	100	

Removed to ease experimentation

Appendix 7 – Novel approach to modelling operator rotation

1. Insert two elements; 1 machine and 1 part
2. Change the part to active arrivals and push to machine. Push the part directly from the machine to ship
3. Set the part to arrive first at 60 minutes and then at 60 minute intervals thereafter. This could also be set to vary slightly if required
4. Set the machine cycle time to the desired distribution. This models the operators variability
5. Set the machine labour rule to all of the operators rotating on the line. Ensure not to select the group leader. If preferred use an individual dummy machine per operator, this adds complexity but allows greater variation
6. Set the priority of the dummy machine to the highest in the model
7. Activate pre-emption in the machine labour rule such that it is highest priority and also able to pre-empt work from the machines. This models the drop tools attitude.
8. Set pre-empt level to 1. Providing the machine priority is set to the highest in the line, as in Step 6, the operators will always respond to the rotation task.
9. Additional variation can be added in allowance and time penalty which modifies the drop tools attitude (if sufficient data is available). Both of these operations can be set to distributions, empirical or standard.
10. Visual display can be enhanced, and is recommended, by inserting string variable and using actions on start/finish to inform of when on and off shift. It is further recommended that the machine icon is either changed or removed to avoid confusion with non-dummy machines.