Instance-Based Model for Predicting Total Fabrication Duration of Industrial Pipe Spools

by

Cristian Petre

A thesis submitted in partial fulfilment of the requirements for the degree of

Master of Science

in

Construction Engineering and Management

Department of Civil and Environmental Engineering
University of Alberta

Abstract

Industrial fabrication for modular installations has its own set of challenges that combine the environments of industrial manufacturing and off-site construction. This hybrid execution strategy means that fabricators need to look at both fields and adopt the best tools and techniques.

This thesis presents an investigation into the improvement of delivery time estimates of industrial fabrication of a pipe spool fabrication shop in Alberta. The main contribution of the work is in the area of predicting the total fabrication duration to be expected in order to assist fabrication shop management in planning for appropriate workforce availability and material delivery date requirements.

In order to address the objective of improving the prediction of fabrication durations, the spool manufacturing process has been modelled for simulation. However, the data required to validate this model was found to be time-consuming and cumbersome to capture at the required level of details. Alternatively, the development of a data-driven knowledge discovery experiment was pursued. The approach employed was to utilize the fabrication information that was already being captured by the manufacturing facility and evaluate it using instance-based classification.

In addition, an effort towards the integration of manufacturing tracking and scheduling estimating is presented. This part of the work will ensure that the schedule is not consulted only at the beginning of a project, but throughout its completion. Updating a schedule with live fabrication progress data will allow production managers to update their completion date estimates and adjust the manufacturing plans to reflect existing issues such as material delivery and labor shortages or performance.

Acknowledgements

First and foremost, I would like to express the deepest appreciation towards my supervisor, Dr. Yasser Mohamed, who has been very supportive and helpful throughout my research and the writing of this thesis. I consider myself very fortunate to have been chosen to pursue my Master's studies under his guidance. I am deeply grateful for the considerable patience and personal guidance he has offered for the entire duration of my program.

I would also like to thank my defense examining committee members, Dr. Aminah Robinson Fayek and Dr. Marwan El-Rich for taking the time to review and critique my work and offer their feedback during my defense.

I would like to present my deepest gratitude towards my partner, Brittany, who has been my closest supporter through the difficult process of writing this document and helped me to persevere and push myself every single day.

Table of Contents

List of Figures

List of Tables

Glossary of Terms

Term	Definition
Acorn	Fabrication drafting and tracking software
Control Number (CN)	Unique identifier for a pipe spool
Diameter Inches (DI)	The sum of the pipe diameters to be welded, expressed in inches
Fit Complete (or FitC)	The date when the fitting process was recorded as completed
Material Issue Date	The date the required material has been delivered to the fabrication shop
Material Issue Report (MIR)	The fabrication package issued for fabrication; sometimes the date when this report was released is abbreviated with only MIR instead of MIR Date
Material Transfer Report (MTR)	The report compiled when a spool is transferred away from the fabrication shop, to either the module yard or to be shipped; sometimes the date when this report was released is abbreviated with only MTR instead of MTR Date
Maximum Pipe Diameter	The diameter of the largest piping item within a spool
Number of Items	The total number of components that make up a pipe spool
Package Issue Date (MIR Date)	The date when the fabrication drawings package is created in Smart Plant; also known as Material Issue Report Date
Pipe Spool	A sub-assembly consisting of pipe pieces and special elements called fittings that is used in an industrial application either as a stand-alone assembly or as part of an installation module
QC Complete (or QCC)	The date when the Quality Check (QC) process was recorded as completed
Smart Plant Materials (SPM)	Material management and tracking software
Visual Inspection	Spool assembly, alignment and weld quality visual inspection; also referred to as Quality Check or QC

CHAPTER 1. Introduction

1.1 Background

Modularization of construction has an increasing share in today's fast-evolving world, with an unusually high utilization in the case of remote worksites, such as the oil sands developments in northern Alberta. It is more convenient, faster and more economical to pre-fabricate and assemble components of buildings and processing plants off-site, in locations closer to major cities. Aside from the advantage of a milder climate and the opportunity to work in more controlled environments, the availability of local workforce can be used as a significant competitive edge (Bedair, 2013). A particular type of modular construction is that of piping structures, whereby pre-fabricated sections of pipe, called pipe spools, are organized and attached to a steel structure to speed-up on-site assembly.

1.1.1 Management of Industrial Fabrication

Core responsibilities of a project manager are to coordinate the delivery of a quality product on time and under budget. All three of these aspects are inter-related:

- Delivering a quality product the first time eliminates the need for rework, which would drive costs up and result in schedule slippage.

- Keeping a project on schedule removes the need for rushing the remainder of the project which in turn can lead to a higher repair rate and increased costs associated with overtime or lack of productivity resulting from over-sized delivery teams.

- Staying on budget throughout the project eliminates the need for cost cutting at later stages, which affects both product quality and timely delivery.

The success of a project depends, therefore, in no small measure, on its timely delivery, typically defined by an initial or baseline schedule. The baseline schedule, however, is generated at the beginning of a project when many details regarding the project execution are still uncertain. It is, therefore, difficult to provide an accurate, detailed schedule, and estimators rely for the most part on experience to derive activity durations. Being a time-intensive task during project planning, duration estimating is typically performed at an installation or work package level, ignoring detailing the plan down to an activity or task level.

The planning and coordination of each activity and their sequencing depends primarily on project coordinators or trade foremen, which use their experience to plan daily work, aiming to achieve the overall work package schedule. The estimation of completion percentages or actual delivery dates relies therefore upon said coordinators or foremen, which have to use their experience to report to project managers.

As most manufacturing facilities are beginning to pursue lean manufacturing techniques, this research can be considered to address two of the most important aspects of the lean process implementation, namely, waste elimination and customer satisfaction (Lean Enterprise Institute, 2016).

One kind of process waste is time, whereby improvement can be operated either by reducing activity times or waiting time between activities; both of these result in a reduction of lead time, but by far the most impact can be obtained by controlling the time an item spends waiting for work activities. By closely planning the execution process, the throughput of different processes can be levelled more efficiently, and the waiting time of products reduced.

Customer satisfaction is often expressed in terms of the quality of the product but equally as important is the timely delivery of products. In construction or fabrication projects, a baseline schedule is either imposed by the customer or agreed upon between the client and contractor at the beginning of the project. However, given the dynamic nature of material delivery, workforce availability and productivity, as well as weather or any other environmental factors, it can be very difficult to abide by the baseline schedule if this is not detailed and flexible enough.

1.1.2 Pipe Spool Fabrication

In the context of industrial manufacturing, shop fabricated elements are assembled into modules before being shipped to site. Some of the module components to be fabricated are in some cases pipe spools. Pipe spool fabrication typically happens in a fabrication yard or shop where heavy equipment, such as overhead cranes or submerged arc welding machines, is utilized to assemble sub-components for on-module or on-site installation (Aecon, 2012). The spools need to be ready before they are required for module fabrication, but too large a stock build-up would increase the need for storage and double handling. Since the factors affecting the time required for spool fabrication are very diverse, this results in a very uneven distribution of manufacturing durations, leading to difficulties in deciding upon the right time to start the work. The process of spool fabrication involves several steps as summarised in Figure 1-1 and further explained below. The acronyms are explained in the Glossary of Terms.

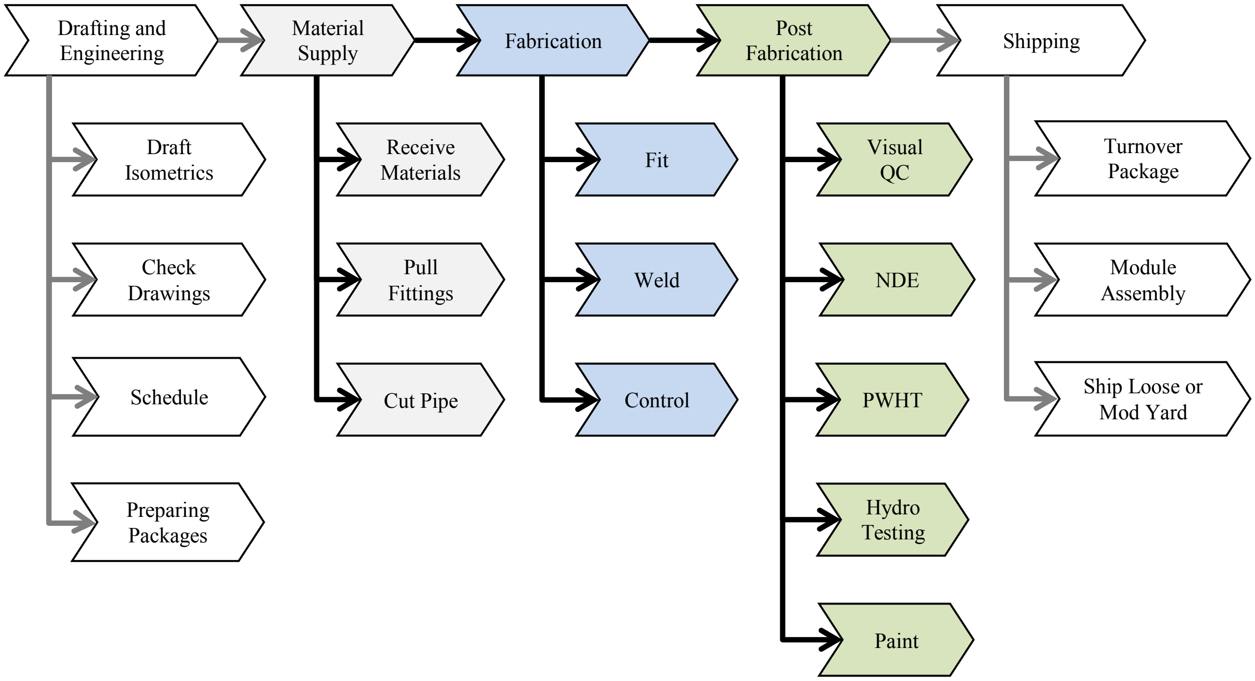


Figure 1-1 Spool Fabrication Processes

The Drafting and Engineering process involves preparing the project for fabrication by breaking it down into smaller, more manageable segments. The client CAD drawings are translated to isometrics with the additional project and manufacturing processes information added for ease of referencing. These drawings are then checked for errors and fabrication procedure details. The higher level construction schedule is broken down to a manufacturing package level. The bill of materials, isometric drawings and other information pertinent to any of the fabrication process downstream are put together in a fabrication package. The date the material required for each package is reserved for that particular set of spools is recorded as the Material Issue Report Date (MIR Date), representing the start of the fabrication processes.

The Material Supply process begins once all the tasks in the Drafting and Engineering category have been completed. The step called “Receive Materials” refers to the process of verifying, storing and recording all the material bulk material shipments that arrive for each project. The fabrication package cannot be issued until all the material required for each package has a status of available in the material management database. Once the package has been prepared it is passed on to the Material Supply group. A material picker uses the bill of materials to assemble on carts or pallets all the non-pipe components such as elbows, valves, supports, etc. (called fittings) and identify and cut the required pipe to the specified dimensions using the cut summary sheet. Once all the fittings are on pallets or carts and delivered to the fabrication shop, and the pipe has been cut and marked with a spool identifier, the remaining documentation, such as the isometric drawings and the package schedule, is passed on to the fabrication manager or foreman.

The Fabrication Processes represent the primary component of the value-added work but, unfortunately, they rely heavily on the previous stages and do not typically produce goods that can be delivered to customers without post-fabrication processes being completed. That being said, the fabrication consists of the processes of fitting and welding the components to produce the spools. The fitting process involves assembling the pieces according to the drawing and specifications; the assembly requires the edge preparation (grinding), alignment and tack welding actions to be completed before fixing the spool subassemblies to an electric roller for the roll welding process. If the spools are too big to be rolled welded, one or more position welds will be required, where the subcomponents are aligned using pipe stands, tack welded and prepared for a welder to fill in the gap without moving the spool out of alignment. The process of position welding is considerably more time intensive than that of rolled welding; therefore considerable efforts are typically taken to reduce the required number of position welds. Research in the area of minimizing the number of position welds is already well advanced (Hu & Mohamed, 2014). The Quality Control (QC) inspection team might get involved to check the alignment of the spools during the set-up process to ensure a quality product that conforms to specifications.

Once the fabrication processes have been completed, the Post Fabrication checks and auxiliary processes begin, starting with a detailed visual inspection of the spool regarding its dimensions, alignment, and weld quality. This process is complemented by Non-Destructive Examination (NDE) which can be performed either on all the welds or a sample of them, depending on project requirements. The NDE testing is typically based around the x-ray technology, used to analyze the quality of the weld throughout its cross section, but other such technologies are emerging that aim to eliminate or reduce radiation exposure and increase the analysis speed.

Depending on the type of material and pressures that the spool will be subjected to during its operating life, a Post Weld Heat Treatment (PWHT) also known as Stress Relief may be required. This process brings the weld and the surrounding area of metal to a temperature point designed to relax and harmonizes the internal structure of the metal which has been subjected to high temperature gradients during welding.

The last type of testing performed is called Hydro-Testing and involves pressurizing the spool to a factor greater than its design operating pressure (typically around 1.5) to ensure the welds can withstand the design pressure loads. Although this test is required for all spools subjected to a pressurized environment, the testing can be performed as part of the overall fabrication of a spool or after its installation in a module or directly in the facility. Hydro-testing is not typically required during fabrication in the case when a number of spools will be welded together after being placed in the module or shipped to the assembly site because they will be tested together after the final on-site welds have been completed.

Once all the testing is complete, the spools may undergo sandblasting and coating in a process performed by the Paint Shop unit. Each spool has specialized testing and painting requirements. In addition, certain post-fabrication process require batching of a group of spools before proceeding, leading to further uncertainty being introduced to the process of scheduling.

The shipping process involves tagging each spool with a unique identifier tag and preparing the turnover package which contains copies of all the documentation utilized during fabrication or testing. Once the turnover package and the spools themselves are ready, the spools are either placed in a piping module in the Module Yard or shipped to the construction site individually (also referred to as Ship Loose).

1.2 Problem Statement

In a pipe spool fabrication environment, it is customary to determine the fabrication durations heuristically. When estimating activity durations, little or no consideration is given towards the implications of shop loading, workforce availability and productivity or number of hours worked in a day. It is typically assumed that as long as there is sufficient float allocated in the schedule, the fabrication shop managers and coordinators will adjust the aforementioned factors to compensate for the variations in the type and volume of spools that need to be fabricated in order to meet the construction execution or module assembly schedule.

One of the management aspects identified as requiring improvement at the Pipe Spool Manufacturing Facility is the out of sequence delivery of products, which sometimes results in deadlines being missed. From a production standpoint, scheduling is a key to the success of a timely project delivery. In addition, it has been identified that the lead time of the spools cannot easily be accurately approximated using heuristic rules for estimating fabrication duration. This is in part due to the difficulty in considering the impact of workforce availability on fabrication duration. These planning errors lead to late delivery of some of the pipe spools and contribute to low customer satisfaction while other spools are ready well in advance of their required date and put pressure on the storage and handling capacity of the facility.

Another critical factor that significantly affects the operation of fabrication shops is the availability of the required materials. If the materials are not available before the fabrication process is scheduled to begin, the disruption in the shop operations will result in overall delays for project completion that are more significant than the material delivery delays on their own. On the other side, having too many of the materials available too early will overload the capacity of the material management and supply group servicing the fabrication shop. This problem is harder to quantify but it can be determined with relative ease whether the delays were the responsibility of the manufacturing facility or the client procurement and material management team. It is, therefore, desirable to have well-defined lead time estimates and be able to quickly alter the baseline schedule to reflect delays that can be attributed to the material delivery. Also, if several spools are critical to the overall progress of the installation project, identifying them and their estimated fabrication duration may allow production managers to change the fabrication priorities and compress the lead time of said spools by eliminating waiting times.

1.3 Objectives

The primary objective of this research is to improve the estimate of the duration to be expected for the completion of pipe spools before they are being fabricated, with the aim of enhancing the accuracy of predicting delivery dates. In addition, the delivery date estimates are expected to be more accurate once partial fabrication milestones are reached; therefore, the integration of updating these estimates within the fabrication control and reporting system is also pursued. The benchmark against which the proposed methods will be compared is represented by a heuristically determined fabrication schedule, defined at a task level.

In order to achieve this objective, the following research questions will be addressed:

- Can modelling and simulation provide an adequate framework for lead time estimation using process modelling and resource constraints?
- Is it possible to collect fabrication data at a level of detail required by a process simulation model?
- Can data-driven models using instance-based classification provide a verifiable improvement in the fabrication duration estimate over the heuristic method?
- How much improvement can be gained by utilizing instance-based models?

1.4 Methodology

The overall structure of the methodology of this study is summarized in Figure 1-2 with a description following thereafter.

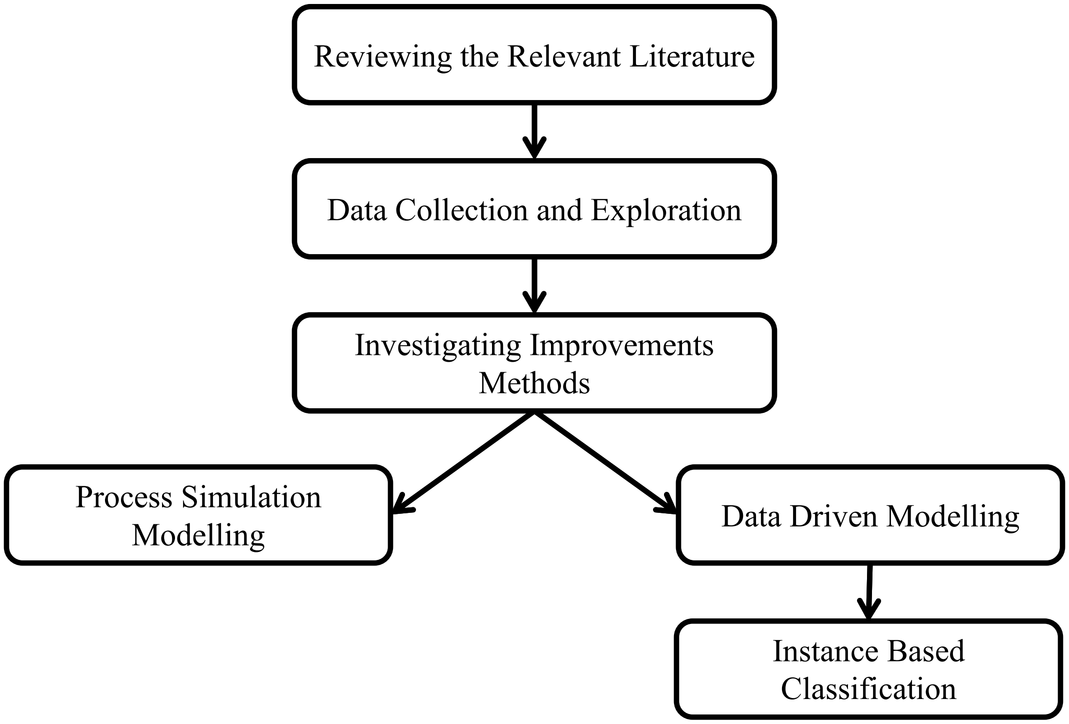

Reviewing the Relevant Literature
Data Collection and Exploration
Investigating Improvements Methods
Process Simulation Modelling
Data Driven Modelling
Instance Based Classification

Figure 1-2 Study Methodology Summary

First, a review of the relevant literature was carried out, covering areas related to fabrication scheduling, simulation modelling, and data mining methods. Next, the process by which the information was compiled, collected, processed and explored is described, followed by the investigation into the improvement methods which can be employed to arrive at the study objectives. The fabrication processes and their interaction was modeled using Discrete Event Simulation (DES) in a fashion that replicates the application using artificial data to show the structure and detail of information needed in order to calibrate such a model. A second modelling approach that can be integrated with the existing drafting, scheduling and tracking data structure was developed and used to show the shortcomings regarding the current quality and composition of collected information. Lastly, a knowledge discovery approach based purely on the most reliable features of the data was developed which demonstrates that the primary objective can still be achieved in spite of the challenges faced.

In order to evaluate the performance of the different models and be able to compare them against existing fabrication scheduling practices, the output of the improvement efforts needs to be evaluated in a consistent way with fabrication tracking data. For the purpose of this comparison, the total fabrication duration, defined from the package issue date (MIR date) to the delivery date (MTR date), was chosen as the main metric, with the estimate of the delivery date as the principal deliverable. The suggested methods also estimate the same total duration as the pre-existing heuristic practices and the evaluation is performed based on the closeness of the estimate to the observed fabrication duration computed from actual observations. Several statistical performance measures such as the relative absolute error and correlation coefficient have been considered in order to quantify whether the proposed methods can bring a significant improvement to the fabrication duration estimates when compared against the observed durations.

1.5 Thesis Organization

The current chapter has presented the background of the identified problems in the spool fabrication scheduling and management field tackled by this study, as well as the study objective and methodology.

The second chapter presents a review of the literature that is relevant to the problems, methods and means employed in the thesis. It includes sections on stochastic scheduling, the integration of scheduling and production tracking, modelling and simulation concepts employed, as well as data mining tools and techniques.

The third chapter showcases the investigation into process driven modelling and simulation, as well as a schedule derivation that integrates with the previously developed production scheduling and control too. The fabrication process model is detailed to the task and resource level that incorporates spool design parameters to determine tool time at each station and waiting time for each spool; in addition, this model showcases the potential implementation of different fabrication scenarios and the integration of equipment breakdown.

The fourth chapter begins with the description of a system developed to integrate the fabrication shop's data collection systems with the heuristic schedule. The development of this data integration system was a pre-requisite for the achievement of the objectives of this study. Besides, it includes a description of the parameters selected for the data-driven knowledge discovery and the cleaning and validation of the data itself.

The fifth chapter addresses the development of a data-driven knowledge discovery experiment using instance-based classification, preceded by an investigation into attribute selection and shop loading estimation. The classification exercise is designed to evaluate different attribute subsets, optimize two types of classifiers of the nearest neighbour family and showcase their potential application as a dynamic prediction tool, completed by a feature selection investigation.

The last chapter presents a summary of the thesis, concluding remarks, recommendations, as well as limitations and potential applications of this work.

CHAPTER 2. Literature Review

Creating a schedule for a spool fabrication environment presents unique challenges by combining aspects of linear production scheduling within a stochastic environment with high product variability. For the purpose of this review, the main problem is split into two distinct categories, that of stochastic scheduling on one hand, and spool manufacturing in itself on the other. In addition, a review of the literature about the concepts and tools utilized in the study such as data mining and simulation concepts is presented herein.

2.1 Stochastic Scheduling

Sarin, Nagarajan, & Liao (2010) define stochastic scheduling as "*dealing with problems when at least one of the parameters is not known with certainty*". Spool fabrication is a prime example of an environment where both the processing time of the job and its arrival time are not known with certainty. Research in the field of stochastic scheduling can be categorized based on its focus on three main categories:

a) the impact of processing time uncertainty on scheduling

Determining the starting time of each product is the main goal of scheduling in a manufacturing environment. However, complex processes and process interactions with a high degree of variability lead to difficulties in estimating processing time with accuracy. The need to develop a scheduling strategy to replace a traditional schedule under uncertain conditions has been identified in the early days of job-shop scheduling research. McKay, Safayeni, & Buzzacott (1988) used a survey of scheduling practitioners to identify the inability of traditional scheduling theory to account for extreme variations in processing times. As research progressed, the analysis of process scheduling uncertainties did not cease to be an area of interest (Li & Ierapetritou, 2008).

b) methodologies to deal with different manufacturing environments

Most researchers in the area have exemplified their work using a certain type of environment, such as: single machine, parallel machine, job shop and flow shop models. For example, Daniels and Kouvelis (1995) address the issue of scheduling robustness using a single stage system. Beraldi (2006)used a scenario tree to model how uncertain parameters evolve and derive a multistage model for parallel machine environment. Jackman, de Castillo, & Olfasson (2011) found similarities between a flow shop machine environment and ship lock operations and therefore employed flow shop scheduling to create a model for the Panama Canal.

Zhu and Wilhelm (2006) have reviewed literature related to the four types of machine configurations mentioned above, but their perspective also included lot sizing. Another comprehensive review of these methodologies is provided by Sarin, Nagarajan, & Liao (2010), together with their comprehensive analysis of schedules for each type of environment.

c) methods to address the issue of variability of performance measures

One of the main drawbacks one encounters when evaluating these existing methodologies and models is the variability of the performance measures employed. Although it is true that different environments or methodologies are better tested using custom performance measures, the resulting inconsistency makes comparing these models quite difficult. However, several researchers have attempted to solve this problem by suggesting more robust performance measures. For example, scheduling robustness has been identified by Daniels & Kouvelis (1995) as a more appropriate performance measure than evaluating processing times in a stochastic environment while Kempf, Bogza, & Maler (2013) have presented a programming procedure for approximating expected durations under stochastic uncertainty.

2.2 Applications of Stochastic Scheduling and Data Mining in Construction

Data mining approaches have been employed in construction applications for over a decade, for example in the work carried out by Soibelman and Kim (2002) towards the identification of schedule delays in construction projects. Their work identifies the data preparation step as pivotal to the success of the knowledge discovery, stating that *"Domain knowledge and a good understanding of the data are key to successful data preparation."*

Knowledge discovery techniques such as the ones presented in this section have been employed with various goals in several manufacturing applications. Li et al. (2013) use regression models and other similar techniques with the aim of describing relationships between dispatching rules and running state of a manufacturing facility. Piroozfard et al. (2014) present the use of a genetic algorithm approach in combination with simulation for solving job shop scheduling problems; an example application of this method to a spool fabrication shop is presented in the work by Moghadam et al. (2014).

Chien, Hsu and Chen (2013) employed principal component analysis and data mining techniques to extract manufacturing intelligence for use in fault detection and classification at a semiconductor manufacturing facility. With an application towards manufacturing flow times and process yield, Braha, Elovici and Last (2007) employed the technique of cost-sensitive classification to monitor and control a fabrication line.

The advantage of using a nearest neighbour classification approach to the problem of estimating time delays in an industrial processing plant has been presented in the work of Stockmann, Haber, & Schmitz (2012), who identified that such a method can be automated.

The use of simulation with the scope of optimizing production planning has been employed in works such as that by Gansterer, Almeder, & Hartl (2014), who aligned their framework with the concept of hierarchical planning. Aiming to solve the problem of manufacturing planning performance assessment under high product variability conditions, Ponsignon & Mönch (2014) introduced a simulation-based approach using both rule-based and genetic algorithm assignment procedures.

2.3 Integration of Scheduling and Tracking

Due to the nature of the uncertainties that are present in the industrial fabrication environment, a good schedule is a dynamic schedule, whereby as soon as uncertainties are reduced or removed, estimates are updated to create an improved schedule. Although some authors, such as Rokni & Fayek (2010) deal with uncertainties such as processing times and system configurations in a comprehensive way, they do not address the issue of the dynamic integration of these and other production parameters back into a live manufacturing production schedule. This integration has been attempted however in the construction scheduling environment using simulation modelling as presented in the research by Xie, AbouRizk, & Fernando (2011) but the work does not mention closing the feedback loop back into a dynamic schedule and is applied to a production environment (tunneling) where product features do not change or have very little change during production. Other research that relates to the integration of progress reporting and schedule updating is found in works such as that of Oliveros & Fayek (2005), where a fuzzy logic model is used to integrate monitoring and control in construction projects.

The scheduling and production control integration in a manufacturing environment has been briefly addressed in relation to the concept of reactive scheduling in works such as the one by Li & Ierapetritou (2008), but the focus remains around machine breakdowns and change orders and does not deal with reacting to other parameter uncertainties such as resource availabilities and processing times. Other authors touched upon the issue in research related to operations scheduling and production simulation (Pereira & Santoro, 2011).

Having considered the aspects of stochastic scheduling and outlined the relative lack of integration between scheduling and production tracking, the need for further research in the area of production integrated stochastic scheduling has been outlined.

2.4 Modelling and Simulation Concepts

Modelling is the representation of a system or process for the purpose of investigating it over a period of time. Systems or process representations can take a variety of forms and functions, so identifying the appropriate method for representation is one of the most challenging aspects of modelling. The approach to follow is to first identify the purpose of a model and its desired outputs before deciding on an appropriate representation (Birta & Arbez, 2013). In addition, from a project collaboration and communication perspective, the most suitable modelling concept is the one that can be understood by all the parties involved in the process.

On the other hand, the process of simulation typically refers to the implementation of the output of the modelling exercise as a computer program in order to study how it responds over a period of time. Estimating project completion time using simulation modelling has been applied to stochastic resource–constrained project networks in works such as that by Sadeghi, Fayek, & Ingolfsson (2012). In the field of Pipe Spool Fabrication, simulation has been employed effectively with respect to different objectives, such as

- studying the impact of changes in process flow and experiment with different production systems to evaluate their performance before any physical changes would be made in the fabrication shop itself (Wang, Mohamed, Abourizk, & Rawa, 2009).

- using simulation experiments to evaluate the impact of automating the process of identifying the optimal fabrication sequencing by assessing effectiveness against the traditional shop foreman approach (Hu & Mohamed, 2014).
- evaluating the development of a framework for optimizing shop scheduling using concepts such as fuzzy set theory and multi-objective optimization (Rokni & Fayek, 2010).

2.5 Data Mining

Data mining can be defined as “the study of collecting, cleaning, processing, analyzing, and gaining useful insights from data” (Agarwal, 2015, p. 1). Advances in technology and computerization resulted in the collection of increasingly large amounts of raw data from every aspect of life. This abundance of collected data however does not yield any insights into the system from which they were derived without significant processing of the information. Data can come in a multitude of formats, sizes or dimensions, all of which must be understood and carefully considered when selecting the appropriate steps along the data mining process. Data structure design, as well as data collection and storage typically happen before the mining process but a feedback loop can be created in order to facilitate the extraction of knowledge from a system.

The process of analyzing the available collected data begins with a two stage preprocessing phase, namely: a) extracting the data from all the available sources, combining it all in a structured format that is suitable for computing, followed by determining the required features for analysis and b) a data cleaning stage which consists of checking the data for errors and dealing with missing or incomplete values and outliers. Reducing the data to a compact, workable set containing only the required dimensions and complete set of records is very important in order to ensure a consistent analysis phase. The data cleaning and feature extraction is application specific and as such will be discussed separately, but the outlier detection and removal stage of data preprocessing is described further below.

2.5.1 Outlier Detection and Removal

In order to ensure the data to be analyzed is consistent, it will be necessary to investigate whether it contains records that vary significantly from the rest of the data, called outliers, which may indicate data that has been recorded erroneously. Evaluating the results of a normality test of the data should provide a good indication regarding the level of confidence in outlier detection methods that rely heavily on the assumption that the data follows a normal behavior. In addition, the selection of an appropriate outlier identification method involves some assumptions about the type of distribution the data will follow. Since at this stage the distribution has not been determined, a method that relies on the normality assumption as little as possible should be employed.

It is common practice to identify possible outliers using the Z-Scores and testing against a threshold. However, in many cases this method does not provide a robust enough detection and may identify too many or too few records as outliers. It has been suggested that modified Z-scores provide a more robust method for outlier identification (e-Handbook of Statistical Methods, 2015). A modified Z-score proposed by Iglewicz & Hoaglin (1993), as presented in Equation 1 has been utilized for the data set at hand, with an upper limit for the absolute Z-scores of 3.5

$$M_i = \frac{0.6745(x_i - \tilde{x})}{MAD}$$ **Equation 1**

where MAD = mean absolute deviation

$\tilde{x}$ = median

The method presented above will be implemented to all the fields that will be deemed to require an outlier detection method, whether that may result in the removal of records from the analysis or just their identification.

2.5.2 Types of Mining Algorithms

Depending on the domain and type of data upon which the analysis is performed, as well as the desired outcome, different mining techniques are suitable to different situations. Association pattern mining and data clustering methods have at their core the analysis of similarities between instances with the goal of finding potentially useful patterns. Data classification, on the other hand is used to predict the class or label of a multi-dimensional record based on a training data set (Agarwal, 2015, p. 285). Here, a test record with an unknown class is compared to the records in a training set and classified based on similarity. Algorithms used in data classification are also referred to as supervised learning methods as the evaluation of a test record is being carried out comparatively to other previously classified instances. The mathematical algorithms for classification are typically designed around two stages, called training and testing. The training stage is the construction of a mathematical model based on the labelled groups in the training data set, whereas the testing stage is that of assigning a test instance with one of the class labels.

Classification algorithms, also known as classifiers, come in different shapes, such as decision trees, rule-based classifiers, probabilistic classifiers, neural networks or instance-based classifiers. Selecting which type of classifier to use for a particular problem depends on the type (numeric or nominal) and number of attributes to be classified and whether or not distinct and accurate relationships between data can be derived.

Lazy Learning Classifiers

A particular type of classifiers, also called lazy learners, differentiate themselves by not constructing a classification model up front based on the training set, but instead simply analyzing similarities between a test instance and the available training instances and deciding on a class based on similarities (Witten, Frank, & Hall, 2011). Variations of this type of classifiers are based on how an algorithm determines the distance between similar instances, also referred to as nearest neighbours, as well as how many instances (k) the algorithm will consider and how the results will be aggregated to decide on the class of the test instance. The general notation for this type of classifiers is KNN and variations of this algorithm are commonly referred to as K Star (also seen written K*).

In order to determine proximity to the test instance a distance function is required, and as identified by Witten, Frank, & Hall (2011) "most instance-based learners use Euclidean distance". Euclidean distance between 2 instances x and y with n attributes is represented by Equation 2. Since Euclidean distance is a second degree L_p-norm distance, the most common variations involve changing the powers to which the numerical difference between attributes is raised. It is also worth noting that the L_p-norm distance degree also refers to the degree of the root for the sum (Agarwal, 2015). Some attribute weighting coefficients can also easily be implemented in the distance function, in case certain attributes should be given more importance over others.

$$\text{Dist}(x,y) = \sqrt{\left(x_1-y_1\right)^2+\left(x_2-y_2\right)^2+\cdots+\left(x_n-y_n\right)^2}$$ **Equation 2**

To avoid some attributes from dominating the distance computation because of scale differences, the attribute values need to be normalized so their values lie between 0 and 1 as per Equation 3. For nominal attributes, the distance to be considered is 0 if the values are the same and 1 if they are different (Witten, Frank, & Hall, 2011).

$$\text{norm}(x_i)=\frac{x_i\text{-min}(x_i)}{\max(x_i)\text{-min}(x_i)}$$ **Equation 3**

The K Star Classification Algorithm

At the base of this clustering algorithm sits the implementation of entropy as a distance measure between instance attributes (Cleary & Trigg, 1995). In essence, entropy as a distance measure involves determining the probability of transforming one instance to another over the shortest path (Horibe, 1973). The concept can be extended to a measure of the interdependence between finite probability spaces (Guiasu & Reischer, 1979). Further manipulating this concept, the K Star algorithm employs as a distance measure the negative logarithm of the sum of probabilities of transforming over all possible paths, as expressed in Equation 4. This particular distance computation was developed in order to avoid sensitivity to small changes in the instance space.

$$\text{K}^*(\text{b}|\text{a}) = -\log_2 \text{P}^*(\text{b}|\text{a})$$ **Equation 4**

In addition, the authors of the K* algorithm attempt to deal with both real and nominal values by expanding the method above two-fold: first, devising a function to compute the distance between integers, and then modifying it to work with real numbers. Given that the data set of this application will only have symbolic and integer attributes, the algorithm expansion to real numbers will not be described herein.

Furthermore, the method of dealing with missing attribute values can either be selected by the user or the program defaults to a setting treating missing data as a random value – this does not add inaccuracies in the computation because of how the algorithm utilizes the probability of transforming from on instance to another as a distance, rather than the distances between values themselves. Another benefit of the K Star algorithm is the ability to assign different weights to neighbouring instances based on their influence on the instance to be classified.

2.5.3 Feature Selection

Although domain knowledge is the main factor in deciding the set of attributes in the data to be analyzed, each attribute's weight in a machine learning algorithm varies primarily with the class to be analyzed. Features may present interdependence which can be exploited to derive conjunctive features that have the potential to enhance the performance of certain algorithms. On the other hand, certain variables might not be relevant at all to the certain searches and as such they may introduce unwanted noise in the data that will reduce the performance of the analysis. Decreasing the computation time is another benefit of reducing a dataset to its most relevant components, as is the identification of attributes that may have a significant contribution to overfitting. These are only but a few of the potential benefits of a feature selection exercise (Guyon & Elisseeff, 2003).

Several algorithms have been developed independently in order to identify the interaction between features and their correlation with the analyzed class in works such as that of Zhao & Liu (2007), with significant breadth and depth of implementation of such algorithms in works such as the University of Waikato's machine learning group (Hall, et al., 2009). The process of selecting attributes can be performed either independent of the classification method at the pre-processing stage, or specifically for certain classes or types of classifiers (Brownlee, 2014). The implementations of such feature selection algorithms aim to improve the performance of learning methods by reducing the dimensionality of the data either independent of the scheme selection or specifically for a particular classification method. When working independently of the classifier scheme the filter method is applied for dimensionality reduction or other types of data cleaning such as replacing the missing attribute values with the median or mean of the population. On the other hand, the wrapper method is a dedicated feature selection tool customizable in order to take into account the particular machine learning algorithm that will be used for learning.

2.5.4 Model Evaluation

In order to compare different learning methods to each other certain systematic techniques need to be employed to determine the performance of a model and its capability to reliably evaluate new information once it has been created. Statistical tools that represent the performance measures are presented in the following subsection, followed by a discussion on assessing a model's validity and prediction capabilities.

Performance Measures

In order to evaluate the performance of a numeric prediction for a set of values – be it a heuristic estimate, classification output or any other type of duration estimation – various measures and statistical tools can be employed, as presented in equations 5 through 9. Here, the actual observed value is represented by the x variable and the predicted value by the y variable, with n the number of instances present.

The **Mean Absolute Error** (or MAE for short in Equation 5) is the average of the magnitude of the individual errors without taking account of their sign. In the case of predicting the spool fabrication durations, the lower the value of MAE the better the result is. However, this performance measure is not easily assessed or intuitive unless one has knowledge of the typical or average duration of fabrication.

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n}|y_i - x_i|}{n}$$ **Equation 5**

The **Relative Absolute Error** (or RAE for short) provides an easier to evaluate performance measure as the number is a percentage of the mean absolute error relative to the mean of the observed duration values.

$$\mathrm{RAE} = \frac{\sum_{i=1}^{n}|y_i - x_i|}{\sum_{i=1}^{n}|x_i - \bar{x}|}$$ **Equation 6**

The **Root Mean Squared Error** (or RMSE) is one of the most commonly used error measures in statistics because of its advantage in ease of mathematical manipulation. It takes higher values than the Mean Absolute Error, being especially vulnerable when the data has a few outliers that are very far from the mean.

$$\mathrm{RMSE} = \sqrt{\frac{\sum_{i=1}^{n}(y_i - x_i)^2}{n}}$$ **Equation 7**

The **Root Relative Squared Error** (or RRSE), just like the Relative Absolute Error, is just an easier to evaluate expression of the Root Mean Squared Error with respect to the mean values of the observations.

$$\mathrm{RRSE} = \sqrt{\frac{\sum_{i=1}^{n}(y_i - x_i)^2}{\sum_{i=1}^{n}(x_i - \bar{x})^2}}$$ **Equation 8**

The **Correlation Coefficient**, presented in Equation 9, measures the statistical correlation between the predicted and actual values. It ranges from 0 for results that are not correlated to -1 or +1 for results that are perfectly correlated negative or positive respectively. For predictions of fabrication durations, such as those that this thesis is aiming to improve, the correlation should be positive and as high as possible, as opposed to the other error measures where the lower their value, the better the prediction. This measure differs from all the other by being scale independent on the number of instances or any other constant that the predictions may be multiplied by.

$$\text{Correlation Coefficient} = \frac{\sum_{i=1}^{n}\{(x_i - \bar{x})(y_i - \bar{y})\}}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \sum_{i=1}^{n}(y_i - \bar{y})^2}}$$ **Equation 9**

Validation Methods

In order to rely on the performance measures of learning schemes, one needs a systematic way to evaluate the reliability of a model's output. The data available to build a model might have significant features that do not appear very frequently, and before a model can be relied upon it needs to be able to deal as robustly as possible with the particularities of the data (Witten, Frank, & Hall, 2011). Another reason a systematic evaluation method is required is driven by the limited amount of available data itself and the quality of its structure. The main model validation methods for model development are:

a) **Using the entire data set** – this method does not involve using any means of actually validating a model using previously unseen information, focusing instead on the use all of the available data in order to create the best matching model using the features presented. Using this method may be appropriate only in the case when the data set presents with a very large number of records with well-defined attributes that are able to capture all the features of the data. The performance measures computed using the entire dataset for training are not typically reliable for predicting the performance of the classifier on future data, given the uncertainty that the future will be an exact replica of the past.

b) Training and Testing – represents splitting the data set in two parts, one used for creating a model, called a training test, and another for testing it with previously unseen information, called a testing set, essentially emulating the future use of a learning method. This represents the main true model validation technique that predicts how a classifier will perform given new data. The main shortfall of this method is that it relies on the assumption that both the training and the testing set are representative samples of the underlying features of the data, which may be difficult to determine when the supply of data is not very large, given that a typical training set uses only up to two thirds of the data. The dilemma with this validation method is that in order to find a good classifier we want to use as much of the data for training as possible, but that would leave a testing set that may not be sufficient to provide a reliable error estimate; this is the problem that cross-validation attempts to solve.

c) K-fold cross-validation – represents the repletion of the training and testing method for a k number of times in order to reduce the variability introduced by sampling training and testing subsets. The number of times the method is repeated represents the number of folds of the method. According to Witten, Frank, & Hall (2011), "*Extensive tests on numerous different datasets, with different learning techniques, have shown that 10 is about the right number of folds to get the best estimate of error, and there is also some theoretical evidence that backs this up*". This method is likely to yield the most reliable results but there is still some randomness and sampling involved. In order to assess the level of confidence in the reliability of the 10-fold cross-validation, the validation method should be employed several times and the deviation of the performance measures analyzed. A typical procedure that yields reliable classification performance estimates is to repeat the 10-fold cross-validation 10 times, therefore running the algorithm through the training and testing phases 100 times.

Comparing Different Data Mining Schemes

Once a few data mining schemes have been identified as having promising results, they need to be compared in order to determine which one should be carried forward towards implementation. Comparing between the different performance measures previously presented may sometimes be sufficient, but when the difference becomes smaller, it needs to be determined if it is within the estimation margins or a clear performance advantage. Using the cross-validation technique for error estimating multiple times improves the reliability in a performance measure, but a statistical tool is required to determine the confidence bounds and statistical significance of the difference. A common tool is the t-test and it is best to be applied on the pair of results from the cross-validation runs. Because of issues introduced by splitting the same data set multiple times into training and testing sets, a modified version of the statistical t-test that is not sensitive to the number of sampling repetitions is required. The solution suggested by Witten, Frank, & Hall (2011), presented in Equation 10, is to use a corrected paired t-statistic.

$$t = \frac{\bar{d}}{\sqrt{\left(\frac{1}{k} + \frac{n_2}{n_1}\right) \sigma_d^2}}$$

Equation 10

where
t = Corrected paired t-statistic
$\bar{d}$ = Mean of the difference between observations
k = Number of error samples
n_1 = Number of instances used for training
n_2 = Number of instances used for testing
σ_d^2 = The variance of the samples

CHAPTER 3. Process Simulation Modelling

This chapter presents a detailed, task-level modelling of the spool fabrication process, accompanied by a simulation implementation that includes a sample application of such a model for comparison between push and pull scenarios. In addition, efforts of integrating the simulation model with the fabrication planning and tracking data are presented. The intent is that the simulation model should be automatically triggered to run at discrete time intervals (i.e. once a day) and update the schedule and the delivery date estimate of spools every time new information is available. The proposed approach is summarized in Figure 3-1 below, which shows that the process simulation model, which handles scheduling under resource constraints, is to be nested within a dedicated schedule Information Management System (IMS) that handles the communication with a database.

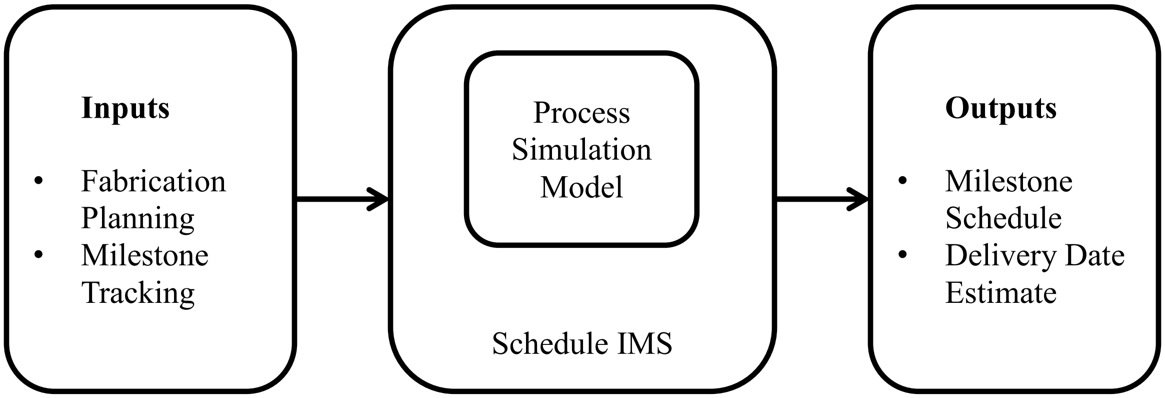


Figure 3-1 Process Simulation Modelling Methodology

The factors affecting the time required for spool fabrication are of a very diverse nature, resulting in a very uneven distribution of manufacturing durations which in turn leads to difficulties in deciding upon the right time to start the work (Mosayebi, Fayek, Yakemchick, & Watters, 2012). This problem lends itself well to the concept of discrete event simulation that can be used to improve predictability of fabrication duration requirements. This is particularly useful given the high product variability characterizing pipe spool fabrication, and therefore, it is expected that simulating the fabrication process may lead to a more accurate estimate of the delivery date of spools. Although the author has only visited one fabrication shop, efforts have been made to generalize the process and model it in such a way that its application can be considered universal for any facility employing a standardized station approach as suggested by lean manufacturing concepts.

The modelling work was performed using the General Purpose Simulation template within the Simphony.NET framework provided by the Hole School of Construction at the University of Alberta (AbouRizk & Mohamed, 2000). Since the platform allows for a graphical implementation of the process flow, efforts have been made to make use of the template's capabilities before using code to generate specialized behavior.

3.1 Fabrication Process Modelling and Simulation

In order for the spool fabrication manufacturing facility to meet the required deadlines, it needs to estimate how long a spool will spend within each station of the fabrication shop. This time is influenced by the spool design characteristics and work procedure requirements, as well as the available shop resources and the work method employed. It is expected that a significant difference in shop performance arises from the way and frequency that spools are released for fabrication.

The initial modelling approach consisted of an implementation of the fabrication processes outlined in Figure 1-1 presented in Section 1.1.2, with the main aim being to develop a model that can evolve into a tool used for deriving project durations at a highly detailed level using information that describes the individual requirements of each spool. Although in the implementation presented in the following pages the characteristics of each spool are assigned based on distributions selected using the author's heuristic estimates, it is believed that given a commercial project, the simulation approach presented can be translated and used with actual pipe spools parameters and requirements. The design characteristics of the spool that were considered to be related to the activity durations are the material type, pipe diameter, overall size, complexity, number of welds and number of parts.

Work procedure requirements refer to the type of welding and testing that is required for each spool. These requirements are usually project specific or module specific, and can also include information with regards to the paint, non-destructive examination or post-weld heat treatment of the spool. The modelling approach presented herein aims to consider these aspects but the author chose not to go into the details of each procedure, but instead sample durations from a distribution scaled with respect to the overall size of the spool expressed in Diameter Inches (DI). Just as was the case for the activity duration derivation approach, when given a real project, these requirements will be provided by the engineering team and will not need to be artificially created.

Resource availability refers to both human and machine/shop resources. A resource pool will be created for each type of worker, i.e. Cutter, Fitters, Welders, Wheel Loaders, as well as for Cranes available in a typical fabrication shop. Tools and other machines such as stands, grinders and welding machines will not be modeled as resources, but instead it will be assumed they will be assigned to each worker or station.

3.1.1 Spool Fabrication Process Description and Abstraction

The intent is to mimic the work required to be performed by a manufacturing facility including processing of spool information encapsulated in the client ISO drawings, preparation of materials, spool fitting, welding, and post-fabrication. The general approach to fabrication was considered to be that of processing a group of spools at a time as a shop package rather than all at once or one by one. In the authors' opinion this approach is the most common one and it allows for the balancing of resource loading within a manufacturing facility.

For the purpose of this analysis, the preparation of shop drawings has not been modelled as a process, but rather as a computation definition of the requirements for each spool. As such, the modelled entities were the pipe spools which are all created at the same time, when a project is awarded to a company. These entities are assigned properties that would individualize them just as pipe spools are unique in the field.

The first set of activities to be modelled for analysis are related to the preparation of materials for fabrication, which includes pipe cutting and pulling the required fittings from the warehouse and grouping them onto a cart specific for each package of spools. The activity durations are derived as a function of the number of pieces and pipes and fittings required.

Once all the materials have been prepared, they are transferred to the first stage of fabrication which is the Fitting of the pipes and accessories together according to specifications. The implementation presented herein does not differentiate between rolled or position welding and as such it regards each spool as a unit, and models the process of fitting and welding and entire spool as whole. In reality, however, complex spools that require a higher percentage of position fitting will take longer to complete than those that can be set-up for rolled fitting and welding only. However, the model does take some of this complexity into account when deriving activity durations by relating them to the number of fittings and pipe pieces required for each spool.

The activities performed within the welding station have been abstracted in a similar fashion to the ones required for fitting, with the exception that uncertainty has been introduced for the duration of some activities by means of sampling a distribution. This is meant to replicate the different welding procedures required by each client. However, the parameters of this distribution have been approximated based on best estimates and do not reflect a specific set of data of any project or company.

With regards to the post-fabrication activities that a manufacturing facility is required to perform before a spool is released, only the painting process has been considered in this model. Non-Destructive Examinations, Post Weld Heat Treatments and other such processes are project specific and it is understood that each fabrication shop approaches them in different ways. As such, they do not form part of this analysis. For the modelling of the painting activities, it was assumed that an entire shop package shall be processed at once.

Resource Modelling Assumptions

A single fabrication area or bay was considered in this exercise, and as such, resources that author considers typical for one such bay were modeled. These are the workers such as the pipe cutter, fitters, welders and wheel loaders and the main large equipment such as the overhead cranes, the welding rolling wheels and the paint shop. Two fitters and two welders are considered to be employed by the fabrication shop initially, supported by two wheel loaders and a cutter. It is assumed that the fabrication shop will have three overhead cranes and two rolling wheels, one for each welder.

A distinction needs to be made between each overhead crane based on capacity and their use. As such, the first one may be used by the cutter and fitters, the second by the fitters and wheel loaders and the third only operated by wheel loaders for the welding station and paint shop loading and unloading. Since the spool components are not as heavy as a completed spool, crane 1, used at the beginning of the process, may have a lower lifting capacity than the other two.

The reliability of some of the resources can also be incorporated in a simulation model, and as such breakdown events have been defined for cranes 2 and 3. This behavior can be switched on and off, allowing for the analysis of the impact that such events may have on the fabrication durations and efficiencies. The time between breakdowns has been modelled as an exponential distribution, with a mean of 3000 hours for Crane 2 and 2000 hours for Crane 3. When the breakdown occurs, the time to repair the crane was considered to be a constant of 24 hours for Crane 2 and 30 hours for Crane 3. These repair durations and breakdown intervals are selected purely for demonstration purposes bearing no reflection of any particular fabrication shop. The mean time between breakdowns in particular is believed to be grossly exaggerated; however, these values were selected to demonstrate the impact that such an event might have on the operation of a manufacturing facility.

Defining Spools

In order to achieve the spool definition and randomness required, a number of characteristics have been considered for modelling. These are the pipe diameter, material type, number of fittings and pieces of pipes required by each spool. Because the modelling work presented in this chapter has happened in parallel with the analysis of the data collection structures of the fabrication shop, assumptions regarding spool characteristics had to be made at this modelling stage. The type of distribution and their parameters are only an example and are to be replaced with actual spool parameters once this process simulation model is integrated with the database connection model which will provide this information as read from the drafting software database.

The example distributions for the spool characteristics mentioned are presented in Table 3-1 below. Triangular distributions are used to model each of these, with parameters selected to reflect what the author considers to be a typical project. When sampling values for the pipe diameter, number of fittings and pipes the number used was the integer closest to the sampled value.

Table 3-1 Spool Characteristics Sample Distributions

Characteristic	Type of Distribution
Pipe Diameter	Triangular (2,32,16)
Material Type	Triangular (0,20,10)
Number of Fittings	Triangular (0,6,4)
Number of Pipes	Triangular (1,4,2)

Three types of materials have been considered, Carbon Steel (CS), Stainless Steel (SS) and Galvanized Steel (GS). These are the most common materials that spool fabrication shops will come across and there are significant differences between them with regards to fabrication time. The distribution presented in Table 3-1 for Material Type has been split into three zones for each of the material type as follows:

- For a number from 0 to 10 (50% probability), the assigned material will be CS.
- For a number from 10 to 14 (20% probability), the assigned material will be SS.
- For a number from 14 to 20 (30% probability), the assigned material will be GS.

Once the spool characteristics have been defined, the total amount of weld diameter-inches is computed. Although it is recognized that each fitting and spool configuration is different, in order to abstract this complex situation it was assumed that each fitting and each pipe will require one weld. Therefore the sum of the number of pipes and fittings was multiplied by the pipe/fitting diameter in order to derive the spool DI.

After all the spool characteristics are assigned, the spools are filed in a database waiting for release for fabrication. A fabrication package consisting of five spools is released when requested. This batching method is intended to mimic the practice of issuing spools by package.

Model Activity Duration Assumptions

As previously mentioned, fabrication activity durations have been heuristically derived for this implementation to showcase the process modelling approach, with the intent that the derivations can be adjusted to represent a particular fabrication shop given sufficient data in the correct format. These assumptions are presented below for each of the fabrication processes. The presented breakdown of activities into tasks represents the result of an in-depth investigation of the fabrication process performed by the author.

Preparing Materials

When a shop package is released for fabrication, a pick ticket for the fittings is generated and passed on to the materials warehouse, while a cut sheet is released to the cutting station. The main activities undertaken by the warehouse personnel are to locate the required fittings and prepare a cart specific for each spool. It is assumed that each of these activities is limited to one server and the durations are related to the number of fittings required and their diameter. As such, it is considered that it takes 12 minutes to locate each fitting; since moving heavy and/or large diameter fittings requires more time; this is reflected by modelling the duration of the cart preparation activity as requiring 12 minutes for fittings with a diameter less than 24" and 20 minutes for larger ones.

For pipe cutting, a sequence of six tasks has been identified; these are: locate pipe; place pipe on table; measure, mark and position; preheat; cut; and unload pipe. All these require a pipe cutter, which is modelled as a resource and captured for each spool for the entire time the cutting related activities are undertaken. Some of these activities, such as the placing of pipe on the cutting table and its unloading might require a crane if the pipe has a large diameter (considered here to be over 24 inches) and as such those activities will capture Crane 1 but release it as soon as they are finished. The following durations have been assumed for the activities mentioned above:

- Locate Pipe: 20 min per pipe.
- Place on Table: if pipe diameter is less than 24", 6 minutes per pipe; otherwise, 12 minutes per pipe.
- Measure, Mark and Position: 6 minutes per pipe.
- Preheat: only required if pipe diameter is more than 5 inches; 1 minute per pipe per DI (i.e. 48 min for a spool with two 24" pipes).
- Cut: if pipe diameter is less than 5", 3 minutes per pipe; otherwise, 1 minute per pipe per DI plus 6 minutes.

- Unloading Pipe: if pipe diameter is less than 24", 3 minutes per pipe; otherwise, 12 minutes per pipe.

Pipe Fitting

Although all the pipes and fittings required for a spool are provided to the fitting station, they need to be moved again by the fitters and the pipe ends need to be grinded before the spool can be assembled and tack welded. Following the spool assembly, it needs to be checked, marked and moved to the welding area. The durations assumed for each task and the resources required are as follows:

- Move Pipe and Fittings: 10 min for each pipe and fitting; requires the first fitter and Crane 1.
- Grind Pipe Ends: 2 minutes per diameter for each pipe and fitting (i.e. 120 minutes for a 12" spool with 2 pipes and 3 fittings); requires Fitter 1.
- Set-up Spool: 20 minutes for each pipe and fitting; requires Fitter 1 for all spools and Fitter 1 and 2 for spools with more than 24" diameter.
- Tack-Weld: 10 min for each pipe and fitting; requires the first fitter.
- Check and Mark Spool: 6 min for each part; requires Fitter 2
- Move spools to welding area: 10 minutes plus 2 minutes per spool DI (i.e. 130 minutes for a 12" spool with 2 pipes and 3 fittings); requires Fitter 2 and Crane 1 if the pipe diameter is less than 24"; Fitter 2, Crane1 and Crane 2 if the pipe is larger

As it can be observed, some activities only require Fitter 1 whereas others only require Fitter 2; however, these activities have similar duration so the resource loading is expected to be balanced to a reasonable degree.

Rolled Welding

The general approach with respect to the welding station is that a wheel loader will move the fitted spool to the welding area and secure it to a rolling wheel; the operation was assumed to take 10 minutes per spool plus 2 minutes for each DI (i.e. 290 minutes for a 140 DI Spool). The wheel loader will need only Crane 2 for spools with less than 100 DI; both cranes are required for larger spools. Following loading, the welder prepares for welding (which is assumed to be an activity with a constant duration of 12 minutes) and performs the welding operation, before the loader and crane are required again to unload the wheel.

The welding operation can be modeled very specifically depending on the procedure required by each project but since the object of this approach was to provide a project-independent model, no such particulars have been implemented. However, the present abstraction samples a triangular distribution and scales it according to the total amount of welding required for each spool to produce a sample duration for the welding activity. The parameters of the distribution are (DI/20, DI/7, DI/10) – this can be translated, for a spool with 140 DI for example, into (7, 20, 14) which represent the low, high and mod parameters that can be sampled which are equal to an activity duration, expressed in hours.

As mentioned before, no differentiation is made between rolled and position welding. In reality, the time required for a position weld is somewhere between 2 and 3 times longer than that of a similar rolled weld. It is considered that the scaled duration distribution employed is enough to capture the variation in activity durations that spools with different geometrical arrangement have.

When the welding is complete, the loader will unload the wheel and transfer the spool to the post-fabrication area. This activity was assumed to take 10 minutes per spool plus 1 minute for each DI (i.e. 150 minutes for a 140 DI Spool). The wheel loader will need only Crane 3 for spools with less than 100 DI; both cranes are required for larger spools.

Post-Fabrication

As mentioned previously, of the post-fabrication activities typically performed by a manufacturing facility, only the painting process has been considered within this implementation. This is due to the different approaches that manufacturing facilities have, some choosing to sub-contract them, others performing them in-house if required. However, they can be analyzed and simulated independently of the main spool fabrication.

Once a spool is completed, it is moved to the paint shop by the wheel loader; the activity has been modeled as having the same duration and requiring the same resources as the activity of unloading the rolling wheel following welding. The painting operation duration is proportional to the total amount of welds on each spool but the ratio is dependent on the material type. As such, a CS Spool requires 3 min per DI, a SS Spool requires 2 min per DI and a GS Spool requires 5 minutes per DI.

3.1.2 Simphony.net Implementation

Simphony.NET supports the implementation of the process previously described using primarily the General Template elements available to the user. Modelling elements designed for defining, capturing and releasing resources have been used to model the workers and major shop components, as well as their breakdown. The resources described in Section 3.1.1 have been modeled as shown in Figure 3-2 below. Since it was desired to distinguish between Fitter 1 and Fitter 2 within the model, the two workers have been modeled separately but are waiting within the same queuing file when idle. The same approach has been employed for the modelling of the cranes. Since the breakdown of cranes 2 and 3 is also modeled, a Preemption File has been added in order to allow for this later.

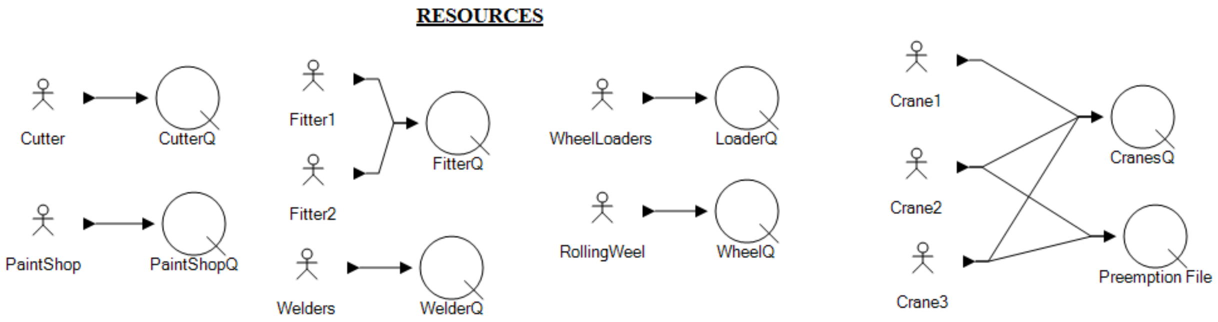


Figure 3-2 Modelling of Resources

Composite elements have been employed to represent each work station described in Section 3.1.1 and keep the model easily customizable and traceable. This high level abstraction of the fabrication process has been implemented as shown in Figure 3-3. The entities are created within the Define Spools element and a Destroy element has been used to simulate the shipping of the spools. The spool properties have been set using Set Attribute elements whereas work activities have been modeled using Task elements that encapsulate the desired behavior of each task. Statistics collected within the composite elements with regards to the fabrication duration are easy to observe using the Statistic element presented at the high level of the model.

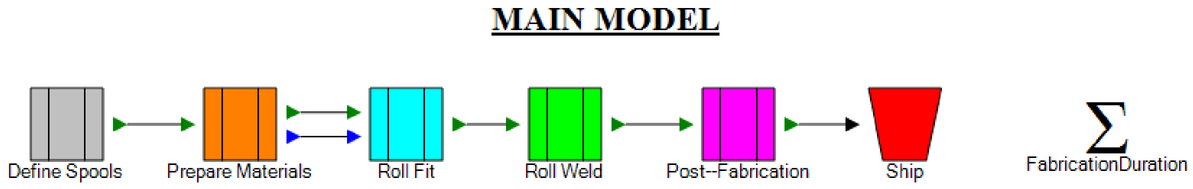


Figure 3-3 High Level Representation of Spool Fabrication

Example Model Application

Provided that the model presented above can be calibrated and validated using real data to replace the heuristic duration assumptions, the model can serve to evaluate different what-if scenarios regarding resource availability, equipment breakdowns or shop layout changes. The following discussion presents the description of such example model implementations, focused around the objective of improving predictability of fabrication duration requirements under different working conditions. This has been achieved by creating two scenarios using the model in order to simulate a push and a pull behavior, as well as the impact of crane breakdowns.

Manufacturing facilities are mostly operating using a push scenario, where fabrication packages are released for fabrication as soon as the fitting of the last spool is complete, in order to avoid long idle times for the cutter and fitters. This operating mode will, however create clustering within the facility and give rise to queuing situations for some of the spools that often involve utilizing the limited floor space for longer periods of time than necessary. In addition, double handling of these spools is also required, which is wasteful and therefore seeks to be eliminated when implementing lean fabrication concepts. A way to manage this situation is to ensure resources are balanced within the facility and activities taking place towards the end of the process do not create significant bottle necks. However, some resources cannot be balanced as easily, an example of this being the number of welding rolling machines, which are a function of the available floor space within a shop, sometimes regardless of the dimensions of a spool.

In order to avoid these issues, a pull behavior can be employed, where each package is released for fabrication when the manufacturing facility has nearly finished the previous one. More specifically this is achieved when the welding of the last spool within a package is complete. An operation of this type will ensure the critical space limited resources are available before a new package is released. However, this involves operating the cutting and fitting stations at less than maximum capacity. One way to avoid this issue may be to group similar spools together when releasing the shop package.

The scenarios described herein have been implemented using a valve method whereby all the spools are stored and released in packages of five when requested from other stages in the model. For the push scenario, the valve is opened when the fitting of package has finished, whereas for the pull scenario this is further delayed until all welding is complete for a given assembly of spools. Special emphasis has been placed on developing a comprehensive trace output of the model that allows for the observation of each spool's behavior, potentially useful in scheduling their release for manufacturing.

One of the limitations of the present implementation is that grouping spools together in the order they have been created introduces high variability between the packages and between spools within the package is highly likely.

Example Application – Results Discussion

Each scenario was simulated 100 times in order provide accurate results that are not influenced by random sampling of distributions. The most relevant results to be compared are the waiting times for each of the resource queues. A summary of the average waiting times for resources compiled for all scenarios is presented in Table 3-2.

Table 3-2 Comparison of Waiting Times for Different Simulation Scenarios (in hours)

Resource type	Push Scenario	Pull Scenario	Pull Scenario With Breakdowns
Cranes	1.30	0.62	0.68
Wheel	65.58	1.56	1.79
Paint Shop	15.11	3.38	3.41
Cutter	5.66	4.54	4.56
Fitters	4.97	3.38	3.49
Loaders	0.38	0.14	0.16
Welders	0.00	0.00	0.00

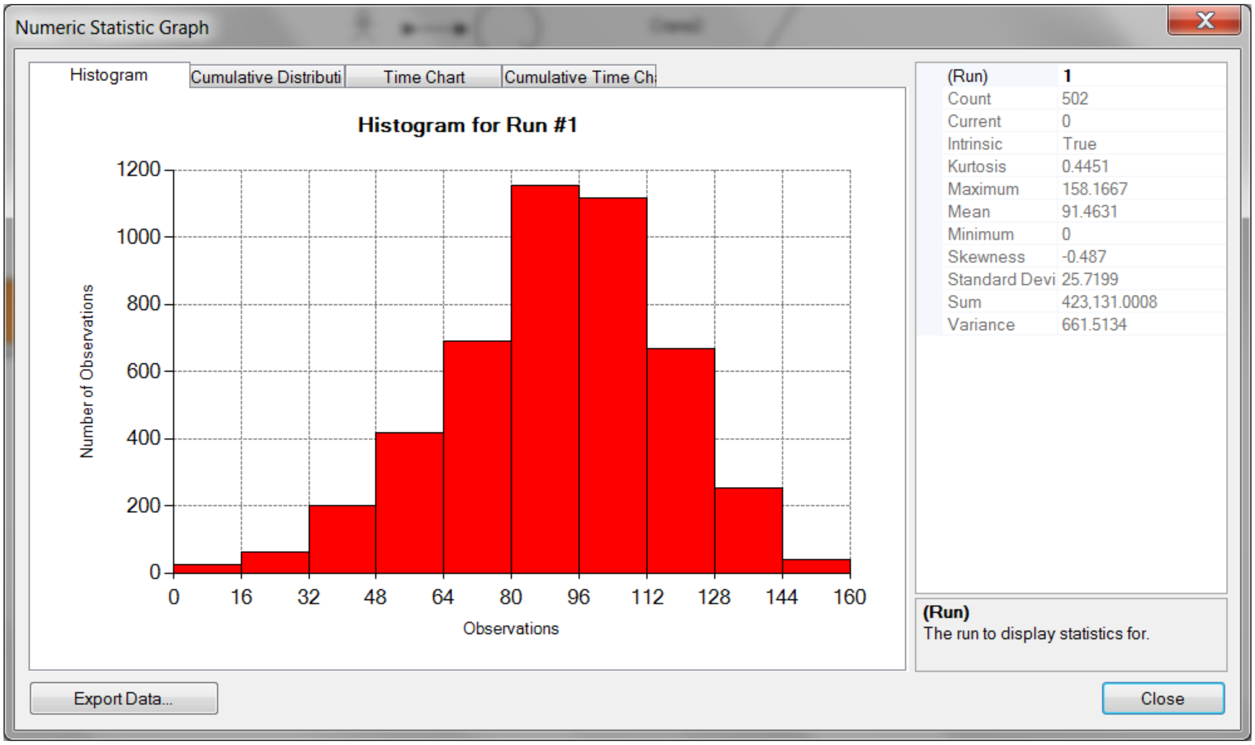
The waiting times for the Pull scenario are smaller than those for the Push scenario for all of the resources modelled. A significant difference is observed in the case of the rolling wheel, where a reduction in waiting time by a factor of 40 is noticed. The waiting times for other resources are significantly different too, such as a factor of 2 differences in the average waiting time for the cranes.

Introducing the crane breakdown events for the Pull scenario shows slight increases in the waiting times, in the region of 5-10%. However, as previously discussed this impact can be observed in more detail only if the breakdown behavior is modeled using historical data.

The waiting time for the welders queue is zero for all scenarios because no welder is captured before a spool is on a wheel, therefore no spools are waiting specifically for welders. The wheel waiting times can be an indication of the welder waiting time.

An important aspect to be analyzed is the distribution of fabrication times and the differences that arise between scenarios. This is of particular importance when trying to assess the implementation of lean manufacturing system that employ a pull rather than a push system.

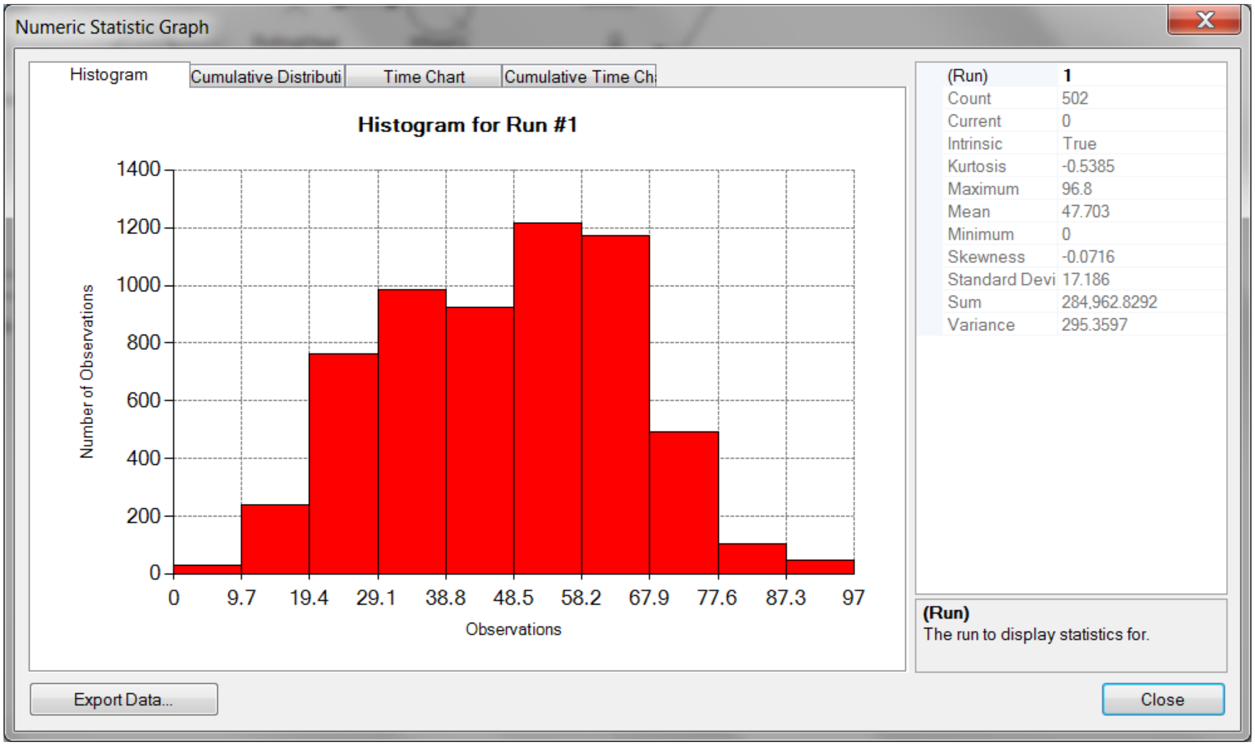
The figures below are histogram representations of the fabrication duration as reported by the "FabricationDuration" statistic element. This statistic represents the time it took for the completion of one spool from the time it was released for fabrication to the time it was transferred out of the welding area. Figure 3-4 depicts the histogram for the Push Scenario, whereas Figure 3-5 presents the pull scenario.



(Run)	1
Count	502
Current	0
Intrinsic	True
Kurtosis	0.4451
Maximum	158.1667
Mean	91.4631
Minimum	0
Skewness	-0.487
Standard Devi	25.7199
Sum	423,131.0008
Variance	661.5134

(Run)
The run to display statistics for.

Figure 3-4 Fabrication Duration Distribution for the Push Scenario



(Run)	1
Count	502
Current	0
Intrinsic	True
Kurtosis	-0.5385
Maximum	96.8
Mean	47.703
Minimum	0
Skewness	-0.0716
Standard Devi	17.186
Sum	284,962.8292
Variance	295.3597

(Run)
The run to display statistics for.

Figure 3-5 Fabrication Duration Distribution for the Pull Scenario

As it can be observed, the mean fabrication duration for the push scenario is 91 hours per spool, with a standard deviation of approx. 26 hours. Of significant importance to the predictability of the fabrication duration requirements is the high variance of this distribution, as well as the very large maximum, of 158 hours.

The mean duration for the pull scenario, on the other hand, is approx. 48 hours, nearly half that required when employing a push operation. Also, the maximum duration of the fabrication is reduced to 97 hours. Although the standard deviation of this distribution is 17 hours, smaller than that of the push scenario, the histogram shows a more even duration distribution.

Some of the variability can be attributed to the randomness of the spools, as well as their different characteristics, but an important conclusion can still be drawn due to the high variation between the two scenarios, which is that releasing a spool for fabrication too early will increase its waiting time for resources and therefore its total completion time, as well as reduce the chance that it will be completed in the estimated time.

3.2 Database Connection Model

This section aims to describe how a simulation model that is designed to derive a fabrication production schedule can be linked with a fabrication planning and tracking database. To allow for the previously investigated process simulation model to be implemented, first the fabrication tracking and scheduling user interface presented in Section 4.1 and currently developed in Microsoft Excel needs to be connected to the simulation engine. Since the Excel interface is already a very complex file, it was decided to convert the primary production scheduling and tracking data sheet to a flat table that can be opened using Microsoft Access. This conversion will also make the connection to the Simphony program easier to achieve while at the same time providing a common place for the simulation engine to deliver its outputs.

3.2.1 Connecting to the Database

In order to set-up a connection between the simulation model and a database, a "Database" element, found in Simphony.General → Miscellaneous needs to be set-up using the particular details of the machine where the Simulation is being run.

The first step is to specify the connection provider. When working with a Microsoft Access Database, this provider is called "Microsoft Office 12.0 Access Database Engine OLE DB Provider". For the provider to be available in the Simphony element, first the Microsoft Access Database Engine (Microsoft, 2015) needs to be installed on the local machine.

Next, a connection string for the Data Source needs to be specified in the Connection tab. This is a local address of the database on the machine where the simulation is being run. Lastly, the connection needs to be tested to ensure the simulation model will run without issues.

3.2.2 Model Description

The model consists of 3 execute elements, a Trace element and a Destroy element, as pictured below in Figure 3-6. The functions of the execute elements will be described below with the visual basic formatted code presented in Appendix 1. The Trace element is used only when the Update element is bypassed for debugging or code altering. The Destroy element erases the entities from the memory of the simulation engine.

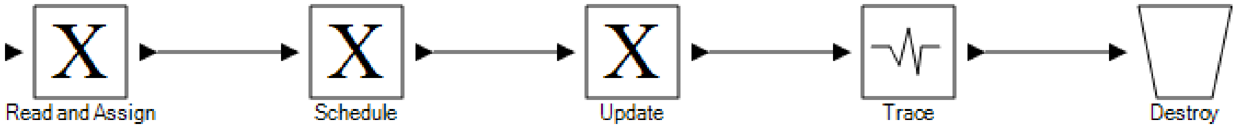


Figure 3-6 Initial Scheduling Engine

The structure of the simulation has been determined based on the consideration that reading from and writing to a complex database using an external application is a time-consuming process and therefore it is desired to limit the application to only open the database twice: once to read from it and another to write to it.

"Read and Assign" Element

The first execute element, is named Read and Assign, and it handles the task of reading from the database and creating each spool in Simphony as an entity. It will assign attributes to each entity corresponding to each spool that is found in the database.

Being the first element in the series and tasked with creating the entities, the entire coding work will be in the Initialize expression of the element. To keep this description fluent, the code is provided in Appendix 1a).

Firstly, the database connection element is called upon to open a reading link between the database and the simulation engine. An SQL string is used to list all the parameters that need to be retrieved and read. Particular attention needs to be exercised when developing or modifying this string as it will need to contain the required column names in the database in the correct order and with the correct spelling.

Secondly, this element's code loops the read command for all the records found in a database; once each record is read, the information is stored in the local entity properties as either numbers of type Float or character strings of type String. When modifying this part of the code to add or subtract attributes to be recorded, it is important to keep in mind the sequence in which the attributes are listed in the SQL command. For better visualization, numbers to keep count of the attribute order have been added on the line immediately bellow the SQL query.

Once the code has looped through all the records in the databased and created entities with the information stored locally, the database connection is closed and the simulation engine is instructed to proceed to the next element in the simulation.

“Schedule” Element

The main function of the Schedule element is to compute durations between fabrication milestones based on the specific routing and requirements of each pipe spool. The method employed is incremental and additive, meaning the duration is always in reference to the completion date. These durations are working days and will need to be processes over a working day calendar in order to create a schedule that shows milestone dates for each spool. The most convenient application for this conversion is Microsoft Excel, given that the software already has ready to use functions for adding and subtracting dates over a custom working day calendar.

The results of the calculation of the Schedule element are stored internally as attributes of the simulation entity, which represents a spool. This element is executed each time an entity passes through it and therefore does not employ any looping technique in the code. For reader reference, the code for this element is provided in Appendix 1b).

The detailed simulation model of the fabrication process complete with resource requirements presented in Section 3.1 shall replace the coded Schedule element once it has been validated.

“Update” Element

The function of this Execute element is to put together all the computed information into an Update SQL query, which will then be executed in order to write the information into the database. Just as was the case with the database reading function performed by the “Read and Assign” element, particular attention needs to be exercised when building the update SQL string in order to reflect the correct column names and their order.

This Execute element works by using a string builder to add each column name followed by a value stored in the local memory of the simulation entity. Once the string for the update query is completed, the database connection is called and employed to execute the update. For reader reference, the code for this element is provided in Appendix 01c).

3.3 Limitations and Remarks

The simulation components presented in the two previous sections of this chapter, namely the process simulation model and the database connection model, could not be completed to function as intended due to significant functionality limitations, addressed in the following paragraphs.

Discrete Event Simulation is a tool very well suited to simulating industrial manufacturing activities such as pipe spool fabrication. Depending on the modelling approach, various aspects of these processes can be investigated. The model presented in Section 3.1 of this chapter aims to aid project planners in understanding how different approaches to executing a project can impact its overall duration and timely delivery. However, because validating the model requires activity duration information at a level of detail much higher than that present in existing fabrication tracking structures at the manufacturing facility, the description of potential applications of this model are presented solely for demonstration purposes.

Even though the process simulation model may be converted to reflect the spool variability of any particular project by updating the Define Spools composite element to accept parameters from the Read and Assign element in the database connection model. However, the two models have not been integrated as originally intended. This development step has been omitted because of the lack of a validation procedure for the process simulation model as a result of the difference in data structure between available and required information.

While improvements in the accuracy of predicting fabrication durations and predicted completion dates are expected to be achieved using the fabrication process modelling approach presented, obtaining the data required to calibrate and validate an implementation with this level of data is extremely challenging and time-consuming. Some of the data required has to be collected from multiple projects and by various departments, which adds to the complexity of the data collection efforts.

Although it is believed that the detailed trace output of the process simulation model can be used by experienced operations managers to derive yet more significant conclusions than those presented, the simulation results cannot be relied upon until the process has been sufficiently validated. Such findings may include, for example, deriving a relationship between spool complexity and fabrication duration, as well as allowing for experimentation with different resource availabilities and equipment breakdowns.

The implementation of the database connection model presented in Section 3.2 is mainly a showcase of the potential integration of an existing fabrication scheduling and tracking system with a simulation engine derived schedule. It is limited in functionality mainly because it requires a clean set of data with a fixed structure and data types, as well as a consistent approach to dealing with missing and incorrect information at the fabrication tracking stage. As such, the current implementation contains a link to a database containing only an example dataset which has been cleaned to present the consistency required by the model.

The development efforts presented in this chapter, although not fruitful, led the author to the conclusion that different duration prediction and schedule derivation avenues need to be explored, focused on the structure and composition of the existing data collection practices, which are presented in the following chapter.

CHAPTER 4. Data Collection and Exploration

This chapter presents the description of a system developed to integrate the fabrication shop's data collection systems with heuristic scheduling rules provided by the manufacturing facility professionals working in an Alberta fabrication shop. This system, developed by the author, was required in order to achieve a platform that can be used to plan and control production sequencing while at the same time collecting data for analysis. This tool has been operational in the fabrication shop in the form presented for various tasks including spool package issuing decision making, fabrication prioritization according to the schedule and spool status reporting. The scope, requirements, mode of operation and reporting functions of the system were developed working very closely with manufacturing facility planning and execution control professionals. The resulting integrated tracking and scheduling data that has been captured during approximately one year of operations has been made available for analysis. A data cleaning and validation exercise that was performed prior to the investigation of the potential schedule improvement methods is also presented in this chapter

4.1 Integrating Fabrication Tracking and Scheduling

In order to address some of the above issues, the manufacturing facility needs to have a reliable and realistic schedule that incorporates and disseminates all available information in order to estimate the fabrication duration. In addition, integrating the schedule with the tracking system can provide the base of an evaluation platform that could be used to analyze the impact of process changes or workforce availability to the overall project delivery performance.

A multi-level spreadsheet-based system has been developed to integrate, enhance and facilitate the access to the above-mentioned information which will be further referred to as the "Scheduling and Tracking System". The number of formulas in each of the three spreadsheets is very high, resulting in a relatively slow updating time; in addition, the updating procedure is complex and any missed step will lead to errors of scheduling. For these reasons, the entire updating procedure of the Scheduling and Tracking System has been automated using Visual Basic code. A Data Flow Diagram of the system is presented in Figure 4-1 with a description of the functions of each of its parts in the following paragraphs. Complementing the summarized description of the system presented in this section is a more detailed design and usage manual developed for the industrial partner presented in Appendix 2.

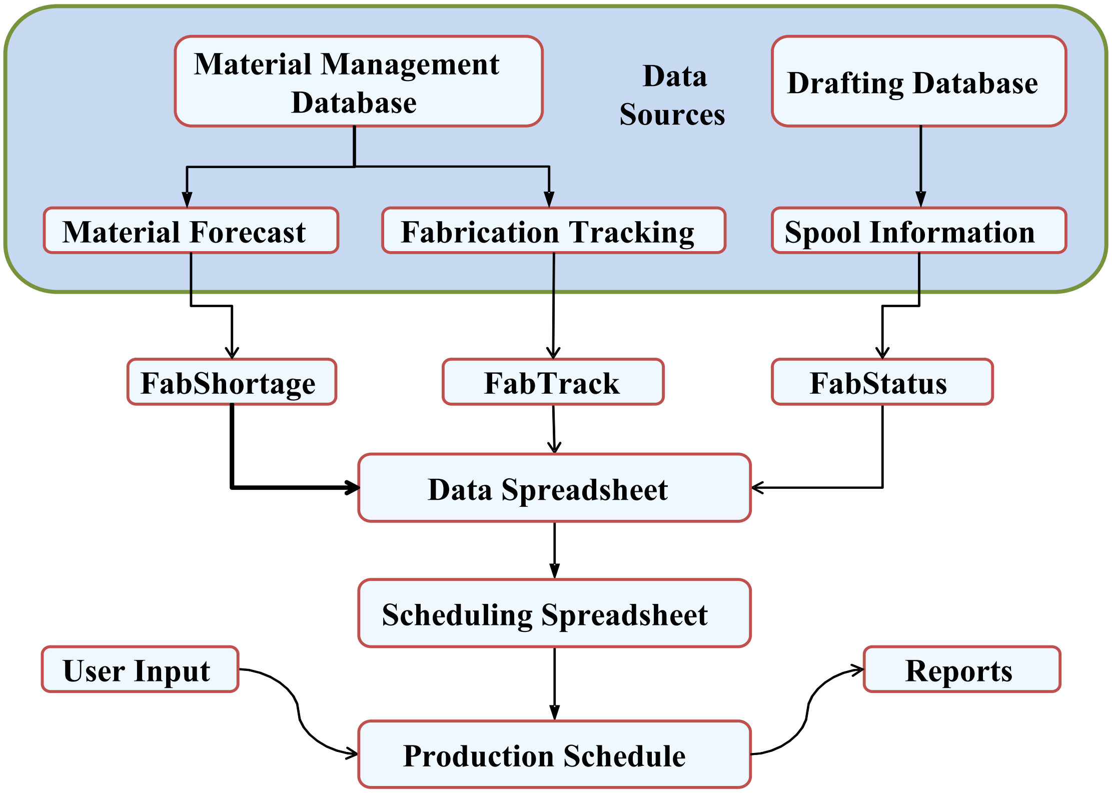


Figure 4-1 Scheduling and Tracking System Data Flow Diagram

The information required was scattered across three different data sources managed by 2 databases, as presented in the "Data Sources" part of Figure 4-1. Both the drafting and the material management software systems have proprietary, custom-built databases with tables that contain information regarding spool characteristics, milestone tracking, material management and much more. For the fabrication analysis, the drafting database provides spool characteristics and some milestone tracking information in the form of a "Spool Information" table. The material management database is used to generate a "Material Forecast" containing material availability information and a "Fabrication Tracking" table that stores milestone tracking information additional to that found in the drafting database.

The first level of functionality is managed in a "Data Spreadsheet", where the information exported from the databases in the tables mentioned above is cleaned and concatenated. Three files are created for each of the several projects that are to be included in the Production Schedule in an automated stage of extracting information from the previously mentioned data sources, called “FabShortage, “FabTrack” and “FabStatus”. A unique identifier of the spool is created using the “Control Number” and last 3 digits of a project number and represents a primary key for each spool. All redundant and repeated fields are removed at this stage and the material availability, number of items and highest item diameter of each spool is determined by consolidating information available at a component level. These files are combined in a single table containing all pertinent information with a line associated with each pipe spool.

The compiled data is passed on to a “Scheduling Spreadsheet” where a fabrication schedule is calculated based on heuristically determined durations. The derivation of fabrication durations has been performed at the planning and drafting stages, and the results are included in the Spool Information “FabStatus” table. Two types of schedules are created using finish-to-start relationships between activities and a 5 day working week calendar with custom holidays. The first, called the “As Planned” schedule, is a backward pass calculation starting at the required completion date providing a required start date each spool. The second, called the “As Issued” Schedule, is a forward pass calculation starting at the package issue date and providing an expected completion date based on the tracked progress of achieved milestones. Actual fabrication process durations are also calculated in this spreadsheet using the fabrication tracking information.

The resulting schedule is transferred to the user accessible spreadsheet called the “Production Schedule” which includes the functionality of generating production reports and allowing for quick fabrication schedule referencing and evaluation at an individual spool level. This spreadsheet employs conditional cell formatting to highlight whether the progress of a certain spool is on track or has been delayed based on a comparison of the actual fabrication progress and the scheduled date, for both the “As Planned” and the “As Issued” Schedule. In addition, various standardized filter options and reports have been created in order to facilitate production planning, monitoring and progress reporting. These functions have been developed with the help of manufacturing facility planning and production management team and tailored specifically for the requirements of each function. A sample of the resulting “As Planned” schedule is presented in the following subsection, together with a discussion of the selection of the information that will be used in the data analysis.

4.1.1 Description of Collected Data

In order to perform the required investigation of fabrication duration and delivery dates estimate improvements, data regarding fabrication progress milestones that have been tracked for a year in an Alberta fabrication shop has been made available for analysis. It is expected that by looking at past fabrication durations, useful relationships can be derived to estimate the scheduling of future projects. A snapshot of the above mentioned "As Planned" sheet of the Production Schedule was provided by the partner company for data analysis in order to allow for the identification of possible fabrication duration patterns and enhance the process of fabrication scheduling. Most of the processes and milestones presented are considered to be self-explanatory for a reader with a background in industrial fabrication. In the interest of completeness; however, the terms which the author believes are less trivial or critical for understanding of the work presented in this thesis are included in the Glossary of Terms.

The "As Planned" sheet of the "Production Schedule" has 68 fields of data that has been combined and processed as previously described. The sheet's headers are presented in Table 4-1 organized by category and accompanied by a short description. The scheduling headers for the 7 fabrication processes are repeated, but are shown in full in order to present the complete composition of the data scheduling and tracking system. The scheduled dates are computed from the RAS Date, whereas the actual dates are brought in from the tracking databases. A few example records with data from the Spool Information, Fabrication Details and Issue Scheduling categories are presented overleaf in Table 4-2, Table 4-3 and Table 4-4 respectively. The Fit Schedule and the transfer report dates (MRR and MTR) for the example records are presented in Table 4-5.

Table 4-1 Production Schedule – As Planned Sheet Headers

Category	Schedule Heading	Description
Spool Information	Control Number	Unique identifier of the spool
	Project	The project that the spool is part of
	FIWP	The Field Installation Work Package that the spool is part of
	Priority	The project schedule assigned priority order of the FIWP
	Diameter Inches	The sum of the Diameter Inches to be welded
	Maximum Size	The diameter of the biggest item that is part of the spool
	Material Grade	The code for the material grade or type for the spool
	Weight	The total weight of the spool, computed at the drafting stage
	Surface Area	The surface area that a spool occupies
	N0/ of items	Total number of items (pieces of pipe and fittings) of a spool
Fabrication Details	Status	The fabrication status of a spool (i.e. On Hold or In Fab)
	Bay #	Number of the shop bay where the spool is fabricated
	ECN Column	Engineering Change Notice (contains Subcontractor Info)
	Hold Details	Any details regarding a client hold
	Drawing Check Date	The date the spool drawing has been checked
Issue	Material Available?	A Yes/No attribute derived using Material Management Data
	RAS Date	Required At Site Date - Spool completion deadline
	Expected Ship Date	Date computed after a spool fitting has been completed
	MIR Number	Material Issue Report Number
	Planned MIR Date	Required package issue date of the spool, based on RAS Date
	Actual MIR Date	Actual package Issue Date of the Spool
	Planned Matl Issue Date	Date when material is required to be prepared by
	Actual Matl Issue Date	Actual date when material preparation was complete
	Location	The spool fabrication location (either a Bay# or Sub-Contractor)
Fit Schedule	Scheduled Duration	The scheduling and tracking headers for the Fitting process
	Scheduled Start Date	
	Actual Start Date	
	Scheduled Finish Date	
	Actual Finish Date	
	Actual Duration	
Weld Schedule	Scheduled Duration	The scheduling and tracking headers for the Welding process
	Scheduled Start Date	
	Actual Start Date	
	Scheduled Finish Date	
	Actual Finish Date	
	Actual Duration	

Category	Schedule Heading	Description
Visual Inspection (QC) Schedule	Scheduled Duration	The scheduling and tracking headers for the Visual Inspection process
	Scheduled Start Date	
	Actual Start Date	
	Scheduled Finish Date	
	Actual Finish Date	
	Actual Duration	
PWHT Schedule	Scheduled Duration	The scheduling and tracking headers for the Post-Weld Heat Treatment process
	Scheduled Start Date	
	Actual Start Date	
	Scheduled Finish Date	
	Actual Finish Date	
	Actual Duration	
NDE Schedule	Scheduled Duration	The scheduling and tracking headers for the Non-Destructive Examination process
	Scheduled Start Date	
	Actual Start Date	
	Scheduled Finish Date	
	Actual Finish Date	
	Actual Duration	
MRR	MRR Date	Material Receiving Report - represents a transfer of the spool between business units
Hydro-Testing Schedule	Scheduled Duration	The scheduling and tracking headers for the Hydro-Testing process
	Scheduled Start Date	
	Actual Start Date	
	Scheduled Finish Date	
	Actual Finish Date	
	Actual Duration	
Paint Schedule	Scheduled Duration	The scheduling and tracking headers for the Painting process
	Scheduled Start Date	
	Actual Start Date	
	Scheduled Finish Date	
	Actual Finish Date	
	Actual Duration	
MTR	MTR Date	Material Transfer Report - represents a transfer of the spool to the client and is considered to be the completion date of the spool.

Table 4-2 Spool Information Example Records

Control Number	Project	FIWP	Priority	Diameter Inches	Max Size	Material Grade	Weight	Surface Area	No. of Items
130 - 010011	3215130	MD-02-51	06	37	4	G	915.88	75.74	8
140 - 030166	3215140	MD-02-51	09	14	4	A	92.72	5.93	8
290 - 090078	3215290	MD-02-51	36	72	36	G	3354.34	79.75	3

Table 4-3 Fabrication Details Example Records

Status	Bay #	ECN Column	Hold Details	Drawing Check Date
F	Fabrication Bay # 2	/	/	3/25/2014
F	Fabrication Bay # 3	Company X	/	9/25/2014
F	/	/	/	7/15/2014

Table 4-4 Issuing Schedule Example Records

Material Available?	RAS Date	Expected Ship Date	MIR Number	Planned MIR Date	Actual MIR Date	Planned Material Issue Date	Actual Material Issue Date	Location
Y	7/4/2014	6/10/2014	MIR-130036	5/20/2014	5/9/2014	5/27/2014	5/12/2014	Bay # 2
Y	10/1/2014	1/21/2015	MIR-140143	8/21/2014	10/31/2014	8/28/2014	12/20/2014	Bay # 3
Y	9/1/2014	10/17/2014	MIR-290182	6/27/2014	8/26/2014	7/8/2014	9/18/2014	/

Table 4-5 Fit Schedule and Transfer Dates Example Records

Scheduled Duration	Scheduled Start Date	Actual Start Date	Scheduled Finish Date	Actual Finish Date	Actual Duration	MRR Date	MTR Date
7	6/3/2014	5/14/2014	6/11/2014	5/20/2014	4	5/23/2014	5/29/2014
5	9/5/2014	/	9/11/2014	12/31/2014	/	1/21/2015	1/21/2015
6	7/15/2014	9/22/2014	7/22/2014	9/22/2014	1	1/14/2015	11/28/2014

Out of the available data fields previously presented only some were selected for analysis from each of the categories mentioned. The main reason for omitting certain fields was the lack of data or lack of consistency in the data collection. It is of utmost importance to identify unreliable or inconsistent data and carefully work around it, avoiding as much as possible including the affected fields in automated analysis or reporting functions. Identifying such data fields required careful examination of the data and close collaboration with the fabrication personnel in order to understand what information was recorded in each database field and whether the data collection was consistent over time between different projects. This process begun during the development of the integrated fabrication scheduling and tracking system and provided the author with a solid understanding of the shortcomings of collected data. In order to avoid analyzing fields with incomplete or incorrect information, once certain columns have been removed based on heuristic methods, a comprehensive data cleaning and validation exercise was performed as described in the following section.

4.2 Data Cleaning and Validation

In order to make the most use of the information available, a data cleaning exercise was deemed necessary. The main issues surrounding the available information regard the completeness of the records, in particular with regards to the tracked fabrication milestone dates. Other criteria to be considered when selecting the data for analysis include:

- Client holds during the fabrication process
- Missing records of fabrication milestones
- Incomplete fabrication progress at the time of the data collection
- Removing outliers for increased data consistency.

The process to be followed is described in the following subsections begins with the presentation of the descriptive statistics of the total fabrication duration, continues with an outlier detection and removal exercise and concludes with a logical validation of the data based on the comparison of recorded fabrication milestones.

4.2.1 Descriptive Statistics

The initial fabrication duration data is characterized by the descriptive statistics presented in Table 4-6. The data these statistics are based upon is the total fabrication duration and it reflects all the records available where this information is valid.

Table 4-6 Descriptive Statics for Fabrication Duration

Descriptive Statistics	
Mean	42.6
Standard Error	0.25
Median	40
Mode	30
Standard Deviation	22.3
Sample Variance	498.0
Kurtosis	1.67
Skewness	0.98
Range	201
Minimum	1
Maximum	202
Sum	336597
Count	7905
Confidence Level (95.0%)	0.492

An initial graphical representation of the data can be a useful tool to evaluate it. Using the Normal Inverse function, expected duration values have been derived based on the Cumulative Distribution Function of a normal distribution with the same mean and standard deviation. This graph, presented in Figure 4-2, can be used to conclude that the data does not closely follow a normal behavior. This is due to the lack of a “tail” to the left of the duration spectrum and a concentration of the values towards to the mid-low values (i.e. 15 to 45 days).

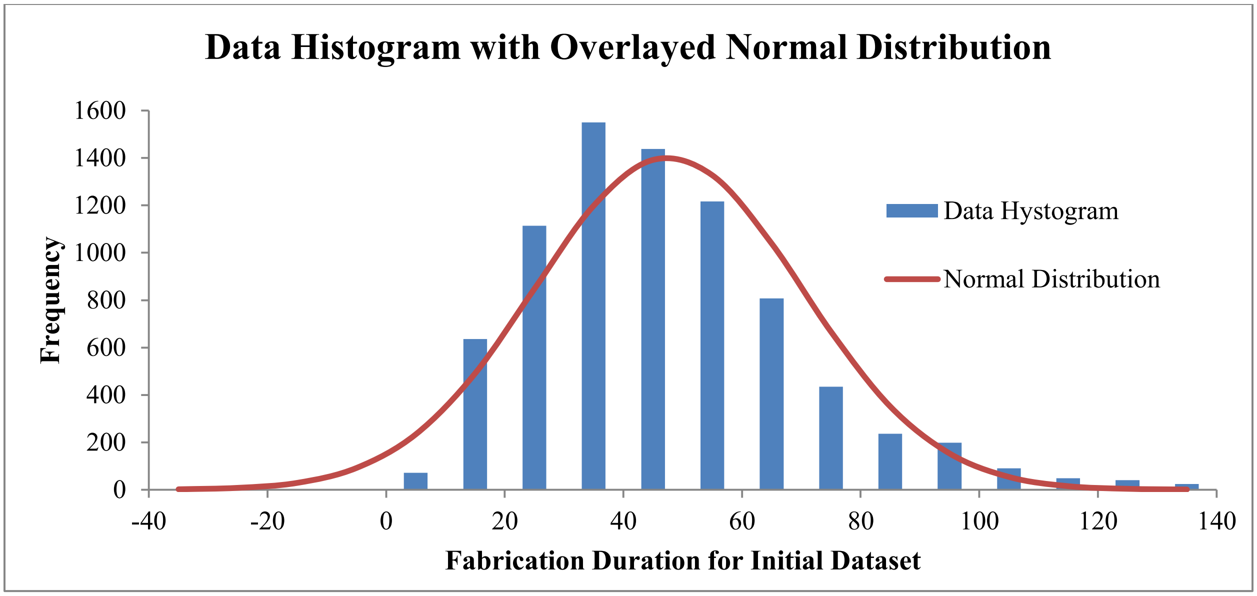


Figure 4-2 Fabrication Duration Data

In order to further assess how closely the data follows a normal distribution, a normality test was performed based on the Fabrication Duration Data. The test, presented in Figure 4-3 below, shows that the original data matches the expected values only for the central part of the graph, whereas the values towards the ends of the data diverge from the normality plot. As mentioned in (Brown, 2015), when the ratio of the skewness to Standard Error of Skewness (SES) expressed as presented in Equation 11 is higher than 2, then the sample is positively skewed. This is the case for the present data set.

$$\text{SES}=\sqrt{\frac{6n(n-1)}{(n-2)(n+1)(n+3)}} \qquad \textbf{Equation 11}$$

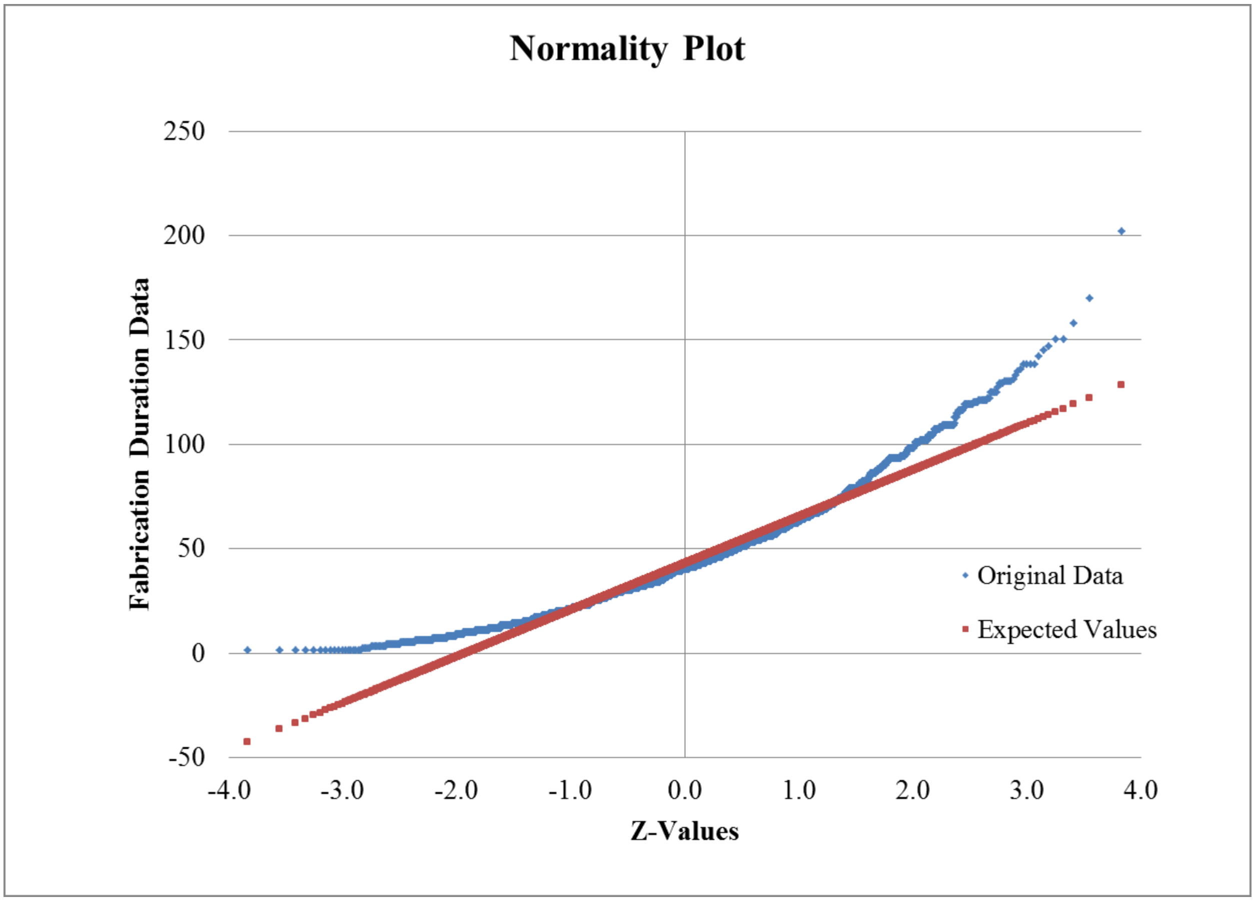


Figure 4-3 Normality Plot for Fabrication Durations

4.2.2 Logical Data Validation

In addition to statistical validation, a logical data validation exercise was performed in order to identify records that do not reflect the expected fabrication process. For example, a considerable number of records were affected by a data collection issue whereby the Material Issue Date column contains a later date than that of the completion of the fitting process. This cannot happen in practice since all the material has to be available for fabrication to begin. Given that a significant amount of data has been found to be affected by this issue (Over 15%) it had to be inferred that this information does not accurately reflect the fabrication process and therefore it cannot be used reliably. The amount of data affected may be considerably greater when one considers that the test has been performed using Fit Complete instead of Fit Start Dates, and the fitting process itself is estimated to take between a few days and a couple of weeks.

Although technically the fabrication process begins once material has been delivered to the fabrication shop, the Material Issue date could not be used for a reliable data analysis because of the aforementioned issue regarding data validity. The date when the fabrication package has been created, the Package Issue Date, had to be used instead since the data is considered to be more reliable. This process of issuing the package happens typically about a week before the material is delivered to the shop. However, if a particular package or set of packages need to be rushed, they can become available for fabrication in as little as 2 days from the Package Issue Data. Conversely, if the warehouse backlog is significant, lower priority packages can end up waiting to be delivered for up to 2 weeks. This variation in warehouse processing time is therefore included in the analysis if the Package Issue Date is considered as representing the start of the fabrication process.

To perform this data validation exercise, formulas to compare the logical sequencing of the fabrication dates were employed using Microsoft Excel. The results of this exercise, together with information regarding the normality behavior of the data and the outlier percentage, have been tabulated and are presented in Table 4-7 below. The information is expressed in terms of the number of records available, and the data validity increases with each scenario.

Table 4-7 Data Availability Summary

Scenario No.	Criteria	Number of Records	Of which Normal (median/MAD)	Outlier Percentage (median/MAD)
i)	Initial	7,911	7,840	0.90%
ii)	MIR to MTR Valid	7,906	7,836	0.89%
iii)	MIR to FitC to MTR Valid	7,773	7,705	0.87%
iv)	MIR to FitC to QCC to MTR Valid	7,099	**7,040**	0.83%
v)	MIR to FitS to FitC to QCC to MTR Valid	2,735	2,735	0.00%
vi)	MIR to FitS to FitC to QCS to QCC to MTR Valid	2,512	2,512	0.00%

Although it is desired to use a data set that is as clean and consistent as possible, as it can be seen from the summary presented in the table above, progressively more records need to be removed in order to achieve higher levels of data completeness. It is, therefore, impractical to use the highest level of logical data validation since the lack of consistency in recording the Fit Complete Date means that only approx. 40% of the records include this information. It was decided to proceed with the data available after performing validation scenario iv), which represents the best compromise between data validity and availability. Once the invalid and outlier records are removed, the remaining 7,040 records are carried over to perform the next stages of the study, namely the investigation of the fabrication schedule improvement methods.

4.3 Conclusions and Recommendations

In order to arrive at a data set that is suitable for analyzing the fabrication durations across the complete fabrication process, a thorough understanding of each activity and the planning and control methods employed at each stage was required. This was achieved by working closely with the fabrication management personnel while providing them with a map of the data collection structures and developing a system that can present all of the available information in an easily accessible and comprehensive system. The integration of a scheduling component with the spool drafting, material management and fabrication tracking information was found to add value to the management process even at a stage when the schedule was heuristically derived.

The consolidated output of the scheduling and tracking system represented a suitable dataset for analysis because of its composition and completeness. However, certain validation and cleaning steps were required, dependent mostly on heuristic rules and knowledge obtained directly from the fabrication management personnel with regards to the actual use of the data collection structures. Although it was desired to use a dataset with as many tracked milestones as possible, gaps in the available data identified during the cleaning and validation stage led to the conclusion that a simpler dataset should be further analyzed in order to maintain the majority of the available records.

CHAPTER 5. Data Driven Models

Once the data has been cleaned and validated, the total fabrication duration has been plotted again to produce an updated frequency distribution histogram, as presented in Figure 5-1 below; this exercise is mainly for visualization purposes and not for analysis at this stage. However, some preliminary observations can be made based on this data, as mentioned below.

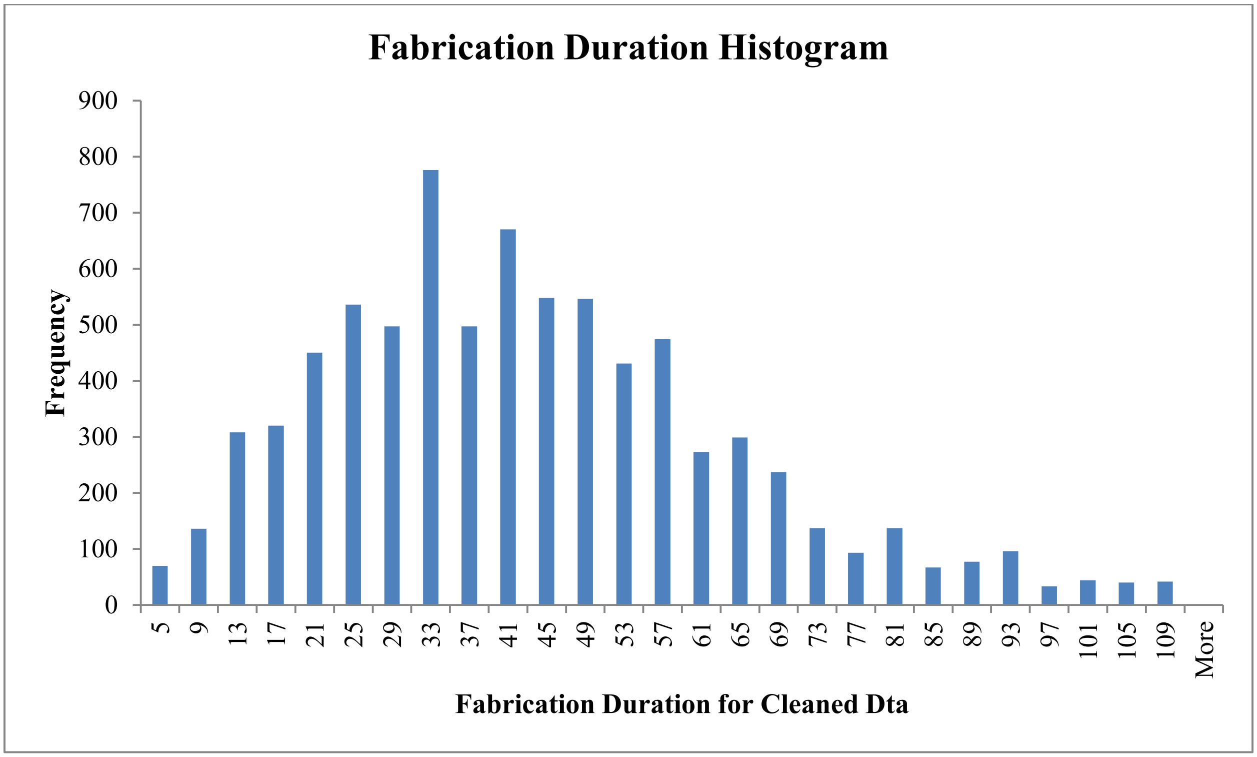


Figure 5-1 Complete Fabrication Duration Histogram

Based on the histogram presented in Figure 5-1, it can be observed that:

- The fabrication duration varies relatively widely, from a few days to about six weeks;
- The histogram is not smooth, presenting peaks and troughs that may be associated with certain criteria or spool characteristics;
- Relationships must be drawn to dissect and classify the data in order to obtain more meaningful correlations between fabrication parameters and duration.
- The histogram has a significant tail to the right, which does not taper off, suggesting a good fit will be hard to determine at this level.

A data exploration exercise has been performed using Weka Explorer (Hall, et al., 2009). The data utilized contains 7,040 records filtered following Logical Data Validation exercise presented in Section 4.2.2, and the fields arrived at following the Attribute Selection exercise in Section 5.1.The file has been formatted with the ARFF extension, which allows for the attributes properties to be read before reading the data itself.

The exploration and analysis exercise has several phases, namely:

- Attribute Selection – the stage where the available data fields are analyzed and reduced to a relevant dataset based on heuristic rules
- Instance-Based Classification – this type of classifiers have been evaluated in more detail because their performance on this data set is significantly improved over the other types of classification. The sub-section contains several stages:
    - an evaluation of the data subset using a simple KNN classifier
    - a fine-tuning of the algorithm parameters on this data set
    - a discussion on applications of the classifier at different stages of the planning process
    - a feature selection stage where the available attributes are evaluated for their impact towards classification algorithms and several attributes are removed
- Conclusions and Recommendations derived from this analysis.

The details and outcomes of each of these phases are further explained in the following sections.

5.1 Attribute Selection

In order to perform an efficient exploration and analysis exercise, it is important that the data to be analyzed contains only relevant fields, leaving out information that can obscure a good analysis. It is, therefore, desirable to keep only the fields that are expected to lead to the creation of a relationship tree between design parameters and fabrication duration. Information such as milestone completion dates is not relevant compared to the fabrication duration between these milestones. Also, the unique identifier of the spool and fields that are constant or blank will be removed as well. Also, due to concerns related to the accuracy of the information available, fabrication location information (the bay number where the spool was fabricated) will be omitted.

Processes that are only required for certain spools before they leave the fabrication shop, such as Painting, Hydro-Testing and Post Weld Heat Treatment are included in the investigation but only as Boolean attributes and not as a fixed duration allocation as they are initially available. Furthermore, the fabrication duration has been split into 3 segments according to the selection made in the logical data validation, namely, MIR to Fit Complete, Fit Complete to QC Complete and QC Complete to MTR. This operation was performed with the aim of identifying fabrication durations trends for each segment of the fabrication, and not only for the entire duration.

Although not readily available, information regarding the loading of the fabrication shop is believed to have a significant impact on the overall fabrication time by affecting fabrication priorities and waiting time between stages, as well as the congestion levels in the fabrication process. As such, it is desired to compute a measure of the shop utilization that reflects the volume of work to be performed. Based on the available fields described so far, the Diameter Inches to be welded was used in conjunction with the fabrication tracking (and hence duration information) to derive a Shop Loading parameter (or SL) that represents the shop utilization at the start date of fabrication, henceforth named Shop Utilization Index (or SUI). This index is a ratio between shop loading and a theoretical maximum capacity, referred to as Max Daily Loading, arbitrarily set to 8,000 DI/day for the fabrication shop where the data was obtained from.

To clarify, the following shop loading derivation was implemented, computed by adding the DI of all the spools that were still in fabrication at the issue date of a new spool, to derive the Shop Loading at Start Date ($SLStart_p$) presented in Equation 12.

$$SLStart_p = \sum_{t=St_p}^{Ft_p} \sum_{i}^{n} DI_{ti}\ , St_i \leq St_p \text{ and } Ft_i \geq St_p \qquad \textbf{Equation 12}$$

where

$SLStart_p$ = Shop Loading at St_p for Spool p

St_p = Start time for spool p

St_i = Start time for spool i

Ft_i = Finish time for spool i

DI_{ti} = Diameter Inches for spool i on shop floor at time t

In addition, a modified version of this index called SUI Sum represents an integration of the Shop Loading Sum over the fabrication lifespan of a spool. Firstly, the equation for Shop Loading at Start Date was modified in order to reflect a sum of welded DI in the Shop over the fabrication lifetime of each spool p in order to compute the $SLSum_p$ (Equation 13).

$$SLSum_p = \sum_{t=St_p}^{Ft_p} \sum_{i}^{n} DI_{ti}\ , St_i \leq Ft_p \text{ and } Ft_i \geq St_p \qquad \textbf{Equation 13}$$

where

$SLSum_p$ = Shop Loading Sum for Spool p

St_p = Start time for spool p

St_i = Start time for spool i

Ft_p = Finish time for spool p

Ft_i = Finish time for spool i

DI_{ti} = Diameter Inches for spool i on shop floor at time t

Next, a theoretical maximum fabrication capacity computed over the same duration was calculated based on the previously defined Max Daily Loading (Equation 14). It should be noted that the Max Shop Loading at Start and Max Loading Sum will not be utilized in the analysis directly, but are just used in order to derive the Shop Utilization Indices (SUI's).

$$SLSum_{max} = DI_{max} \times (Ft_p - St_p)$$ **Equation 14**

where $SLSum_{max}$ = Maximum Shop Loading Sum
DI_{max} = Maximum Daily Diameter Inches
Ft_p = Finish time for spool p
St_p = Start time for spool p

Lastly, the Shop Utilization Index Sum (SUI Sum) is computed simply by dividing the Shop Loading Sum by the Max Loading Sum.

Two variations of the Shop Loading Sum have been computed, using the planned durations and the historical average of actual durations in order to calculate an estimated completion date for each spool. They have been named Planned SL Sum and Mean SL Sum respectively. These variations represent alternatives to the Actual SL Sum that can be estimated more realistically.

The remaining spool information fields to be investigated, together with the duration and SUI attributes are presented in Table 5-1 below. The table also contains basic descriptive statistics for each numeric attribute. The 21 attributes can be divided into 4 distinct categories, namely:

- Spool Information Attributes, which are used to describe the spool that needs to be fabricated; these attributes are mostly numeric, with the exception of the material grade which is nominal and can take four values;
- Fabrication Details Attributes, which contain information regarding Paint, Hydro-Test and PWHT requirements in a Boolean format; and
- Shop Loading Attributes for both the start date of each spool's fabrication as well as the integration over the fabrication duration (sum), together with the SUI indices computed as previously described.
- Fabrication Duration Attributes, where the total fabrication duration and its comprising subcomponents are found. Since the total duration (MIR to MTR) is going to be the class attribute for most experiments, it has been placed at the end.

Table 5-1 Data Exploration Information Fields

Category	Attribute	Type	Min	Max	Mean	StdDev
Spool Information Attributes	Diameter Inches	Numeric	0	552	19.4	25.1
	Weight	Numeric	0	26,305	419.3	1129.5
	Surface Area	Numeric	0	763	28.0	52.1
	Nr of Items	Numeric	1	24	4.5	2.9
	Maximum Size	Numeric	1	36	4.6	4.6
	Material Grade	{A,G,E,D}				
Fabrication Details Attributes	Paint	{No,Yes}				
	Hydro-Test	{No,Yes}				
	PWHT	{No,Yes}				
Shop Loading Attributes	Shop Loading at Start	Numeric	161	43,572	30,195	13,272
	SUI at Start	Numeric	0.02	5.447	3.774	1.659
	Shop Loading Sum	Numeric	1257	109,541	57,963	23,943
	SUI Sum	Numeric	0.005	1.391	0.202	0.118
	Planned SL Sum	Numeric	764	75,073	46,814	14,896
	Mean SL Sum	Numeric	2,073	78,607	59,339	17,330
Fabrication Duration Attributes	MIR to FitC (A*)	Numeric	1	86	15.4	12.1
	FitC to QCC (B)	Numeric	1	82	8.1	8.8
	QCC to MTR (C)	Numeric	1	99	20.3	17
	FitC to MTR (B&C)	Numeric	1	106	27.4	18.8
	MIR to QCC (A&B)	Numeric	2	93	22.6	16.4
	MIR to MTR (A&B&C)	Numeric	2	110	41.9	20.9

*where A, B and C and combinations thereof in the brackets of the Fabrication Duration Attributes are just names for the duration segments that will be used in following sections.

While experimenting with the distance measure of the KNN classifier, a pre-processing filter was required to be applied in order to be able to use a tree type search algorithms; the filter replaced the missing values with the mean for that respective attribute. Although changing the search algorithm was found not to change the results by itself, the pre-processing filter did have an impact on the results of all the other various optimization runs. There is only one attribute with a significant percentage of missing records, and that is the Number of Items attribute with 11% of the records missing a value. The filter mentioned replaced the missing values with the mean of the available ones. The K* classifier results are reported using the original dataset and applying this filter does not seem to impact the results. That is because, as explained in 2.5.2 regarding The K Star Classification Algorithm, this functionality of replacing missing values with the mean is embedded in the K* algorithm.

5.2 Instance-Based Classification

Using instance-based classifiers, a set of experiments have been performed with the aim of selecting an optimal classification method, each of them representing a stage in the classification exercise, summarized as follows:

1. The performance of attribute subsets with various levels of available information has been evaluated. The expected outcome of this selection of datasets is to determine how significant different metrics related to shop loading information are to the performance of the classifiers. The most promising attribute set and relevant performance measures will be carried forward.
2. Several parameters of the available nearest neighbour type classifier implementations have been optimized with the goal of improving classification performance using the selected attribute set.
3. Using the optimized classifier parameters with the most promising set of attributes, partial fabrication durations have been added to the attribute set in order to emulate the classification at different stages in the fabrication process as milestones are reached.
4. The optimized classifier has been employed as an evaluator in a wrapper type feature selection exercise, aimed at eliminating attributes that have a negative contribution on overall performance.
5. Various other promising classifiers have been evaluated using the resulting reduced attribute set in order to reinforce the conclusion of the experiment.

Instance-based classification was selected over the other types of classification algorithms such as functions, rules, and trees, because of its promising results on this dataset. An evaluation of other data mining methods has also been conducted but given the lower performance the results are presented in Appendix 3.

5.2.1 Attribute Subset Evaluation

Subset Selection

A total of 7 subsets have been selected for experimentation with the main differentiating factor being which of the shop loading measures – or combination thereof – they utilize. Each data set combination has been assigned an experiment number, as presented in Table 5-2. A brief description of each subset is included in the table.

Table 5-2 Attribute subset experimentation

Dataset No.	Attribute Set Description	Diameter Inches	Weight	Surface Area	Nr of Items	Maximum Size	Material Grade	Paint	Hydro-Test	PWHT	Shop Loading at Start	SUI at Start	Actual SL Sum	SUI Sum	Planned SL Sum	MeanSL Sum	MIR to MTR
1	Reference set without Shop Loading parameters (Core Set)	X	X	X	X	X	X	X	X	X							X
2	Set with only the Shop Loading at the start of fabrication available	X	X	X	X	X	X	X	X	X	X						X
3	Set with only Actual SL Sum available – this attribute has a 0.7 correlation to the actual duration by itself	X	X	X	X	X	X	X	X	X			X				X
4	Set with both SL at Start and Actual SL Sum included – expected to perform best	X	X	X	X	X	X	X	X	X	X		X				X
5	Set with both Shop Utilization Indices present	X	X	X	X	X	X	X	X	X		X		X			X
6	Set with SL at Start and Planned SL Sum estimated using heuristic durations	X	X	X	X	X	X	X	X	X	X				X		X
7	Set with SL at Start and SL Sum estimated using the mean of actual durations (Base Set)	X	X	X	X	X	X	X	X	X	X					X	X

Baseline Schedule Evaluation

The performance measures have been computed for the existing fabrication scheduling approach in order to provide a benchmark for comparing the results of this subset selection exercise. The equations implemented are the ones presented in Section 2.5.4 where the observed fabrication duration was compared to the heuristically derived computation. The values for these metrics can be found in the row corresponding to subset number 0 in Table 5-3.

Subset Evaluation

In order to evaluate these various combination of attributes defined as subsets in Table 5-2, all of the performance measures presented in Section 2.5.3 have been retained for comparison using a linear search algorithm nearest neighbour classifier with k=1 neighbour; they are summarized in Table 5-3.

Table 5-3 Performance Metrics of the Attribute Subsets

Subset No.	Correlation Coefficient	Mean Abs. Error	Root Mean Square Error	Relative Abs. Error	Root Relative Squared Err.
0*	0.215	17.4	23.4	105.4%	112.1%
1	0.398	14.4	22.1	96.4%	106.1%
2	0.608	10.9	18.4	65.9%	87.9%
3	0.853	6.4	11.3	38.7%	54.2%
4	0.913	4.2	8.7	25.4%	41.6%
5	0.824	5.7	12.4	34.6%	59.2%
6	0.621	10.3	18.0	62.6%	86.4%
7	0.635	10.0	17.8	60.8%	85.1%

*values reported as subset 0 correspond to the benchmark scheduling approach.

It can be observed that the performance measures are, for the most part, consistent for each subset (i.e. they are correlated to each other); therefore it is possible to use any one of them for comparison purposes. The error related performance measures do in fact exhibit an almost exact negative correlation with the correlation coefficient, that is, they move together with performance in an inversely proportional fashion. This behavior can also be observed by examining the bar chart presenting the correlation coefficient and the percentage measures for each of the attribute subset presented in Figure 5-2. Therefore, in the interest of rigor, both the correlation coefficient and the relative absolute error will be used henceforth in order to evaluate the performance of various other classifiers.

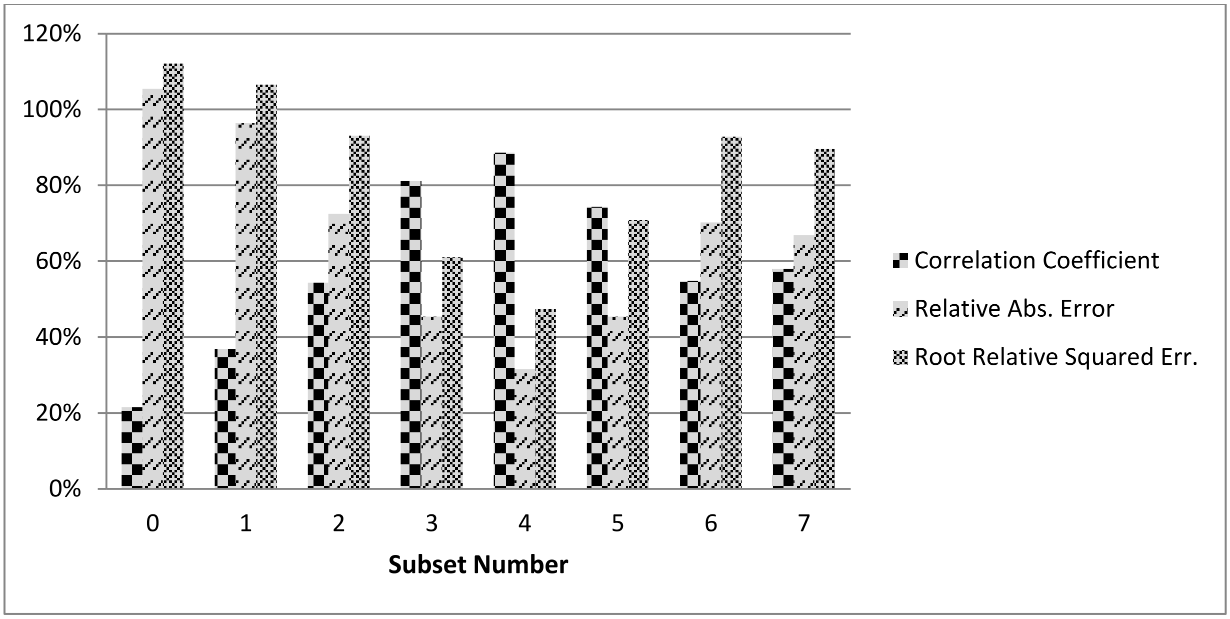


Figure 5-2 Comparison of Performance Measures for the Attribute Subset Experiment

The first observation based on the results in Table 5-3 is that the existing method for scheduling presented in Section 4.1, the one represented by Subset 0, exhibits a very poor behaviour with very high errors; a relative absolute error of over 100% is poorer performance than a simple average of historical fabrication durations would produce. Looking further at Subset 1, the first where a classification algorithm was applied, the performance is already improved but not significantly. Adding a level of Shop Loading information brings more significant improvements to the duration estimates.

The Actual SL Sum, attribute which is added by itself to the core set in Subset 3, produces the first significant improvement, bringing the correlation coefficient above 0.8, much better than the Shop Loading at Start can improve the performance by itself (Subset 2). Using them together in Subset 4 raises the performance to its peak but, as mentioned earlier, one should be careful when utilizing the Actual SL Sum attribute because it is a very difficult parameter to estimate. Using the Shop Utilization Indices in lieu of their shop loading counterparts in Subset 5 exhibits a decrease in performance compared to heir shop loading counterparts, but these indices might be easier to approximate than the loading itself.

In order to provide a similar Shop Loading attribute to the Actual SL Sum, two types of estimates have been employed: the Planned SL Sum and Mean SL Sum; they are both computed with information readily available before the start of the fabrication process and therefore can be relied upon. They have been used in conjunction with the Shop Loading At start in Subsets 6 and 7 in order to compare their performance. As expected based on the poorer than average performance of the existing fabrication plan, the Mean SL Sum performs slightly better.

Based on these observations, Subset 7 will be selected as the best compromise between performance and estimation availability for further investigations. It has a good base performance with a correlation coefficient of 0.58 and a relative absolute error of approx. 67%

5.2.2 Classifier Optimization

For Subset 7, which includes the Shop Loading at Start and Mean SL Sum, several variations of the nearest neighbour algorithm have been applied in order to fine-tune the classification results. The performance of the KNN classifier was found to vary with distance weighting and with the number of neighbours; therefore the optimization exercise was focused on selecting the best possible combination of these parameters. A number of observations can be made based on the results of the optimization exercise. These are explained in the following sections.

A summary of the classifiers' outputs expressed in terms of the correlation coefficient and relative absolute error is presented in Table 5-4. There are 2 main parts to this table according to the type of classifier implementation chosen, either the KNN or K*. The optimization then branches out for each of them according to different parameters that were varied. For the KNN these are the Distance Weighting method and the Number of Neighbours (k) and for the K* it was the blending parameter b, which has a similar function to selecting the number on neighbours in KNN and therefore was placed in the same column.

The observations regarding the optimization experiment are presented following the data table. The results are also visually represented in the figures presented on the following pages for easier visual comparison. Figure 5-3 shows all the results of the KNN classifier optimization part with k on the horizontal axis plotted on a logarithmic scale with base 3. The K* optimization results are presented in Figure 5-4 with the blending parameter on the horizontal axis.

Table 5-4 Classifier Optimization Experiment

Classifier Type	Parameters		Performance Measures	
	Distance Weighting	Number of Neighbours (k)	Correlation Coefficient	Relative Absolute Error
KNN	No distance weighting	1	0.6532	**60.82%**
		3	**0.6696**	62.90%
		10	0.6599	67.02%
		30	0.6037	73.06%
	1/distance	1	0.6532	60.82%
		3	0.6880	59.36%
		4	0.6963	**59.20%**
		5	0.6999	59.23%
		6	0.7009	59.52%
		10	**0.7078**	60.24%
		30	0.6414	68.98%
	1-distance	1	0.6532	**60.82%**
		3	**0.6700**	62.85%
		10	0.6610	66.91%
		30	0.6062	72.87%
K*		b=10%	0.6718	57.07%
		b=15%	0.6813	56.51%
		b=20%	0.6896	56.11%
		b=25%	0.6979	55.72%
		b=30%	0.7049	55.39%
		b=35%	0.7106	**55.22%**
		b=40%	0.7151	55.25%
		b=45%	0.7188	55.39%
		b=50%	0.7213	55.66%
		b=60%	**0.7227**	56.57%

KNN Classifier Results

The first observation to be made regarding the optimization of the KNN classifier is that when the number of Neighbours k is equal to 1, the distance weighting does not affect the result; this is because this algorithm parameter is used to weigh the contribution of each neighbour and when there is only one neighbouring instance it is irrelevant. As the number of neighbours increases so does the correlation coefficient, but only up to a certain maxima point, after which it declines again. The relative absolute error, on the other hand, tends to be lower with fewer neighbours.

The next set of observations regard the distance weighting method; using no distance weighting yields the poorest performance, regardless of the number of neighbours used. Marginal improvements are observed by changing to 1-distance as a method, which are so insignificant in fact that the points overlap when plotted, as shown in Figure 5-3. However, when using a distance weighting method of 1/distance, significant improvements can be observed with increasing number of neighbours. As such, at k=3 the improvement in relative error is of 3.5 percentage points, whereas at k=30 the improvement is of 4 percentage points between no distance weighting and 1/distance weighting.

After asserting that 1/distance was the best performing distance measure with the highest correlation coefficient at k=10 and the lowest relative error at k=3, a fine-tuning was performed for the number of neighbours in that region. As it can be observed from Figure 5-3, the difference is not significant – being in the region of 0.2 percentage points – but is an improvement nevertheless. A number of k=4 neighbours produces the lowest relative absolute error and is therefore the selected setting for future exercises, despite the fact that best correlation coefficient is achieved when k=10.

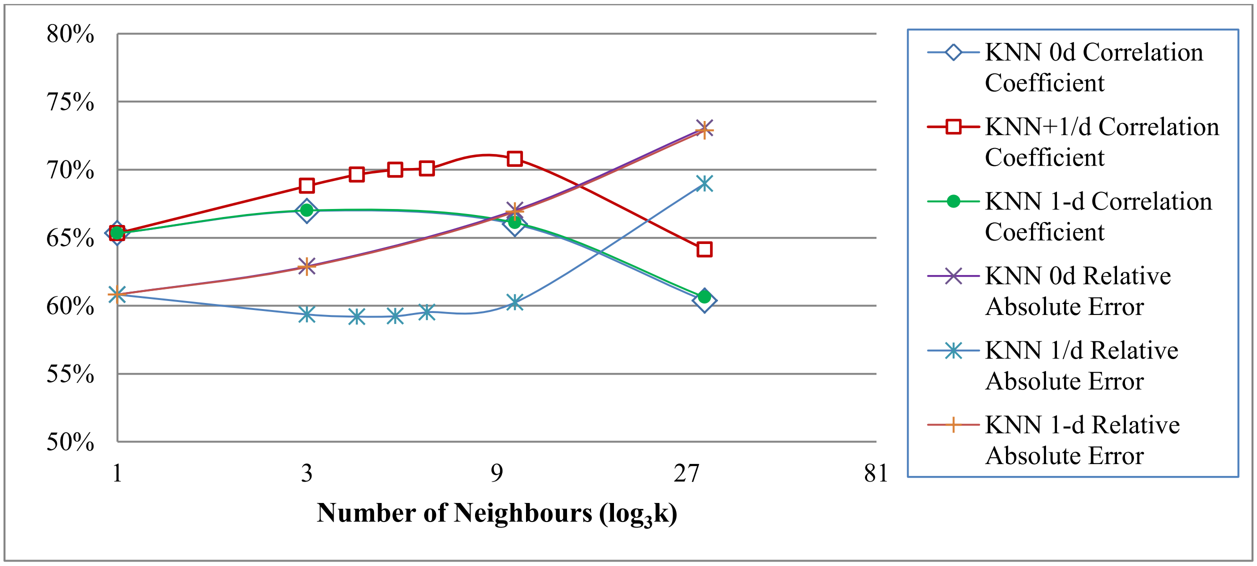


Figure 5-3 KNN Classifier Optimization Results

K Star Classifier Results

The more complex K Star algorithm exhibits better performance with this dataset regardless of the blending parameter value than even the most finely tuned KNN classifier. Optimizing for the blending parameter in 5% increments leads to the observation that the 35% global blend yields in a minimum relative error before the performance according to this measure begins to steadily deteriorate. The correlation coefficient however keeps increasing further with increasing values for the blending parameter, but this may be an indication of data overfitting behaviour. The best performing blending parameter for the K* algorithm with a value of 35% presents an improvement of 4 percentage points in relative absolute error when compared to the algorithm performance with a blending parameter of 5%.

Figure 5-4 presents the results of the K* classifier optimization exercise with the results of the KNN with 1/d as a distance weighting superimposed for ease of comparison. Although a clear improvement is observed based on this graphical representation, due to the nature of the test, which was performed on a particular set of data features, and the nature of the experiment validation itself, which uses a 10 fold hold 1 out cross validation to produce the reported results, it is believed necessary to analyze the statistical significance of this difference in the context of the standard deviation of these results.

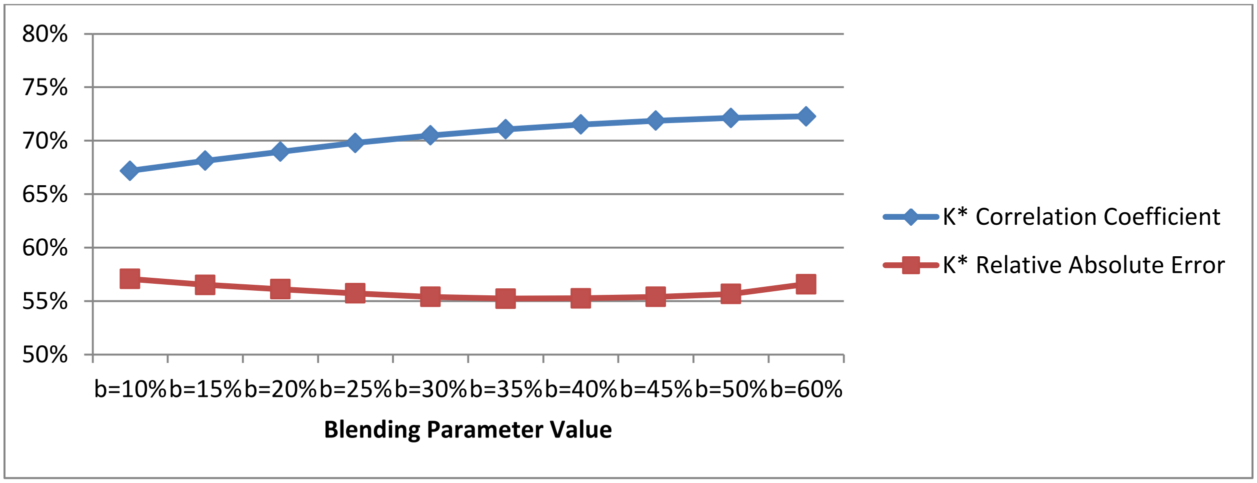


Figure 5-4 K Star Classifier Optimization Results

Evaluating the Significance of the Performance Difference

An experiment was set up in order to evaluate the significance of the performance advantage of the best K* algorithm (with b=35%) versus the best KNN classifier (1/d, k=4). The experiment comprises of 10 repetitions of the algorithm runs, each with 10-fold cross-validation. The results for each fold of each repetition were saved as a csv file in order to perform a statistical analysis using a paired corrected T-test. Two comparison fields have been selected, the correlation coefficient and the relative absolute error, in order to remain consistent with the performance measures previously selected. The T-test compares the distributions of each of the 10 repetitions once the results of the 10 folds for each repetition have been averaged. The significance factor was selected to be 0.05.

Using the relative absolute error as a comparison field, the K* algorithm with b=35% performs significantly better than the KNN algorithm with k=4 and 1/d as distance weighting. The average of the 10 repetitions is 59.43% for the KNN and 54.97% for the K* algorithm. The correlation coefficient average is 0.70 for the KNN and 0.72 for K*. For both measures the paired corrected T-Test categorized the difference as significant in favour of the K* algorithm.

Although it has been shown that the K* algorithm outperforms the optimized KNN algorithm in a statistically significant fashion, both classifiers will continue to be employed side by side in the following investigations. This is because the classifier optimization exercise focuses on a particular set of attributes and the performance gain of K* over KNN, although statistically significant, is relatively small and may potentially be reversed when testing with other subsets.

5.2.3 Applying the Classifiers

For each classifier type (KNN and K*) the settings that yield lowest Relative Absolute Error were applied to variations of the data set that contain different levels of fabrication duration information upon milestones being reached, in order to investigate if they can further improve the prediction performance. This exercise is aimed to emulate a real world application where the estimates need to be updated and refined as fabrication progresses. The base data subset was Subset 7 as described in Table 5-2, namely the one including the Shop Loading at Start and the Mean SL Sum. The attributes of the base Subset 7 are still being shown in full below for reference purposes. Combinations with different fabrication duration segments have been set-up in order to evaluate the increase in accuracy of estimation for later stages of fabrication. The class attribute was also varied from the total fabrication duration to the remaining duration segments and for each class a control run with the base subset and no additional partial durations was carried out.

The attribute selection matrix, together with the results reported in terms of the relative absolute error and correlation coefficient for each of the classifier types (KNN and K*), are presented in Table 5-5. The performance measures have also been plotted in the graph presented in Figure 5-5 for more convenient comparison and visualization. For reference purposes each subset was assigned a letter in the first column next to the radical of the base Subset 7. The class attribute was also highlighted in dark grey. Comments regarding the findings and their interpretation are presented following the table and figure showing the results.

Table 5-5 Investigation of Improvement with Added Duration Information

Subset No.	Diameter Inches	Weight	Surface Area	Nr of Items	Maximum Size	Material Grade	Paint	Hydro-Test	PWHT	Shop Loading at Start	MeanSL Sum	MIR to FitC (A)	FitC to QCC (B)	MIR to QCC (A&B)	QCC to MTR (C)	FitC to MTR (B&C)	MIR to MTR (total)	KNN Relative Absolute Error	KNN Correlation Coefficient	K* Relative Absolute Error	K* Correlation Coefficient
7a	X	X	X	X	X	X	X	X	X	X	X						X	59.2%	0.696	55.2%	0.711
7b	X	X	X	X	X	X	X	X	X	X	X	X					X	47.9%	0.776	48.2%	0.758
7c	X	X	X	X	X	X	X	X	X	X	X			X			X	39.3%	0.835	42.6%	0.797
7d	X	X	X	X	X	X	X	X	X	X	X					X		60.3%	0.680	57.4%	0.696
7e	X	X	X	X	X	X	X	X	X	X	X	X				X		56.4%	0.726	52.4%	0.724
7f	X	X	X	X	X	X	X	X	X	X	X				X			58.8%	0.686	56.1%	0.704
7g	X	X	X	X	X	X	X	X	X	X	X	X			X			53.7%	0.719	52.2%	0.726
7h	X	X	X	X	X	X	X	X	X	X	X			X	X			48.9%	0.752	49.2%	0.747

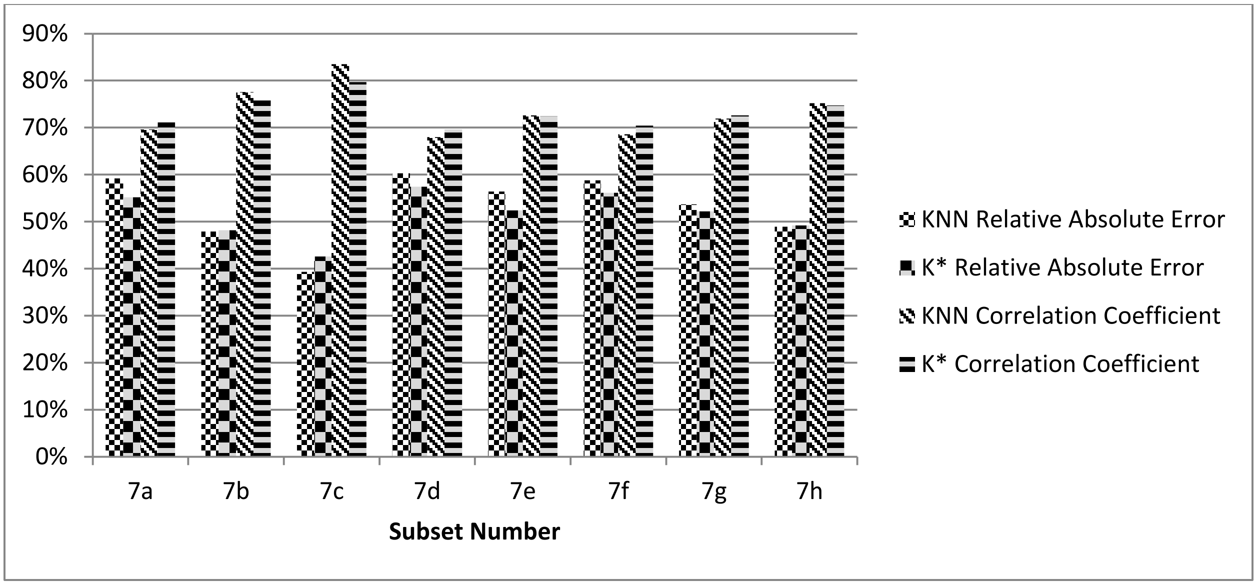


Figure 5-5 Investigation of Improvement with Added Duration Information

As expected, the result of the classification does improve significantly with added information. In the case where the class attribute remains the total duration (Subsets 7a, 7b and 7c), the improvement in relative absolute error after the first milestone, that of Fit Complete, is reached is of 11 percentage points for the KNN classifier. Once the second milestone is reached, the completion of the Quality Control, a relative absolute error improvement of a further 8 percentage points is observed for the KNN classifier. Relative absolute error improvements using the KNN classifier are not as high when classifying for the remaining segments, such as FitC to MTR (Subsets 7d and 7e) and QCC to MTR (Subsets 7f, 7g and 7h), but they are still present and significant.

For the K* algorithm computations, it can be observed that although it performs better for the base Subset 7a, its relative performance to the KNN algorithm deteriorates when adding information regarding partial fabrication completion when the class attribute is the total duration (i.e. in Subsets 7b and 7c). When the class attribute is changed to the FitC to MTR (Subsets 7d &7e), the K* algorithm maintains its advantage over the simpler KNN. For the prediction of the duration of the last fabrication segment, that of QCC to MTR, the K* algorithm continues to perform better using the base set (Subset 7f) but again loses its competitive edge when the previous duration segments are added (Subsets 7g and 7h).

An improvement in performance still exists when additional information becomes available, but the competitive advantage of the K* algorithm over its more basic cousin, the KNN algorithm, becomes less significant in most cases except for the FitC to MTR class when the advantage is maintained.

5.2.4 Feature Selection

Using the wrapper method for attribute selection described in Section 2.5.3 – that is, specifying the nearest neighbour classification algorithm as an evaluation method – and with the aim of finding the best set of attributes that can predict the fabrication duration before the execution has begun, a feature selection exercise has been performed. While keeping the attribute evaluators unchanged, namely the best performing KNN and K* algorithms optimized in Section 5.2.2, variations of the core subset (Subset 1 in Table 5-2) and the base subset (Subset 7 in Table 5-2) were investigated. New dataset numbers were assigned for comparison purposes, keeping the radical of the original subset numbering. The Results are presented in Table 5-6 below, which shows the attributes selected for each subset, as well as the performance of both classifiers, using both previously selected performance measures, namely the Relative Absolute Error and the Correlation Coefficient. The results are also plotted in the chart presented in Figure 5-6 for visual representation and ease of comparison. The interpretation of the results is presented following the table and figure.

Table 5-6 Feature Selection Investigation

Dataset No.	Diameter Inches	Weight	Surface Area	No of Items	Maximum Size	Material Grade	Paint	Hydro-Test	PWHT	Shop Loading at Start	Mean SLSum	MIR to MTR	KNN Relative Absolute Error	KNN Correlation Coefficient	K* Relative Absolute Error	K* Correlation Coefficient
1-1	X	X	X	X	X	X	X	X	X			X	84.8%	0.476	85.1%	0.475
1-2							X	X	X			X	94.4%	0.311	95.4%	0.300
1-3	X	X	X	X	X	X						X	92.2%	0.369	90.2%	0.379
1-4						X	X	X				X	91.1%	0.388	94.1%	0.336
7-1	X	X	X	X	X	X	X	X	X	X	X	X	59.2%	0.696	**55.2%**	**0.712**
7-2		X				X	X	X		X	X	X	57.6%	0.715	60.1%	0.703
7-3	X	X				X	X	X	X	X	X	X	**55.8%**	**0.732**	56.6%	0.721

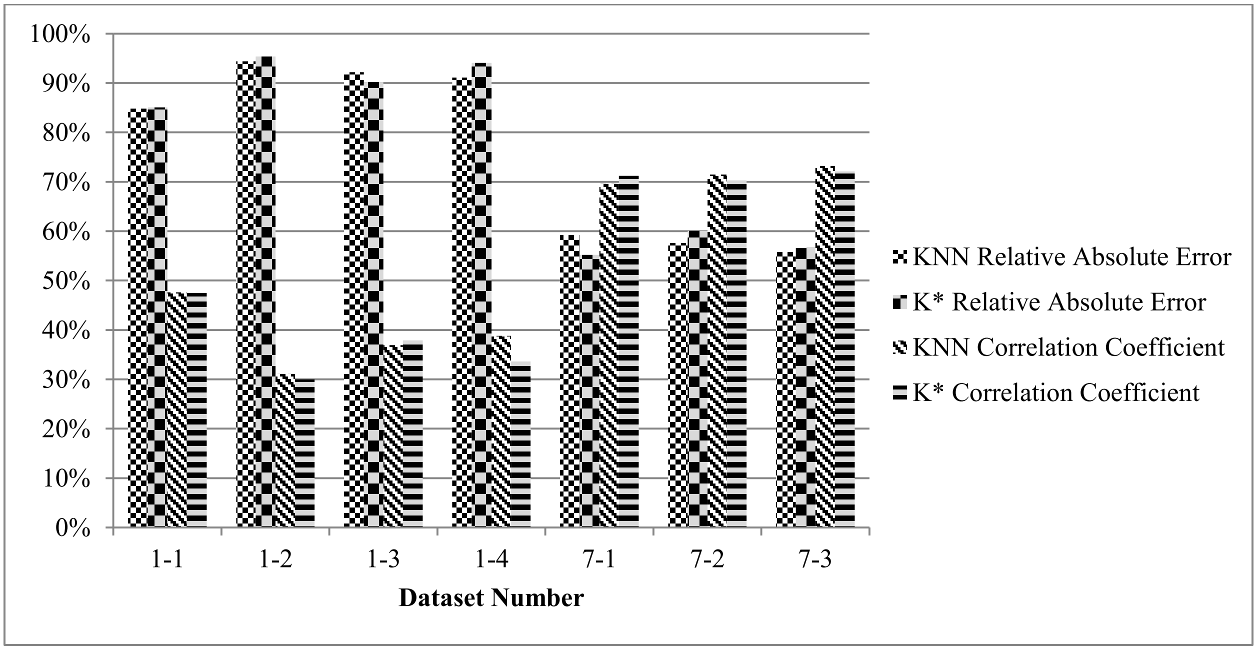


Figure 5-6 Bar Chart for Feature Selection Results

Reducing the number of attributes from the core subset (Subset 1-1) does not improve the classification performance. Subset 1-4 is the one arrived at using a genetic algorithm search method in a KNN-wrapped feature selection, whereas subsets 1-2 and 1-3 represent a heuristic selection of the core set based on the category of attribute. The behaviour exhibited by the K* classifier for this family of subsets (from 1-1 to 1-4) is similar to that produced by the KNN classifier, namely a decrease in performance is observed with reduced features. Also of note is that for the core subset 1-1, the optimized KNN classifier is slightly better than its K* sibling which provided a statistically significant improvement for the base subset 7-1.

An attempt to reduce the number of features starting with base subset (Subset 7-1) selected in the evaluation process presented in Section 5.2.1 does yield improvements following the feature selection using a genetic search algorithm (Subset 7-2). The improvement observed has a magnitude of 2.8 percentage points in relative error using the KNN Classifier with the best settings resulting from the optimization exercise in Section 5.2.2. Since removing the Diameter Inch attribute is counter intuitive considering how widely used it is in practice, it was reintroduced following the feature selection, together with the PWHT attribute which completes the picture of testing and pre-processing requirements. The augmented Subset 7-3 and was found to actually contribute to a further 0.6 percentage points reduction in the relative absolute error for the KNN classifier. Therefore, for base Subset 7, the attributes found to introduce more “noise” than useful information were: Surface Area, Number of Items, Maximum Size and PWHT.

5.3 Conclusions and Recommendations

Although the K* classifier performed statistically better on the base attribute subset (Subset 7-1) compared to the optimized KNN classifier, its performance declined during the feature selection exercise. Subset 7-3, which saw the best improvement compared to the base Subset 7-1 got to a relative absolute error of 55.8%, very close to the K* classification using the base Subset 7-1 with a relative absolute error of 55.2%. It can therefore be concluded that the K* classifier tends to find the global best result without a reduction in the number of attributes, being able to more accurately represent their influence in the classification exercise.

A recommendation based on these findings is that a K* classifier, although more computationally demanding, tends to yield improvements that are similar to a highly customized and feature sensitive KNN classifier. It is therefore recommended that a K* classifier be employed to estimate fabrication durations and its output used in calculating the estimated delivery date for each spool.

In addition, the classifier should be employed a second time once the first reliably tracked fabrication milestone, that of Fit Complete, has been reached. This will significantly improve the accuracy of the duration estimate, reducing the relative absolute error in the total duration classification by approximately 7 percentage points.

CHAPTER 6. Conclusion

6.1 Summary

This thesis began with the hypothesis that the scheduling process of pipe spool fabrication, and by implication the estimation of the delivery date, can be improved by making use of process interaction modelling and fabrication tracking data. The approach, presented herein, was to begin the investigation using a discrete event simulation model created based on typical fabrication processes and practices employed in an Alberta Manufacturing Facility. It was intended that the process simulation model will be linked to a real-time integrated production scheduling and tracking system, developed closely with the fabrication management team, once fabrication data required for the model calibration became available.

However, as was presented in the chapter describing the Data Collection and Exploration, the fabrication tracking processes and infrastructure available at the time in the manufacturing facility did not provide enough level of detail to calibrate the process simulation model and pursue its further development and validation. It was decided that the continued investigation towards fabrication duration improvements should make use of the existing fabrication tracking infrastructure while pursuing improvements towards data collection reliability with fabrication management personnel. Concurrently, the collaboration was extended towards making the integrated fabrication scheduling and tracking system as user-friendly as possible. Filtering functions were developed for the use in production control, and summary reporting functions were created aimed at improving management's visibility into the fabrication progress.

After a year of close work with the manufacturing facility personnel, the collected data was made available for further analysis, which began with a data cleaning and validation process in order to evaluate the validity and composition of the captured fabrication tracking information. This dataset was found to be less suited to the calibration of the process simulation modelling previously developed. Alternatively, an investigation using data mining techniques was performed. The aim was to improve the fabrication duration estimating process, which in turn leads to an improvement and more accurate estimates of the delivery date of pipe spools. This data mining exercise focused on the application of instance-based classification methods, which showed promising results on this dataset, especially when associating a shop loading or shop utilization component to the dataset. Since the estimation of shop loading is rather difficult, various types of estimates were added to the analysis and their contribution to the prediction of fabrication duration was discussed in the context of their expected estimation ease and accuracy. Although the initial improvements in duration estimation were found to be significant, fine-tuning the classification algorithms brought further incremental improvements.

6.2 Conclusions

The key findings based on the analysis and study methodology for each of the main parts of this research are presented herein.

Process Simulation

Although well suited for representing a process as complex as spool fabrication at a very detailed level, discrete event simulation was found to require validation data at a level of details that is very challenging to collect in a fast-paced manufacturing environment. However, a model such as the one presented in CHAPTER 3. , can provide project planners and fabrication managers with useful information about potential fabrication output and resource utilization under different operational conditions prior to the commencement of a project. In addition, it has been shown that such a simulation model can be linked with production planning and tracking data if the data structures were consistent and presented at the right level of details.

Data Collection and Exploration

The information collected by the manufacturing facility regarding spool drafting, material delivery, scheduling, and fabrication tracking was found to be structured around each process or business unit. The available data across these processes has been consolidated in order to allow visibility of the entire process at once and provide a base for scheduling improvement analysis. A data cleaning and validation exercise identified that record completeness is one of the main reasons why from a total 7,905 spool fabrication records provided by the partner manufacturing facility, only 7,040 were considered viable for analysis. This stage was selected as it provided a balanced outcome between data reliability and availability since aiming for a more complete dataset, such as one that also provided the Fit Complete date would reduce the available records to only 2,512.

KNN Methods

Instance-based classification algorithms were applied to the available dataset with the aim of predicting the fabrication duration of a spool based on its similarities with other spools with regards to its attributes. Deriving a measure of the shop utilization was found to bring significant performance improvements, although a compromise had to be made between performance gain and derivation accuracy. The dataset that contained the shop loading at start and the shop loading sum derived using the mean of past fabrication durations was selected based on the consideration that it is realistic and easy to estimate, while at the same time bringing a performance improvement of 36% decrease in relative absolute error over the reference set without shop loading parameters.

Once the suitable attribute subset was selected based on the performance/availability compromise, an optimization of the nearest neighbour algorithm parameters was performed. The findings reflect that the performance gain from fine-tuning the KNN algorithm is only in the region of 1% reduction in relative absolute error. However, employing a modified nearest neighbour algorithm that utilizes entropic blending was found to bring an improvement of a further 4% reduction in relative absolute error over the best KNN results. This improvement was found to be statistically significant using a corrected paired T-test, leading to the recommendation that this K* algorithm be employed.

Following the determination of the best performing nearest neighbour algorithm on this particular dataset, it was found that the relative absolute error of the total fabrication duration estimate can be further reduced, down to 48%, once other reliably tracked fabrication milestones, namely Fit Start, QC Start, QC Complete etc. are achieved.

A feature selection exercise performed on the dataset that yielded the best results led to the conclusion that certain attributes (Surface Area, Number of Items, Maximum Size, and PWHT) were introducing more noise to the total duration prediction than their positive contribution. Although they only affect the basic KNN algorithm and not the recommended K* variation, this reduced attribute set can be utilized when computation time is a priority since it provides results within 1% relative absolute error to the K* with a complete attribute set.

6.3 Contributions

6.3.1 Academic Contributions

The work presented herein contributes to the body of knowledge mainly through the unique application of advanced data mining techniques to the problem of pipe spool fabrication scheduling. The instance-based approach presented, that of nearest neighbour classification with the aim of predicting fabrication duration, can be further generalized to other industrial applications where the main characteristics of the sub-components to be fabricated are known and a measure of shop loading or utilization can be either measured or computed from the available data. This generalization can be made on the basis of similarities of this spool manufacturing environment with other highly dynamic industrial manufacturing processes such as the fabrication of structural steel components.

In addition to the main academic contribution presented above, a secondary contribution results from the conclusion that process modelling and simulation, although theoretically suited to such environment, is significantly more difficult to calibrate if one models the application with a high level of detail. The author considers this conclusion a valuable contribution to the industrial construction simulation domain, which recommends that future studies should not fully develop simulation models of a production system prior to the analysis of the data collection practices and resulting information structures in that particular system.

6.3.2 Industrial Contributions

The most significant research contribution from an industrial application perspective is the development of a successful data classification methodology that can be employed by the manufacturing facility without further modification. The improvements to the fabrication duration prediction accuracy are significant and will lead to better planning and sequencing of the manufacturing scope of work with a relatively simple implementation. The future implementation of this work will be facilitated by the author’s efforts to integrate the data collected between the different data sources into a unified data warehouse that has already been tested for use within the production control of the facility.

6.4 Recommendations

Given the outcome of the work presented in this thesis and summarised above, the author recommends that the output of the instance-based classification applied to the dataset with estimated shop loading (Subset 7 in Table 5-2) be used as an estimate of fabrication duration and to predict the delivery date of a spool once the issuing process has triggered its fabrication. In addition, this estimate should be updated once further information becomes available as the fabrication process progresses. This update is recommended because, as detailed in Section 5.2.3, not only does the prediction of fabrication duration improves but predicting the remaining duration makes use of the certainty of achieved partial fabrication process.

With regards to the fabrication tracking process, it has been shown that the fabrication duration estimates improve with increased levels of details. It is therefore recommended to the manufacturing facility that more efforts to capture additional information be carefully weighed against their benefits. This is particularly applicable with reference to the reliability and process definition of fabrication dates already collected.

The process simulation modelling, although not recommended to be utilized in its current format without additional calibration, is believed to be able to provide further improvements in process interaction prediction and fabrication duration estimation once further data becomes available for its verification and validation.

6.5 Limitations and Future Work

Although efforts have been made to generalize the applicability of the research presented herein to any spool manufacturing facility through process literature review, the author's experience with only one such facility might present limitations in the transferability of this research to a different spool fabrication facility. This limitation is particularly relevant in the case of the fabrication scheduling and tracking system developed with the partner company, as well as the analysis of the data made available.

In order to apply the results of the instance-based classification exercise, further work is required to integrate the data mining algorithm with the production scheduling and tracking system in order to allow new spool records to be classified as their fabrication begins and new training data to be incorporated once it becomes available. In addition, an integration of the output of the classifier with the scheduling will enable these duration predictions to be utilized in the derivation of delivery date estimates.

The process simulation model is believed to require the most significant amount of further work, both with regards to the collection of fabrication shop data at a more detailed level and with regards to the model development, calibration and validation once this data becomes available. However, the real value that this modelling approach brought extends far beyond its limitations: the data requirements of this model represented the trigger point behind the efforts to integrate the fabrication planning and scheduling process with the manufacturing tracking and control. It was this integration that the partner company found most valuable and encouraged them to continue efforts to improve their fabrication tracking processes and data collection structures independently of this investigation towards improving delivery date estimates.

Bibliography

AbouRizk, S., & Mohamed, Y. (2000). Simphony - An Integrated Environment for Construction Simulation. *Proceedings of the 2000 Winter Simulation Conference* (pp. 1907-1914). San Diego, CA, USA: Society for Computer Simulation International.

Aecon. (2012). Industrial Fabrication and Modularization. Retrieved 02 08, 2016, from http://www.aecon.com/ModuleFile/Fabrication+Brochure+2012.pdf?id=2555

Agarwal, C. C. (2015). *Data mining: the textbook.* Cham: Springer. Retrieved January 2016

Bedair, O. (2013, 11). Engineering Challenges in the Design of Alberta's Oil Sands Projects. *Practice Periodical on Structural Design and Construction, 18*(4), 247-260.

Beraldi, P., Ghiani, G., Guerriero, E., & Grieco, A. (2006, 05). Scenario-Based Planning for Lot-Sizing and Scheduling with Uncertain Processing Times. *International Journal of Production Economics, 101*(1), 140-49.

Birta, L., & Arbez, G. (2013). Modelling and Simulation: exploring dynamic system behaviour. (2nd). London: Springer-Verlang. Retrieved January 21, 2016, from http://library.books24x7.com/toc.aspx?bookid=76965&refid=R1OZ4

Braha, D., Elovici, Y., & Last, M. (2007, July 1). Theory of actionable data mining with application to semiconductor manufacturing control. *International Journal of Production Research, 35*(13), 3059-3084.

Brown, S. (2015, 12 30). *Measures of Shape: Skewness and Kurtosis*. Retrieved 02 03, 2016, from BrownMath.com: http://brownmath.com/stat/shape.htm

Brownlee, J. (2014, 03 12). *Feature Selection to Improve Accuracy and Decrease Training Time*. Retrieved 01 27, 2016, from Machine Learning Mastery: http://machinelearningmastery.com/feature-selection-to-improve-accuracy-and-decrease-training-time/

Chien, C.-F., Hsu, C.-Y., & Chen, P.-N. (2013). Semiconductor fault detection and classification for yield enhancement and manufacturing intelligence. *Flexible Services and Manufacturing Journal, 25*, 367-388.

Cleary, J. G., & Trigg, L. E. (1995). K*: An Instance-based Learner Using an Entropic Distance Measure. *Proceedings of the 12th International Conference on Machine Learning* (pp. 108-114). Tahoe City: Morgan Kaufmann Publishers, Inc.

Daniels, R. L., & Kouvelis, P. (1995). Robust Scheduling to Hedge Against Processing Time Uncertainty in Single-Stage Production. *Management Science*, 363-376.

e-Handbook of Statistical Methods. (2015, 05 13). Retrieved from NIST/SEMATECH: http://www.itl.nist.gov/div898/handbook/

Gansterer, M., Almeder, C., & Hartl, R. F. (2014). Simulation-based optimization methods for setting production planning parameters. *International Journal of Production Economics, 151*, 206-214.

Gonzalez-Villalobos, C. V. (2011). *Analysis of industrial construction activities using knowledge discovery techniques.* Edmonton: University of Alberta.

Guiasu, S., & Reischer, C. (1979). Some remarks on entropic distance, entropic measure of connexion and hamming distance. *R.A.I.R.O. Theoretical Informathics, 13*(4), 395-407.

Guyon, I., & Elisseeff, A. (2003). An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*, 1157-1182.

Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., & Witten, I. (2009). The WEKA Data Mining Software: An Update. *SIGKDD Explorations, 11*(1), 10-18.

Horibe, Y. (1973). A note on entropy metrics. *Information and Control, 22*(4), 403-404.

Hu, D., & Mohamed, Y. (2014). A dynamic programming solution to automate fabrication sequencing of industrial construction components. *Automation in Construction, 40*, 9-20.

Iglewicz, B., & Hoaglin, D. (1993). *Volume 16: How to Detect and Handle Outliers.* (E. F. Mykytka, Ed.) ASQC Quality Press.

Jackman, J., de Castillo, Z. G., & Olfasson, S. (2011). Stochastic flow shop scheduliing model for the Panama Canal. *The Journal of the Operational Research Society, 62*(1), 69-80.

Kempf, J.-F., Bogza, M., & Maler, O. (2013). As Soon as Probable: Optimal Scheduling under Stochastic Uncertainty. *Tools and Algorithms for the Construction and Analysis of Systems* (pp. 385-400). Rome: Springer Berlin Heidelberg.

Lean Enterprise Institute. (2016). *What is Lean?* Retrieved 02 08, 2016, from http://www.lean.org/WhatsLean/

Li, L., Zijin, S., Juacheng, N., & Fei, Q. (2013). Data-based scheduling framework and adaptive dispatching rule of complex manufacturing systems. *International Journal of Advanced Manufacturing Technologies, 66*, 1891-1905.

Li, Z., & Ierapetritou, M. (2008). Process scheduling under uncertainty: Review and challenges. *Computers and Chemical Engineering, 32*, 715-727.

McKay, K., Safayeni, F., & Buzzacott, J. (1988). Job-Shop Scheduling Theory: What is Relevant? *Interfaces, 18*(4), 84-90.

Microsoft. (2015, 12 1). *Microsoft Access Database Engine 2010 Redistributable* . Retrieved from Microsoft: https://www.microsoft.com/en-ca/download/details.aspx?id=13255

Moghadam, A. M., Wong, K. Y., Piroozfard, H., Derakhshan, A., & Hutajulu, T. S. (2014). Solving an Industrial Shop Scheduling Problem Using Genetic Algorithm. *Advanced Materials Research, 845*, 564-568.

Mosayebi, S. P., Fayek, A. R., Yakemchick, L., & Watters, S. (2012). Factors Affecting Productivity of Pipe Spool Fabrication. *International Journal of Architecture, Engineering and Construction, 1*(1), 30-36.

Oliveros, A. V., & Fayek, A. R. (2005). Fuzzy Logic Approach for Activity Delay Analysis and Schedule Updating. *Journal of Construction Engineering and Management, 131*(1), 42-51.

Pereira, M. T., & Santoro, M. C. (2011, 10). An integrative heuristic method for detailed operations scheduling in assembly job shop systems. *International Journal of Production Research, 49*(20), 6089-6105.

Piroozfard, H., Hassan, A., Moghadam, A. M., & Derakhshan, A. (2014). A Hybrid Genetic Algorithm for Solving Job Shop Scheduling Problems. *Advanced Materials Research, 845*, 559-563.

Ponsignon, T., & Mönch, L. (2014). Simulation-based performance assessment of master planning approaches in semiconductor manufacturing. *Omega - The International Journal of Management Science*, 21-36.

Rokni, S., & Fayek, A. R. (2010). A multi-criteria optimization framework for industrial shop scheduling using fuzzy set theory. *Integrated Computer-Aided Engineering, 17*, 175-196.

Sadeghi, N., Fayek, A. R., & Ingolfsson, A. (2012). Simulation-Based Approach for Estimating Project Completion Time of Stochastic Resource–Constrained Project Networks. *Journal of Computing in Civil Engineering, 26*(4), 558-560.

Sarin, S. C., Nagarajan, B., & Liao, L. (2010). *Stochastic Scheduling.* New York: Cambridge University Press.

Soibelman, L., & Kim, H. (2002). Data preparation process for construction knowledge generation through knowledge discovery in databases. *Journal of Computing in Civil Engineering, 16*(1), 39-48.

Stockmann, M., Haber, R., & Schmitz, U. (2012). Source identification of plant-wide faults based on k nearest neighbor time delay estimation . *Journal of Process Control, 22*, 583-598.

Wang, P., Mohamed, Y., Abourizk, S. M., & Rawa, T. A. (2009, October). Flow Production of Pipe Spool Fabrication: Simulation to Support Implementation of Lean Technique. *Journal of Construction Engineering and Management, 135*(10), 1027-1038.

Witten, I. H., Frank, E., & Hall, M. A. (2011). *Data mining: practical machine learning tools and techniques* (3rd ed.). Burlington, MA: Morgan Kaufmann Publishers.

Xie, H., AbouRizk, S., & Fernando, S. (2011). Integrating realtime project progress input into a construction simulation model. *Proceedings of the 2011 Winter Simulation Conference*, (pp. 3448-3459).

Zhao, Z., & Liu, H. (2007). Searching for Interacting Features. *Proceedings of the 20th International Joint Conference on Artificial Intelligence*, (pp. 1156-1161). Hyderabad.

Zhu, X., & Wilhelm, W. E. (2006, 11). Scheduling and lot sizing with sequence-dependent setup: A literature review. *IIE Transactions, 38*(11), 987-1007.

Appendix 1   Database Connection Code

1a) Read and Assign Initialize Code

```
Imports System.Data.OleDb
Imports Simphony.General
Imports Simphony.Simulation

Public Partial Class Formulas
   Public Shared Function Formula(ByVal context As Simphony.General.Execute)
As System.Boolean

      'Get a reference to the database element.
      Dim DB As Database = context.Scenario.GetElement(Of
Database)("Database")

      'Open a connection to the database.
      Using Connection As OleDbConnection = DB.GetConnection()

            'Create the command to execute against the database.
            Using Command As New OleDbCommand("SELECT JobNrCN, JobNumber,
            '                                            0          1
            CONTROLNO, USER20, USER3, INFABDATE, DIAINCH, USER12, PRIORITY,
            '   2        3       4        5         6       7         8
                  MatlIssueDate, BayNo, USER50, COMPLDATE, USER43, USER51,
                  '     9          10     11        12       13      14
                  USER17, USER44, USER52, USER6, USER45, USER54, STRSRDATE,
            USER53,
                  ' 15      16      17      18     19      20         21
                  22
                  USER18, RecvDate, HTPRESSURE, USER31, USER32, USER46,
            USER55, USER8,
                  ' 23        24       25         26      27      28      29
            30
                  ShippedDate, USER47, STATUS FROM SubsetA", Connection)
                  '    31        32       33

            'Execute the command against the database.
                  Using Reader As OleDbDataReader = Command.ExecuteReader()

                        'Loop through the records that were returned.
                        While Reader.Read()
                                    'Create the appropriate number of
                                    entities, initialize their attributes,
                                    and transfer them out of the element.

                              Dim Entity As New GeneralEntity
                                    ' This line changes the size of the local
                                    Strings and Floats arrays so that they
                                    each have 61 elements, indexed 0 through
                                    60.
                              Redim Entity.Floats(60)
                              Redim Entity.Strings(60)

                              Dim JobCN as String = Reader.GetString(0)
                              'Entity.Strings(0) = Reader.GetString(0)
                              Entity.Strings(0) = JobCN
                              Dim JobN As Double = Reader.GetDouble(1)
                              Entity.Floats(1) = JobN
                              Dim Spool As String = Reader.GetString(2)
                              Entity.Strings(1) = Spool
                              Dim FIWP As String = Reader.GetString(3)
                              Entity.Strings(2) = FIWP
                              Dim MIRNo As String = Reader.GetString(4)
                              Entity.Strings(3) = MIRNo
                              Dim MIRdate As Double = Reader.GetDouble(5)
                              Entity.Floats(11) = MIRdate
                              Dim DiaInch As Double = Reader.GetDouble(6)
                              Entity.Floats(0) = DiaInch
                              Dim RASDate As Double = Reader.GetDouble(7)
                              Entity.Floats(10) = RASDate
                              Dim Priority As String = Reader.GetString(8)
                              Entity.Strings(4) = Priority
                              Dim MatlDate As Double = Reader.GetDouble(9)
                              Entity.Floats(12) = MatlDate
                              Dim BayNo As String = Reader.GetString(10)
                              Entity.Strings(5) = BayNo
                              Dim FitStartDate As Double =
Reader.GetDouble(11)

                              Entity.Floats(13) = FitStartDate
                              Dim FitComplDate As Double =
Reader.GetDouble(12)

                              Entity.Floats(14) = FitComplDate
                              Dim WeldDur As Double = Reader.GetDouble(13)
                              Entity.Floats(4) = WeldDur
                              Dim WeldStartDate As Double =
Reader.GetDouble(14)

                              Entity.Floats(15) = WeldStartDate
                              Dim WeldComplDate As Double =
Reader.GetDouble(15)

                              Entity.Floats(16) = WeldComplDate
                              Dim QCDur As Double = Reader.GetDouble(16)
                              Entity.Floats(5) = QCDur
                              Dim QCStartDate As Double =
Reader.GetDouble(17)

                              Entity.Floats(17) = QCStartDate
                              Dim QCComplDate As Double =
Reader.GetDouble(18)

                              Entity.Floats(18) = QCComplDate
                              Dim PWHTDur As Double = Reader.GetDouble(19)
                              Entity.Floats(6) = PWHTDur
                              Dim PWHTStartDate As Double =
Reader.GetDouble(20)

                              Entity.Floats(19) = PWHTStartDate
                              Dim PWHTComplDate As Double =
Reader.GetDouble(21)

                              Entity.Floats(20) = PWHTComplDate
                              Dim NDEDur As Double = 5
                              Entity.Floats(6) = NDEDur
                              Dim NDEStartDate As Double =
Reader.GetDouble(22)

                              Entity.Floats(21) = NDEStartDate
                              Dim NDEComplDate As Double =
Reader.GetDouble(23)
                              Entity.Floats(22) = NDEComplDate
                              Dim MRRDate As Double = Reader.GetDouble(24)
                              Entity.Floats(23) = MRRDate
                              Dim Hydrotest As String = Reader.GetString(25)
                              Entity.Strings(8) = Hydrotest
                              Dim HydroStartDate As Double =
Reader.GetDouble(26)
                              Entity.Floats(24) = HydroStartDate
                              Dim HydroComplDate As Double =
Reader.GetDouble(27)
                              Entity.Floats(25) = HydroComplDate
                              Dim PaintDur As Double = Reader.GetDouble(28)
                              Entity.Floats(9) = PaintDur
                              Dim PaintStartDate As Double =
Reader.GetDouble(29)
                              Entity.Floats(26) = PaintStartDate
                              Dim PaintComplDate As Double =
Reader.GetDouble(30)
                              Entity.Floats(27) = PaintComplDate
                              Dim MTRDate As Double = Reader.GetDouble(31)
                              Entity.Floats(28) = MTRDate
                              Dim ShipDuration As Double =
Reader.GetDouble(32)
                              Entity.Floats(2) = ShipDuration
                              Dim Status As String = Reader.GetString(33)
                              Entity.Strings(6) = Status

                              context.Engine.ScheduleEvent(Entity, AddressOf
context.OutputPoint.TransferOut, 0)

                        End While

                  End Using

            End Using

      End Using

      'Return True to indicate that simulation should proceed.
      Return True

   End Function
End Class
```

1b) Schedule Function Code

```
Imports System.Data.OleDb
Imports Simphony.General
Imports Simphony.Simulation
Imports Microsoft.VisualBasic

Public Partial Class Formulas
```

```
   Public Shared Function Formula(ByVal context As Simphony.General.Execute)
As System.Boolean
      ' Determine Hydro Duration
      IF
Microsoft.VisualBasic.Strings.Left(context.CurrentEntity.Strings(8),1) = "H"
            Context.CurrentEntity.Floats(8)=5
      Else context.CurrentEntity.Floats(8) = 0
      End IF
      ' Determine Fit Duration (Weld -1 day)
      context.CurrentEntity.Floats(3) = context.CurrentEntity.Floats(4)-1

      'RAS to PaintC
      context.CurrentEntity.Floats(47)= context.CurrentEntity.Floats(2)
      'RAS to PaintS
      context.CurrentEntity.Floats(46)=
context.CurrentEntity.Floats(47)+context.CurrentEntity.Floats(9)-1
      'RAS to HydroC
      context.CurrentEntity.Floats(45)= context.CurrentEntity.Floats(46)+1
      'RAS to HydroS
      context.CurrentEntity.Floats(44)=
context.CurrentEntity.Floats(45)+context.CurrentEntity.Floats(8)-1
      'RAS to NDEC
      context.CurrentEntity.Floats(42)= context.CurrentEntity.Floats(44)+1
      'RAS to NDES
      context.CurrentEntity.Floats(41)=
context.CurrentEntity.Floats(42)+context.CurrentEntity.Floats(7)-1
      'RAS to PWHTC
      context.CurrentEntity.Floats(40)= context.CurrentEntity.Floats(41)+1
      'RAS to PWHTS
      context.CurrentEntity.Floats(39)=
context.CurrentEntity.Floats(40)+context.CurrentEntity.Floats(6)-1
      'RAS to QCC
      context.CurrentEntity.Floats(38)= context.CurrentEntity.Floats(39)+1
      'RAS to QCS
      context.CurrentEntity.Floats(37)=
context.CurrentEntity.Floats(38)+context.CurrentEntity.Floats(5)-1
      'RAS to WeldC
      context.CurrentEntity.Floats(36)= context.CurrentEntity.Floats(37)+1
      'RAS to WeldS
      context.CurrentEntity.Floats(35)=
context.CurrentEntity.Floats(36)+context.CurrentEntity.Floats(4)-1
      'RAS to FitC
      context.CurrentEntity.Floats(34)= context.CurrentEntity.Floats(36)+1
      'RAS to FitS
      context.CurrentEntity.Floats(33)=
context.CurrentEntity.Floats(34)+context.CurrentEntity.Floats(3)-1
      'RAS to MatlDate (5 working days between MatlDate and FitS)
      context.CurrentEntity.Floats(32)= context.CurrentEntity.Floats(33)+5
      'RAS to MIRDate (5 working days between MIRDate and MatlDate)
      context.CurrentEntity.Floats(31)= context.CurrentEntity.Floats(32)+5

      Return True
   End Function
End Class
```

1c) Update Function Code

```
Imports System.Data.OleDb
Imports System.Text
Imports Simphony.General
Imports Simphony.Simulation

Public Partial Class Formulas
   Public  Shared  Function  Formula(ByVal  context  As  Simphony.General.Execute)
As System.Boolean
      Dim      DB      As      Database      =      context.Scenario.GetElement(Of
Database)("Database")
      Dim Builder As New StringBuilder
      Builder.Append("UPDATE [Output]")
      Builder.AppendLine
      Builder.Append("SET JobNumber=")
      Builder.Append(context.CurrentEntity.Floats(1))
      Builder.Append(", ControlNr='")
      Builder.Append(context.CurrentEntity.Strings(1))
      Builder.Append("', FIWP='")
      Builder.Append(context.CurrentEntity.Strings(2))
      Builder.Append("', MIRNo='")
      Builder.Append(context.CurrentEntity.Strings(3))
      Builder.Append("', Priority='")
      Builder.Append(context.CurrentEntity.Strings(4))
      Builder.Append("', BayNo='")
      Builder.Append(context.CurrentEntity.Strings(5))
      Builder.Append("', Status='")
      Builder.Append(context.CurrentEntity.Strings(6))
      Builder.Append("', RAStoMIR=")
      Builder.Append(context.CurrentEntity.Floats(31))
      Builder.Append(", RAStoMatl=")
      Builder.Append(context.CurrentEntity.Floats(32))
      Builder.Append(", RAStoFitS=")
      Builder.Append(context.CurrentEntity.Floats(33))
      Builder.Append(", RAStoFitC=")
      Builder.Append(context.CurrentEntity.Floats(34))
      Builder.Append(", RAStoWeldS=")
      Builder.Append(context.CurrentEntity.Floats(35))
      Builder.Append(", RAStoWeldC=")
      Builder.Append(context.CurrentEntity.Floats(36))
      Builder.Append(", RAStoQCS=")
      Builder.Append(context.CurrentEntity.Floats(37))
      Builder.Append(", RAStoQCC=")
      Builder.Append(context.CurrentEntity.Floats(38))
      Builder.Append(", RAStoPWHTS=")
      Builder.Append(context.CurrentEntity.Floats(39))
      Builder.Append(", RAStoPWHTC=")
      Builder.Append(context.CurrentEntity.Floats(40))
      Builder.Append(", RAStoNDES=")
      Builder.Append(context.CurrentEntity.Floats(41))
      Builder.Append(", RAStoNDEC=")
      Builder.Append(context.CurrentEntity.Floats(42))
      Builder.Append(", RAStoHydroS=")
      Builder.Append(context.CurrentEntity.Floats(44))
      Builder.Append(", RAStoHydroC=")
      Builder.Append(context.CurrentEntity.Floats(45))
      Builder.Append(", RAStoPaintS=")
      Builder.Append(context.CurrentEntity.Floats(46))
      Builder.Append(", RAStoPaintC=")
      Builder.Append(context.CurrentEntity.Floats(47))

      Builder.AppendLine
      Builder.Append("WHERE JobNrCN='")
      Builder.Append(context.CurrentEntity.Strings(0))
      Builder.Append("';")
      Dim Querry As String
      Querry = Builder.ToString
      context.CurrentEntity.Strings(11) = Querry
      Using Connection As OleDbConnection = DB.GetConnection()
            Using Command As New OleDbCommand(Querry, Connection)
                  Command.ExecuteNonQuery()
            End Using
      End Using

      Return True
   End Function
End Class
```

Appendix 2 Scheduling and Tracking System Manual

The Scheduling and Tracking System calculates required activity start and finish dates, both as a backward pass (from the RAS date) for the “As Planned” sheet, and as forward one (from the Package Issue Date) for the “As Issued” sheet. In addition, it displays actual progress dates next to the planned ones, for comparison purposes.

The RAS dates used are the ones saved in the Acorn Fabrication Status file. These dates are entered at the drafting stage, and therefore need to be updated if the FIWP level schedule changes for any reason. Managing and updating the RAS information at a spool level can be cumbersome, but it provides great advantages, mainly the flexibility of changing the schedule only for specific spools within an FIWP.

The main objectives of the Production Schedule include:

- To provide a tool for convenient planning and controlling the execution of pipe spools;
- To allow easier tracking of spool progress;
- To provide an estimate of the Planned Ship Date based on the actual Fit Complete Date;

Data Sources

The production schedule draws information from the existing databases (Acorn, Smart Plant Materials) and re-displays the spool and schedule-related information. A diagram of the flow of data is presented below in Figure A21-1. The information regarding the file name and location for each of the required files is required from the user in the “Source” sheet of the production schedule. This information is passed on to the Scheduling Spreadsheet, and then to the Data Sheet. The code in the “Data” file deals with locating those files, making local copies of them before cleaning and consolidating the date. More details on the data consolidation and cleaning part can be found in the Data Cleaning and Consolidation . Once the information is in a single table it is passed on to the Scheduling Spreadsheet (called “SCHEDULE”), where all the formulas for pulling the required data fields and performing the necessary schedule and data comparison calculations are found. Once the two schedules are calculated (the “As Planned” and the “As Issued”), the information is passed back to the User Spreadsheet (“PRODUCTION SCHEDULE Version 2.xx”) as values. This method enables quick data queries, filters and reporting as the schedule calculations are only performed once.

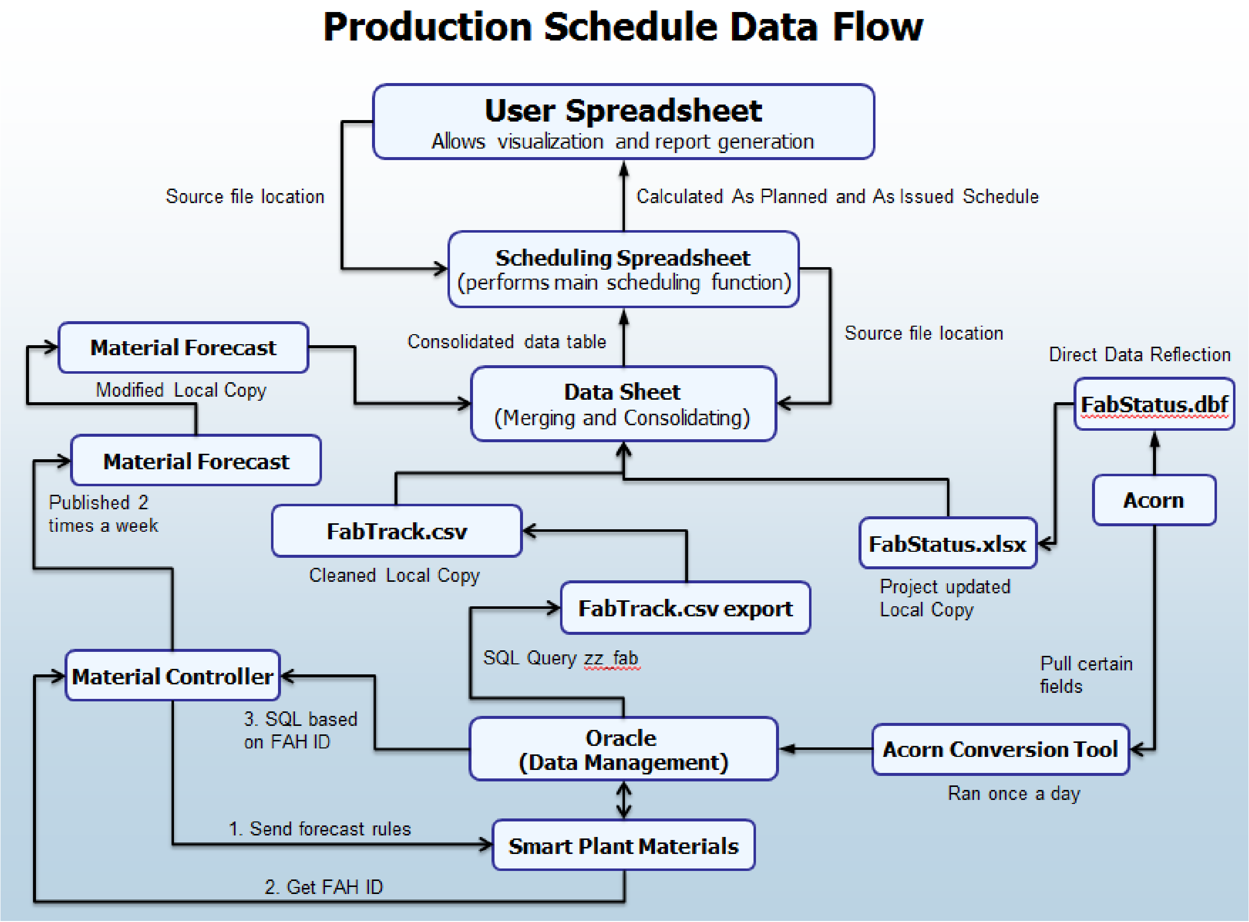


Figure A2 1 - Production Schedule Data Flow

The main data sources for the schedule are as follows:

Acorn Fabrication Status File (FabStatus.dbf)

This is the file from which most of the information is drawn from. This includes general information regarding each spool, such as the work package (FIWP) it is part of, the fabrication package (MIR) it was assigned to, fabrication durations for most of the stages, as well as most of the actual progress dates, which are currently recorded in the Acorn database.

Fabrication Tracking (FabTrack.csv Export)

Some Progress Monitoring is kept in the Smart Plant database, and as such this needs to be brought into the scheduling system. A direct connection to the Oracle database management system was attempted but because of issues such as licencing and dedicated software required led to the decision to use an existing daily export. This is considered to be reliable enough, and the 1 day delay between the data being captured and the export becoming available was deemed acceptable.

Material Forecast (Shortage Report)

As is the case with the fabrication tracking, the material availability information cannot be easily obtained from the Oracle managed Smart Plant database without the need of special software or licencing. It was therefore decided to use an existing report that is prepared twice a week for each project, and includes detailed information regarding each item of material required for the fabrication of a spool.

Functions

The scheduling system has 3 main functions: Data Consolidation, Scheduling and Reporting; in version 2, these functions are handled within a separate file for each one. This allows for a lightweight, easy to use and reliable front end of the schedule.

Data Cleaning and Consolidation

Since the schedule has 3 data sources, and each project has a different file for each of them, it was decided that the consolidation and cleaning of the data should be handled within a stand-alone, purpose-built file. The information is first copied to local records, and then cleaned to remove duplicates and other fields not needed in the schedule, before it is aggregated for all the projects. The project number is also added as a field together with a unique identifier composed of the project number last 3 digits and the control number.

This function is handled by the code in the “Data” file found in the “Supporting Files” folder, which receives the table with the file name and location for each project. It should be noted that this function limits the capacity of the schedule to 5 projects with a total of maximum 20,000 spools.

Data Cleaning

A local copy is made from each data source, saving the file with a name that is always consistent. Then data cleaning is performed for each of these files as follows:

Each of the Acorn Fab Status files is first converted from a .dbf format to a .xlsx format. Then 2 extra columns are added to the left of the data, one for the project number and another to create a unique identifier for each spool, with the structure JobNr-CN where JobNr stands for the last 3 digits of a job number and CN for the control number as it appears in Acorn (always a 6 digit identifier). No fields are deleted nor altered from this file

The Fabrication Tracking exports are left in their .csv format, but significant data cleaning is performed. The source file has data saved in 379 columns, of which only 5 are needed for the schedule. All unnecessary columns are deleted in order to minimize the file size. In addition to the required fields, a Job Number and JobNr-CN field needs to be added in order to identify the information later on. One of the challenges of processing this data is that the required fields are not always in the same position within the file, therefore a search needs to be performed based on the field headers. Since the name of these fields has been added to the code, this represents a point of inflexibility of the current development.

The Material Forecast (Shortage Reports) are left in their .xlsx format but a couple of separate sheets are added to the local copy in order to perform some data manipulation. First, the data needs to be filtered to keep only the information regarding material applicable to the "FAB SPOOL" Discipline. At this stage the Job Number and the JobNr-CN fields are added to the left side of the data. Next, in another sheet, the information is processed to keep only the fields required for the schedule. A shortage calculation also needs to be performed because of the way information is conveyed in this report; if a material item has been ordered but has not been received yet, the report will show a shortage number of zero, but a value in the PO # (Purchase Order Number) column. However, for the purpose of deciding if fabrication can be started or not that material item is not yet available.

Data Consolidation

Once all the data has been cleaned, it is aggregated or consolidated in a single file. This is a process of merging the information found in 15 files (for 5 projects) into one single data sheet.

First, the data in each category is aggregated per project. This means, for example, that all the Fab Status files will become one sheet, as will all the Fab Track files and the cleaned sheets of the Shortage Report files. In Addition to the “Source” sheet where the file location information is stored, the “Data” file has a sheet for each of the 3 data aggregated data sources.

Next, these tabs are combined in a single table, called the “Data” sheet. The Fab Status file is copied first. Then, based on the JobNr-CN unique identifier, the extra fields from the Fab Track sheet are added to the right hand side. Some spools that have been deleted will no longer appear in the Fab Track sheet but they will still be in the Fab Status file. However, the data aggregation is reliable since it is handled by code. Next, information regarding material availability needs to be calculated. Note that a simple aggregation is no longer enough as the information is stored at an item level and needs to be rolled up to a spool level. A sum of the shortage number is therefore calculated for each spool and added to the shortage column. This is handled by a formula that is added in the required cells using code.

Scheduling and Tracking

The calculations that are part of this function are handled by the “SCHEDULE” sheet found in the “Supporting Files” folder. All the formulas are found in this file only and it is here that any changes should be made if needed. The structure of this file should not be changed without checking the code of its source and destination files, as some of them anticipate a certain structure.

Using the cleaned and consolidated data, the “As Planned” and “As Issued” schedules are created. These are the main 2 components of the scheduling system and are described in more details in the Components section. The calculations that are performed regard pulling some select fields regarding general information for a spool, together with the activity durations and dates that have been tracked using the different systems. Using the RAS date, a backwards scheduling pass is performed in the “As Planned” sheet. A forward scheduling pass is performed in the “As Issued” sheet using the package issue (MIR) date.

The calculations are performed over a 5 working day calendar with custom holidays. Some of the activity durations that are not found in the data are stored, together with the official holidays, in the “Holidays” sheet. Here, the user should specify the dates within the cells highlighted in green.

The main advantage of performing the calculations in a back end file is related to the speed of operation. Since a significant number of sometimes complex formulas is required for the correct operation of the schedule, it is best that these are only performed once, at the time of an update, when some of the information changes.

These files are limited to handle up to a maximum of 20,000 spools. This limit has been applied in order to reduce computation time. It is extremely cumbersome to create a dynamic or unlimited number of spools allowed, but it is fairly simple to extend this limit to a higher capacity in the future. When modifying or extending the capacity of this file, the code of the file and its source and dependant files needs to be checked in order to maintain the system reliability.

Reporting

A simple Summary Report was developed, that shows the most important information for each project aggregated on a single Letter sized sheet. The user can specify a querying period, and the cumulative progress is also displayed. More details can be found in the “Components” section under “Spool Manufacturing Summary Report”.

Adding or Removing Projects

The schedule supports as many projects as required; the only limitation is with regards to the number of spools, which up to version 2.09 it is set to a maximum of 20,000 lines. The Information that needs to be provided is the Job Number, Client Name and the location and file names for the Acorn FabStatus, FabTrack and Material shortage files. Also, whatever fill color is used for the Job Number and Client name fields in this sheet will be the one that appears in the “As Planned”, “As Issued” and “Summary Report” sheets.

When adding or removing the information with regards to a project, DO NOT use the Insert or Delete Column functions. These will result in the dashboard and report no longer working. You shall operate on the cells only instead. The same applies when trying to re-arrange the projects – DO NOT use the Cut Column method

Current limitations regard the number of projects that the "Dashboard" and the "Summary Repot" can handle, which is 6 and 5 respectively. These are the first projects from left to right as they appear in the "Source" sheet.

Updating Procedure

One of the major updates of version 2 involved the simplification of the updating procedure; the user only has to provide updated network file locations and file names for the 3 data sources (Acorn, Fab Tracking and Material Shortage Reports) in a "Source" sheet and click the "Update" button. The information is sent to the data file, where custom VBA code deals with gathering, cleaning and consolidating the data. The new information is processed to reflect the updated data and its results sent to the Production Schedule for the user to access and review.

- ❖ Step 1: For each job, update the location and file name for the Fab Track and the Material Forecast report files. The information needs to be provided in the "Source" Sheet, in the cells highlighted in Green. For the location, it is better to use a network address, not a mapped driver. i.e. \\[...]\share\[...]
- ❖ Step 2: Run the Update Macro. Once all the files address and names have been updated, run the Update Macro by clicking the "Update" button in the "Source" sheet. It will take approx. 9 min to run so it's a good idea to do this while on your coffee break. Also, it will work faster if you are not running any other MS Office applications on your computer.

Components

There are two main sheets available for scheduling and controlling the fabrication

- The "As Planned", where the schedule is derived from the Required at Site (RAS) dates
- The "As Issued", where the schedule is derived from the Drawing Issue Date (MIR Date) recorded in acorn

As Planned Sheet

The As Planned scheduling sheet allows for a comparison against the planned progress of the spool and can be used to sequence, prioritize and coordinate the fabrication both within a project and between all projects. It can also be used to determine the sequence in which drawings should be issued as it reflects material availability information based on the latest inventory forecasts.

As Issued Sheet

The As Issued sheet can be used to control the fabrication progress as the schedule is re-calculated once the material is issued for fabrication, therefore allowing for a view that is not skewed by one of the most common cause from delays, the material delivery. Once the fitting for a spool is complete, a new estimated completion date is computed using the available durations.

Spool Manufacturing Summary Report

The Summary Report provides a high level view of the fabrication status for each project, as compiled from the Production Schedule. The data in the report reflects the information stored in the main fabrication tracking data sources, namely:

- The Acorn "Fab Status" file
- The "SPMat Fab Tracking Data" daily export
- The "SPM Shortage Report"
- The "DI Status Report"

The accuracy of the data entered is very important to ensure the accuracy of the report. Remember: "Garbage In, Garbage Out".

The report can be ran to show the progress over any period of time (last week, last month or historical) while showing the cumulative project progress.

It comprises of 3 main parts:

1. Scope and Material Availability – in DI – only one value per project
    a. **Total Scope (DI):** projected scope for each project as reflected in the "DI Status Report" (entered manually)
    b. **Material Available (DI):** summary of the information found in the latest "SPM Shortage Report"

2. Fabrication progress and backlog, expressed in DI per Period and Cumulative, including
    a. **Package Issued (DI):** the sum of DI for the packages prepared for fabrication (information reflected from the latest Acorn "Fab Status")
    b. **Material Pulled (DI):** the sum of DI of the spools for which material has been pulled (information reflected from the latest "SPMat Fab Tracking Data" export)
    c. **Warehouse Backlog (DI):** The difference between the **Package Issued (DI)** and **Material Pulled (DI);**
    d. **QC Complete (DI):** the sum of DI of the spools for which the visual inspection has been completed (information reflected from the latest Acorn "Fab Status")
    e. **Shop Backlog (DI):** the difference between **Material Pulled (DI)** and **QC Complete (DI)**
    f. **Remaining Work (DI):** The difference between **Total Scope (DI)** and **QC Complete (DI)** (Cumulative values only)
3. Number of required and shipped spools
    a. Spools Required
        i. The "Period" value reflects the number of spools with the RAS date between the period start and finish date selected, as recorded by the Acorn "Fab Status" file;
        ii. The cumulative value shows the number of spools with the RAS date between the project start date to the date the report was printed;
    b. Shipped Spools
        i. The "Period" value reflects the number of spools with the MTR date between the period start and finish date selected, as recorded in the latest "SPMat Fab Tracking Data" export;
        ii. The cumulative value shows the number of spools with the MTR date between the project start date to the date the report was printed;

Warehouse Schedule

The Warehouse Schedule provides a summary of the Production Schedule, keeping only the information needed to sequence the material pulling sequence by the Warehouse group. This is a stand-alone excel file that can be updated from the "PRODUCTION SCHEDULE Version 2.xx". The data in the Schedule reflects the information stored in the main fabrication tracking data sources, namely:

- The Acorn "Fab Status" file
- The "SPMat Fab Tracking Data" daily export
- The "SPM Shortage Report"

The information reflected is summarized at Package level, and shows all packages (complete or partial) where the drawings have been issued but the material issue date is missing from "SPMat Fab Tracking Data" – In other words, these pick tickets and cut sheets are still with the warehouse group or have been transferred over but not posted.

It is therefore possible that the schedule will show very old pick tickets, if these have been pulled but not posted. If identified, these packages should be addressed and the relevant information posted in Smart Plant.

There are 2 categories of columns in the Warehouse Schedule

1. Automatically Populated from the Schedule
    a. **MIR #**
    b. **Actual Drawing Issue Date (MIR Date)**
    c. **Planned Matl Issue Date** (calculated from the RAS Date and fabrication durations) – this is the earliest any spool from that package should have started fabrication. If this date has passed it needs to be addressed as soon as possible. This is the date by which the schedule will be sorted to get the sequence required for pulling material. Some dates might be missing here, in which case the production coordinator should be contacted to address the issue.
    d. **DI** – the sum of Diameter Inches that require welding within that MIR .
2. Manual Entry fields that need to be added to a data store
    a. **Bay #**
    b. **Received**
    c. **Pulled**
    d. **Heat# Checked**
    e. **Ready for Issue**
    f. **Comment**

Operation

The schedule allows the visualization of data from three different sources in one place, and enables the user to filter and sort this data based on any criteria deemed necessary. All current projects are implemented and they can be easily identified as the required data sheet tabs and control number cells in the schedule are colored with the assigned color for each project at a global fabrication level (i.e. the same color that is painted on the pipe and fittings).

The main scope coverage of calculations performed within the sheet includes:

- Aggregation of the data;
- Schedule dates calculation based on a 5 day working week calendar with custom holidays
- Cell lookups used to show only spool identification and collected dates
- Conditional formatting used to compare actual execution dates and durations to planned ones.
- Spool counts at various stages between two given dates selected using the Dashboard

Some of the features of the scheduling spreadsheet include:

- Highlighting late spools for each station, compared to either As Planned or As Issued Schedules
- Highlighting Spools that are on hold
- Enabling tracking by Priority, RAS (Required at Site) Date or Expected Finish Date
- Can be used for batching spools by MIR (Material Information Request) Number or FIWP (Field Installation Work Package)

The convention used for conditional formatting is that red colored dates are late compared to the scheduled ones and green colored ones are on time or earlier. Same goes for durations, the actuals are colored red if activities took longer than planned, and green if their duration was less than or equal to the planned one. Spools that are on hold are highlighted orange with red text.

Filters

The following filters are an example of what a user can obtain from the schedule. The steps presented can be performed manually or developed as a one-click button. The first two are implemented in the current version while the others were present in previous iterations and are no longer live, but their specifications are included in this this description.

- The “Clear” Filter is present on both sheets and will:
  - Shows All columns
  - Shows all drafted spools for all projects
  - Removes spools not allocated to an FIWP
  - Sorts results by:
    - Control Nr
- The Issue Filter
  - Hides All rows to the right of the Planned Matl Issue Date (except for Paint Duration and MTR Date)
  - Only shows spools that have not been issued AND have material Available

  - Sorts results by:
    - Planned MIR Date, then by
    - Control Nr.
- The “Fab Track” Filter
  - Shows all spool information and all collected actual dates
  - Hides spools that have not been Issued (MIRed) or those that have been shipped (MTRed)
  - Sorts results by:
    - Project, then by
    - RAS Date, then by
    - MIR Number, then by
    - Control Number
- The Shop Filter
  - Hides the Weld, NDE, PWHT, Paint, MTR columns
  - Shows only spools that have had material issued but are not Fit Complete
  - Sorts results by:
    - RAS Date, then by
    - Priority, then by
    - MIR Number
- The FIWP Filter
  - Hides the Weld, NDE, PWHT, Paint columns
  - Shows only spools that have been issued (MIRed) and not shipped (MTRed)
  - Sorts results by:
    - FIWP, then by
    - Expected Ship Date
  - Can show results aggregated at an FIWP level.

Challenges and Lessons Learned

Challenges

- ➢ Getting information from the drafting (Acorn) and materials (Smart Plant) databases is currently only possible via data exports, as opposed to using an SQL connection. This means the schedule relies on the exports constantly being updated.
- ➢ Consolidating the data for different projects is cumbersome and involves copying the information multiple times in order to manipulate it.
- ➢ Project files or data exports are not all in the same place, and their location can change without notice.
- ➢ As projects develop, the number of items can be in excess of 5000. The spreadsheet therefore can become very slow and crush unexpectedly.

- Computation time is very slow as all the data needs to be stored in one file and formulas re-calculate when changes such as filters are applied.
- Activity durations are heuristic, and do not consider resource availability.
- The file needs continuous maintenance as bespoke formulas or methods were developed. For example, if the location of particular information changes within a sheet it can impact the entire system.

Lessons Learned

- A spreadsheet provides and easy and quick to learn development platform. It allows all the required data to be imported and manipulated, and some reports to be developed.
- As data gets added and the spreadsheet gets bigger, it gets slower to operate and less reliable.
- Because many data sources are used, the operation of the spreadsheet requires a long time to upgrade and maintain.
- Complex formulas are needed for data manipulation. They are difficult to document and debug.

Appendix 3 Additional Data Classification

Using the core data set (Subset 1 in Table 5-2), a broader classification exercise was performed using the Weka Explorer tool (Hall, et al., 2009). The aim of this exercise is to search for a correlation between the spool information parameters and the fabrication duration using functions, rules, and tree type classifiers. Since the class of the duration data is numeric, the number of classifiers available is more limited than the availability of nominal classifiers.

The inputs to the model will be:

- Total weld diameter-inches
- Spool Weight
- Surface area of the spool
- Total number of items to be assembled
- Maximum diameter of any element in the spool
- Material grade
- The requirement for Paint, Hydro-Testing or Post Weld Heat Treatment (as a Yes/No Attribute)

The test method used for this exercise is the 10 folds cross-validation on the full training set, and will be kept consistent for all tests. At this stage the data has not been split into training and testing sets because it is not intended to keep any of the models, but only to compare how they perform against each other. For the comparison, the following classifier performance will be stored:

- Correlation Coefficient
- Mean Absolute Error
- Root Mean Squared Error
- Relative Absolute Error

Before proceeding with the classification exercise, the dataset has been reduced for each of the Fabrication Duration Attributes (see Table 5-1), such as the only one available at any one time was the one being investigated, with the other ones removed. This essentially resulted in the dataset being split into 4 subsets, one for each of the fabrication duration attributes.

Firstly, the desired output of the attribute classification is the total fabrication duration, defined from the Package Issue (MIR) date to the Spool Transfer (MTR) date. The summary of the classifier evaluation is presented in Table A3-1 below. The choice of classifiers presented was made with the following considerations in mind:

- The correlation coefficient was to be positive
- The models had to be implementable, visible in the report and non-constant
- The classification program was able to build a model and test it within 30 minutes.

Table A3-1 Classifier evaluation for the total fabrication duration

Type	**Classifier Name**	**Correlation Coefficient**	**Mean Abs. Err**	**Root Mean Sq. Err.**	**Relative Abs Err.**	**Implementation Difficulty**
Function	Least Median Squared	0.3774	14.96	19.52	90.62%	Easy
Function	Linear Regression	0.382	15.19	19.29	92.02%	Easy
Function	Multilayer Perceptron	0.3494	15.98	20.12	96.76%	Difficult
Function	SMO Regression	0.3798	14.92	19.67	90.38%	Medium
Rule	M5Rules	0.4263	14.67	18.89	88.87%	Difficult
Tree	M5P Tree	0.4453	14.43	18.70	87.40%	Very Difficult
Tree	REP Tree	0.4346	14.38	19.0	87.08%	Extreme

All classifiers presented above have a correlation coefficient of less than 0.5, suggesting a very poor classification of the data; an acceptable coefficient would be higher than 0.7 and as close to one as possible. If one was to make a choice based on these models, the implementation difficulty of the model would be one of the most important factors to consider, given the very close performance results.

The Mean Absolute Error is not a very good measure by itself unless the Mean and Standard Deviation of the data are considered. For the case of the total fabrication duration, these parameters are 41.9 and 20.9 days respectively (see Table 5-1). The Root Mean Squared Error provides an indication of the sensitivity of the deviation but again for it to be easier to analyze it shall be evaluated against the Mean Absolute Error. For ease of assessment, the Relative Absolute Error is a better measure to look at without having to compare against the original distribution parameters.

Considering how much simpler the Linear Regression function is compared to the other classifiers, its performance is quite remarkable, being only in the region of 15% less accurate than the best performing but most complex classifier, the M5P Tree.

In addition to the search for a correlation of the total fabrication duration, the classification exercise has also been applied to the search for the duration of each main segment of the fabrication process, namely:

i. MIR to Fit Complete
ii. Fit Complete to QC Complete
iii. QC Complete to MTR;

Although the first two segments of the fabrication process are not related to the requirement of Painting, Hydro-Testing or Post Weld Heat Treatment for the spools, it is the author's belief that the fabrication shop might rush spools that require some of these processes in order to keep total fabrication times low. In addition, the more attributes are made available for the classifiers, the better the resulting model should be. However, in order to prove these suspicions, the classification exercise has been performed for each of these stages on two types of data:

- A complete set with all the Spool Information Attributes and the Fabrication Details Attributes (see Table 5-1)
- A reduced set where all the Fabrication Details Attributes have been removed, keeping only the Spool Information Attributes.

The complete results tables of the classification exercise performed for each of the 3 duration segments both either the complete and the reduced datasets can be found in Tables A3-2 to A3-7. A few general trends may be observed if looking a little closer at these results side by side:

- Firstly, the correlation coefficients drop from around 0.4 for the complete durations classifications down to even the region of 0.1 for the Fit Complete to QC Complete duration (see Table A3-3 and Table A3-6)
- Using the complete data sets (including Fabrication Details Attributes) tends to yield better classifiers than the reduced data sets, as expected. The one exception here is the Linear Regression classifier for the MIR to Fit Complete duration, where the reduced data classifier has a higher value for the correlation coefficient than the one that uses the complete data set; however, as previously mentioned, the correlation coefficient is not the only nor the single best measure of classifier performance. The other classifier markers reinforce the trend that using the data sets with more attributes yields better results.
- When looking at the correlation coefficients, the overall performance of the classifiers tends to follow the mean durations of each of 3 fabrication segments. For example, although the classification for the Fit Complete to QC Complete duration has the poorest performance (around 0.1-0.2 for the correlation coefficients), its mean duration is also the shortest at 8.1 days versus the 15.4 days and 20.3 days for the MIR to Fit Complete and QC Complete to MTR durations respectively.

Given this analysis, it is recommended that if one were to use a classifier for implementation in a scheduling system, the choice should be made not only based on classifier performance, but also on the implementation difficulty of the resulting function, rule or tree. As one of the simplest of the selected classifiers, the Linear Regression function proves to be not too far behind the more complex rules or trees investigated, and would therefore be the classifier of choice for pursuing further analysis.

With regards to the decision of implanting either a classifier based on the complete fabrication duration or segments thereof, one would not be able to determine at this stage which approach would be better unless using a different analysis technique.

Table A3-2 Reduced classifier evaluation for the MIR to Fit Complete duration

Type	Classifier Name	Correlation Coefficient	Mean Abs. Err	Root Mean Sq. Err.	Relative Abs Err.
Function	Least Median Squared	0.2113	8.60	12.81	91.5%
Function	Linear Regression	0.328	9.02	11.72	95.96%
Function	Multilayer Perceptron	0.176	10.05	12.43	106.95%
Function	SMO Regression	0.2352	8.47	12.32	90.18%
Rule	M5Rules	0.2996	8.57	11.51	91.21%
Tree	M5P Tree	0.3163	8.49	11.45	90.34%
Tree	REP Tree	0.3316	8.38	11.51	89.17%

Table A3-3 Reduced classifier evaluation for the Fit Complete to QC Complete Duration

Type	Classifier Name	Correlation Coefficient	Mean Abs. Err	Root Mean Sq. Err.	Relative Abs Err.
Function	Least Median Squared	0.1038	4.871	9.41	86.99%
Function	Linear Regression	0.1258	5.52	8.71	98.63%
Function	Multilayer Perceptron	0.0418	6.39	9.43	114.25%
Function	SMO Regression	0.1047	4.84	9.25	86.36%
Rule	M5Rules	0.1936	5.43	8.63	96.96%
Tree	M5P Tree	0.2045	5.40	8.61	96.36%
Tree	REP Tree	0.2208	5.39	8.68	96.37%

Table A3-4 Reduced classifier evaluation for the QC Complete to MTR Duration

Type	Classifier Name	Correlation Coefficient	Mean Abs. Err	Root Mean Sq. Err.	Relative Abs Err.
Function	Least Median Squared	0.2905	11.46	17.06	87.81%
Function	Linear Regression	0.2988	11.94	16.23	91.53%
Function	Multilayer Perceptron	0.1757	13.50	18.24	103.49%
Function	SMO Regression	0.2963	11.34	16.78	86.91%
Rule	M5Rules	0.3593	11.56	15.88	88.58%
Tree	M5P Tree	0.3636	11.50	15.86	88.12%
Tree	REP Tree	0.3575	11.45	16.01	87.71%

Table A3-5 Complete classifier evaluation for the MIR to Fit Complete duration

Type	Classifier Name	Correlation Coefficient	Mean Abs. Err	Root Mean Sq. Err.	Relative Abs Err.
Function	Least Median Squared	0.2423	8.53	12.72	90.80%
Function	Linear Regression	0.2805	8.8654	11.58	94.33%
Function	Multilayer Perceptron	0.2164	9.86	12.32	104.91%
Function	SMO Regression	0.2762	8.38	12.13	89.21%
Rule	M5Rules	0.372	8.26	11.20	87.94%
Tree	M5P Tree	0.38	8.19	11.17	87.10%
Tree	REP Tree	0.3863	8.12	11.25	86.41%

Table A3-6 Complete classifier evaluation for the Fit Complete to QC Complete Duration

Type	Classifier Name	Correlation Coefficient	Mean Abs. Err	Root Mean Sq. Err.	Relative Abs Err.
Function	Least Median Squared	0.1357	4.85	9.39	86.62%
Function	Linear Regression	0.1604	5.48	8.67	97.95%
Function	Multilayer Perceptron	0.0876	6.36	9.37	113.61%
Function	SMO Regression	0.1459	4.80	9.17	85.80%
Rule	M5Rules	0.2181	5.38	8.58	96.16%
Tree	M5P Tree	0.1984	5.41	8.65	96.66%
Tree	REP Tree	0.2659	5.33	8.56	95.18%

Table A3-7 Complete classifier evaluation for the QC Complete to MTR Duration

Type	Classifier Name	Correlation Coefficient	Mean Abs. Err	Root Mean Sq. Err.	Relative Abs Err.
Function	Least Median Squared	0.3849	10.73	16.4	82.19%
Function	Linear Regression	0.3953	11.12	15.63	85.82%
Function	Multilayer Perceptron	0.3271	12.51	16.96	95.91%
Function	SMO Regression	0.3938	10.57	16.16	80.97%
Rule	M5Rules	0.4766	10.52	14.96	80.66%
Tree	M5P Tree	0.4889	10.49	14.84	80.40%
Tree	REP Tree	0.4798	10.39	15.02	79.65%