Manufacturing Execution Systems Integration and Intelligence

Basil Hadjimichael

Department of Electrical and Computer Engineering
Centre for Intelligent Machines – Systems and Controls Group
McGill University, Montreal, Canada
August 2004

A thesis submitted to McGill University in partial fulfillment of the requirements of the degree of Masters in Electrical and Computer Engineering.

McGill University	**Abstract of Master's Thesis**

Author: Basil Hadjimichael

Name of Thesis: Manufacturing Execution Systems Integrations and Intelligence

Date: 05.01.2004 **Number of Pages:** 100 + 30

Department: Electrical and Computer Engineering

Research Group: Centre for Intelligent Machines - Systems and Controls

Supervisor: Benoit Boulet

In order to survive in today's competitive manufacturing markets, manufacturing systems need to adapt at an ever-increasing pace to incorporate new technology which can lower the cost of production, while maintaining quality and delivery schedules. The task of the manufacturing system becomes even more challenging in the quest to use a common approach for different manufacturing plants and ever evolving manufacturing processes for specific plants. This thesis introduces a reference architecture that enables such changes between plants and updates within plants. For this, we use the paradigm of Manufacturing Execution Systems (MES). A developed MES architecture by the National Institute of Standards and Technology (NIST) is used as the standard reference architecture. Its flexibility and scalability is applied to a specific steel melt-shop plant case study. In this case study the standard framework is specified through re-labeling standard data and modules to specifics tailored for the melt process of a generic steel plant. Since steel plants are faced with difficult scheduling and disturbance handling problems, specific intelligent algorithms are developed to deal with these issues through integrating some of the control into the MES. Conclusions as to the success of the algorithms along with supporting data and recommendations of further use for them are also included.

Keywords: Manufacturing Execution Systems (MES), integration, architecture, foundation, intelligence, algorithms

Résumé

Afin de survivre dans les marchés compétitifs d'aujourd'hui, les systèmes manufacturiers doivent s'adapter à une vitesse de plus en plus grande pour incorporer de nouvelles technologies qui peuvent diminuer les coûts de production tout en maintenant la qualité et les échéances de livraison. La tâche des systèmes manufacturiers devient même plus difficile dans la quête de l'utilisation d'une approche commune pour des manufactures de différents types et dans l'évolution constante des procédés manufacturiers.

Cette thèse présente une architecture de référence qui permet d'interchanger les usines et les différentes mises à niveau des systèmes. Pour ce faire, nous utilisons le paradigme des systèmes d'éxécution manufacturière (SEM). Une architecture développée par le *National Institute of Standards and Technologies* est utilisée comme architecture de référence. Sa flexibilité et sa capacité d'expansion sont appliquées à une étude de cas pour une fonderie. Dans cette étude, l'architecture de référence SEM est spécialisée par le ré-étiquettage des données et des modules pour le procédé de fonte d'une aciérie générique. Puisque les aciéries font face à d'importants problèmes d'ordonnancement et de gestion des perturbations dans la production, des algorithmes intelligents sont développés pour résoudre ces problèmes en les intégrant comme lois de commande dans le SEM. Nous concluons sur le succès de ces algorithmes tout en fournissant des recommandations sur leur utilisation future.

ACKNOWLEDGEMENTS

I would like to express my gratitude to my supervisor Professor Benoit Boulet, for guiding me through this vast research area but allowing me to explore and learn on my own, and then helping me focus on an approach to handling this research.

To prior students Angelique Mannella and Li Xu of Professor Benoit Boulet, I would like to thank for providing me with a starting point for this research with a basic approach to the melt-shop scheduling problem. Finally I wish to thank my friend Don Weaver for helping with the formatting and editing of the final version of this thesis.

TABLE OF CONTENTS

Chapter 1 ... 1
1. Introduction, Background Literature ... 1
1.1. Manufacturing and MES ... 1
1.1.1. Manufacturing and Management ... 1
1.1.2. Manufacturing Execution System (MES) ... 2
1.1.3. What Else MES Can Bring ... 3
1.2. MES REQUIREMENTS ... 3
1.3. HOW MES WORKS ... 5
1.4. PROBLEMS AND AREAS OF CONCERN ... 7
1.5. Benefits ... 10
1.6. APPLICATIONS ... 11
1.6.1. Prototypical MES Plant ... 11
1.6.2. Specific Plant Applications for MES ... 12
1.6.2.1. FABRICATION AND ASSEMBLY (AUTOMOTIVE) ... 12
1.6.2.2. REGULATED (FOOD AND BEVERAGE) ... 13
1.6.2.3. LIFE SCIENCES (PHARMACEUTICAL) ... 13
1.6.2.4. METAL MANUFACTURING ... 14
Chapter 2 ... 15
2. MES Module Interface Architecture ... 15
2.1. Introduction ... 15
2.2. Process Planning and Manufacturing Execution Boundary ... 15
2.3. Funda mental Activity Model ... 16
2.4. Manufacturing Execution Activity Model ... 17
2.4.1. Whole Plant Enterprise – Develop Product Model ... 17
2.4.2. Enterprise Resource Planning ... 19
2.4.3. Acquire Resources ... 19
2.4.4. Manufacturing Execution Model with Data Breakdown ... 19
2.4.4.1. Develop Operation Sequence & Detailed Schedule ... 23
2.4.4.1.1. Develop Operations Sequence and Detailed Schedule for Work Stations ... 24
2.4.4.1.2. Define Production Units ... 25
2.4.4.1.3. Schedule Production Unit ... 25
2.4.4.1.4. Schedule Resource ... 26
2.4.4.1.5. Allocate Resource ... 26
2.4.4.2. Dispatch Production Units ... 26
2.4.4.2.1. Material Movement ... 26
2.4.4.2.2. Direct Personnel ... 28
2.4.4.2.3. Release Production Units ... 28
2.4.4.3. Track Production Units and Resources ... 28
2.4.4.3.1. Inventory Tracking and Management ... 28
2.4.4.3.2. Monitor Personnel ... 29
2.4.4.3.3. Manage Process ... 30
2.4.4.3.4. Manage Quality ... 30
2.4.4.3.5. Track Products ... 30
2.4.4.4. Manage Factory-Floor Data/Document ... 30

2.4.4.4.1. Collect Product Data 32
2.4.4.4.1.1. Receive Workpiece Measurement 33
2.4.4.4.1.2. Acquire Production Unit Condition 33
2.4.4.4.1.3. Monitor Equipment Condition 33
2.4.4.4.2. Analyze Data 33
2.4.4.4.2.1. Perform Statistical Analysis 34
2.4.4.4.2.2. Analyze Product Throughput 34
2.4.4.4.2.3. Provide Product Genealogy 35
2.4.4.4.2.4. Analyze Machine/Personnel Performance 35
2.4.4.4.3. Manage Document 35
2.4.5. Manage Sales and Services 35
2.4.6. Control Equipment and Process 36
Chapter 3 37
3. MES Module Melt Shop Case Study 37
3.1. Introduction 37
3.2. Melt Shop Model 37
3.2.1. Short Background 37
3.2.2. Problem Definition: 38
3.3. System Model 42
3.3.1. Overview of the System Model 42
3.3.2. Input Module 44
3.3.3. Resource Modules 45
3.3.3.1. The EAF Module 45
3.3.3.2. The LF Module 46
3.3.3.3. The Caster Module 47
3.3.4. Machine Coupling in the Model 49
3.4. Manufacturing Execution Activity Model 50
3.4.1. Whole Plant Enterprise – Develop Steel Billets 51
3.4.2. Enterprise Resource Planning 52
3.4.3. Acquire Raw Material 52
3.4.4. Manufacturing Execution Model with Data Breakdown 52
3.4.4.1. Develop Operations Sequence and Detailed Schedule for Heat Stations 56
3.4.4.1.1. Optimize schedule for heats defined to be done 56
3.4.4.1.2. Define Production Unit Routing 57
3.4.4.1.3. Produce Start and Finish Times for Schedule 58
3.4.4.1.4. Reschedule Resources (Failure Recovery) 58
3.4.4.1.5. Allocate PERSONNEL 58
3.4.4.2. Dispatch Heat Units 59
3.4.4.2.1. Heat Material Movement 59
3.4.4.2.2. Direct Plant-Floor Workers 60
3.4.4.2.3. Release Heat Units 60
3.4.4.3. Track Heats and Machines 61
3.4.4.3.1. Heat Inventory Tracking and Management 61
3.4.4.3.2. Monitor Plant-Floor Workers 62
3.4.4.3.3. Manage Heat Process 62
3.4.4.3.4. Manage Heat Quality 63

3.4.4.3.5. Track Heats ... 63
3.4.4.4. Manage Plant-Floor Data/Document ... 64
3.4.4.4.1. Collect Heat Data ... 65
3.4.4.4.1.1. Receive Heat Measurement ... 65
3.4.4.4.1.2. Acquire Production Unit Condition ... 65
3.4.4.4.1.3. Monitor Equipment Condition ... 66
3.4.4.4.2. Analyze Data ... 66
3.4.4.4.2.1. Perform Statistical Analysis ... 67
3.4.4.4.2.2. Analyze Heat Throughput ... 67
3.4.4.4.2.3. Provide Product Genealogy ... 68
3.4.4.4.2.4. Analyze Machine/Personnel Performance ... 68
3.4.4.4.3. Manage Document ... 68
3.4.5. Manage Sales and Services ... 68
3.4.6. Control Equipment and Process ... 69
Chapter 4 ... 70
4. Algorithms ... 70
4.1. GENETIC ALGORITHM DESIGN (A411 Module) ... 70
4.1.1. Encoding the problem through a chromosome ... 71
4.1.2. Population of Chromosomes ... 72
4.1.3. Penalty Factor and Objective Fitness Function ... 73
4.1.3.1. Makespan and Total Cost of a Schedule ... 73
4.1.3.2. Defining the Penalty Factor ... 74
4.1.3.3. Deriving the Multi-Objective and Fitness Function ... 74
4.1.4. The Selection Method ... 77
4.1.5. Crossover and Mutation Methods ... 78
4.1.5.1. Crossover Algorithm ... 78
4.1.5.2. Mutation Algorithm ... 80
4.1.6. Regeneration ... 81
4.2. GA FOR OTHER APPLICATIONS ... 81
4.3. Disturbance Handling ... 81
4.4. Machine Failure Recovery ... 82
4.4.1. Machine Monitoring ... 82
4.4.2. Decoding the Status of the Melt Shop ... 82
4.4.3. First Stage (LF) Rescheduling ... 84
4.4.3.1. First Stage (LF) Rescheduling without Turnaround ... 84
4.4.3.2. First Stage (LF) Rescheduling with Turnaround ... 85
4.4.4. Second Stage (EAF) Rescheduling ... 85
4.4.5. Final Stage of Rescheduling ... 85
4.5. POWER LEVEL FLUCTATIONS ... 86
4.6. STATEFLOW AND MATLAB GA INTERFACING ... 87
4.7. CONCLUDING REMARKS ABOUT APPLICATION ... 87
Chapter 5 ... 88
5. Algorithm Results ... 88
5.1. Performance of The Genetic Algorithm ... 88
5.1.1. Analyzing the Schedule of the Genetic Algorithm ... 92
5.1.2. Using Elitism in the Genetic Algorithm ... 93
5.1.3. Concluding Remarks about the GA ... 95

5.2. The Performance of Production Unit Rescheduling Algorithm 96
Chapter 6... 98
6. Final Remarks and Conclusions.. 98
6.1. RECOMMENDATIONS AND FUTURE WORK... 98
6.2. CONCLUSIONS... 99
7. REFERENCES 102
APPENDIX A...105
APPENDIX B...114
APPENDIX C...118
APPENDIX D...120
APPENDIX E...123
APPENDIX F...126
APPENDIX G...129
APPENDIX H...131

LIST OF FIGURES

Figure 1.1: A Unified Data Repository for Plant Systems....9
Figure 2.1: Develop Products (Activity A0)....17
Figure 2.2: Activity A0 – Develop Products....18
Figure 2.3: Activity A4 – Execute Manufacturing Orders....23
Figure 2.4: Activity A41 – Develop Operation Sequence and Detailed Schedule....24
Figure 2.5: Activity A42 – Dispatch Production Units....27
Figure 2.6: Activity A43 – Track Production Units and Resources....29
Figure 2.7: Activity A44 – Manage Factory Floor....31
Figure 2.8: Activity A441 – Collect Product Data....32
Figure 2.9: Activity A442 – Analyze Data....34
Figure 3.1: Melt Shop Machines....39
Figure 3.2: SIMULINK Model of the Melt-shop....43
Figure 3.3: Input Queue Model....44
Figure 3.4: Model of EAF and EAF Timer....46
Figure 3.5: Model of LF Timer....47
Figure 3.6: Model of Caster, Caster Timer and Turnaround Timer....49
Figure 3.7: Activity A0 – Develop Products....51
Figure 3.8: Activity A4 – Execute Manufacturing Orders....55
Figure 3.9: Activity A41 – Develop Operation Sequence and Detailed Schedule for Heat Stations....57
Figure 3.10: Activity A42 – Dispatch Heat Units....60
Figure 3.11: Activity A43 – Track Heats and Machines....61
Figure 3.12: Activity A44 – Manage Plant-Floor Data/Document....64
Figure 3.13: Activity A441 – Collect Heat Data....65
Figure 3.14: Activity A442 – Analyze Data....67
Figure 4.1: Solution Encoded as a Chromosome....72
Figure 4.2: Roulette Wheel Selection....77
Figure 4.3: Crossover Algorithm....79
Figure 4.4: Mutation Algorithm....80
Figure 5.1: Convergence of Fitness Function....90
Figure 5.2: Convergence of Makespan....91
Figure 5.3: Convergence of Penalty....91
Figure 5.4: Convergence of Cost....92
Figure 5.5: Fitness with Elitism....94
Figure 5.6: Problem in the GA without Elitism....95
Figure 5.7: GA Outputs using Production Unit Rescheduling Algorithm....97

LIST OF TABLES

Table 2.1: A4 Module Data 21
Table 3.1: Heat Orders 38
Table 3.2: Cost Associated with Transition Steel 40
Table 3.3: Machine Characteristics 40
Table 3.4: Jobs Involved in Steel Production 41
Table 3.5: A4 Module Specific Case Study Data 53
Table 4.1: Schedule represent by 32 21 13 Encoding 72
Table 4.2: Makespan for Schedule 1-2 2-2 3-2 73
Table 4.3: Heat Time Values for Machines 76
Table 4.4: Optimal Schedule: 61 31 22 51 42 12 83
Table 5.1: Heat Time Values for Machines 89
Table 5.2: Comparison of Original and Final Populations 89
Table 5.3: Optimal Schedule: 61 31 22 51 42 12 93

Chapter 1

1. Introduction, Background Literature

1.1. MANUFACTURING AND MES

1.1.1. Manufacturing and Management

When the optimization of a manufacturing process is considered there are several key issues which are prime targets for improvement or streamlining. These include; not meeting promised delivery dates, machines failing thus causing production to halt, inefficiencies in production flow due to bottlenecks, and failing to pass quality assurance (QA). These issues do not standalone, meaning significant improvement can be made if a proper system is in use to handle these problems in a mutual way.

Now if the attention is shifted onto some of the contributions made to manufacturing on the plant floor more tangible forms of control are considered, such as programmable logic controllers (PLCs), robots and computer-aided manufacturing (CAM). All these use automation; the idea of using computers to program and control production equipment. These controls however remain isolated on a localized level, which means each computer is focused on its specific task and does not know what the other computers are doing. More recently a higher level of control known as Supervisory Control and Data Acquisition (SCADA) has come into use. This is used to provide more of a relationship among the different localized islands of computers. Still all this data gathered by the data acquisition components of the system is large and disorganized and creates several important problems. It is important to decide whom in the company will need what data and whether it will be relevant by the time they receive it. Also, can someone read all this data and respond within the right time?

When assessing the problem from the point of view of manufacturing in terms of production planning, the most prevalent tool used is Enterprise Resource Planning (ERP).

ERP encompasses distribution, product data management, and supplier management. It can also include Material Requirements Planning (MRP) – a planning tool made for creating manufacturing schedules for purchasing, accounting and inventory control. Given product quantities required from the master production schedule, the MRP determines the quantity to make or purchase. Lead-time is taken into the equation to determine when these items must be ordered or manufactured. Even with these sophisticated software packages, the execution does not always go according to plan. This is mostly due to inaccuracies in forecasting, production slowing down due to bottlenecking and process inefficiencies. As a result, "there exists a wide information gap, which is untenable in today's competitive world where customers expect instant fulfillment of orders at globally competitive prices and with the highest quality" [18].

Clearly, there is something missing that can be implemented to greatly improve manufacturing. The machines residing on the plant floor and the planning taking place in the management offices for the use of those machines, does not seem to link in any distinct path. Higher-level functions may have an impact on lower level activities such as process control, and thereby an indirect effect on the process. Similarly, low-level events such as a sensor or actuator failing will have a negative impact on the high level management and may call for revising the plans. Manufacturers need software which can improve all areas of the manufacturing process including better performance, higher consistencies and quicker response to adapting needs from customers, suppliers and internal processes. The traditional tools such as ERP, supply chain, customer relationship, and product life-cycle management systems are not sufficient to achieve the efficiencies required by today's low margin, quick response and scalable market. Manufacturers need a tool which will provide operations with fast, accurate and transparent data. That tool is a Manufacturing Execution System (MES).

1.1.2. Manufacturing Execution System (MES)

An MES controls the production operations that enable the realization of management's plans, and provides feedback from the plant-floor to management thereby closing the execution gap. "MES delivers information that enables the optimization of production activities from order launch to finished goods. Using current and accurate data, MES guides, initiates, responds to, and reports on plant activities as they occur"[14].

The MES is made to feed all the other enterprise systems. The contribution to supply chain inventory accuracy and reduction is seen through the MES's management of work orders in the plant. Management of the product life cycle is also important for MES since it holds the "as built" information. Due to these accurate views into the production status MES allows for more effective customer service, and is particularly essential as companies move closer to "real-time" response.

1.1.3. What Else MES Can Bring

The research of MES is still in its infancy, and the application of true computer engineering and science to manufacturing issues is still in its beginning stages. Imagine where e-business would be without the urgency of demand for computer science and engineers. Manufacturing applications have a much greater potential economic impact and slowly more and more software suites and integrators are coming to the market.

This research focuses on the enablement of the MES to feedback set points and control operations to the plant floor systems. From a control engineering perspective, this requires the MES to not only upload reliable status information from low-level operations and analysis information from high-level management, but also to have built in algorithms for making control decisions.

1.2. MES REQUIREMENTS

In the early 1980's, a number of companies used different size computer systems to assist manufacturing operations. At that time, the general term for these systems was Computer-Integrated Manufacturing (CIM). The idea was to use computers to accomplish total integration of the manufacturing organization in to a coherent integrated whole. MES has branched off from CIM by providing a more organized infrastructure. Unlike CIM, a custom software program from plant to plant, MES has taken on a more standard approach for reusable application software, which results in less implementation time and cost.

To allow for this standard reusable application, MES requires broader system design considerations for all software and hardware to allow for full integration. Meaning all systems must be able to communicate information and not be excluded from the overall system environment. The concept of the information historian database where each computerized activity draws from and delivers data to the system is becoming increasingly important.

Based on SP88 [17], the most fundamental driver for system integration is the need to improve the ability to continuously change a business to meet the changing market demands and utilize new technologies. The need of flexibility for future needs is also required when designing an MES. The design of the software (interface) and hardware (gateway) must have an easy method for expansion to meet the increasing needs of the company as new and better ideas emerge in MES and as operation upgrades to the production process occur. However, to achieve an integrated flexible manufacturing environment at an affordable cost it is evident that the work in the area has to be synchronized. With synchronization comes the need to breakdown all the requirements of the MES. This starts with a framework of data that will be needed to pass the right information between interfacing systems in the plant, to meet the common goals. The next chapter presents just that.

In helping with the exchange of information between the plant software systems, one standard that is being adapted by MES is open connectivity or OPC. OPC is a series of standards specifications. The first standard (originally called the OPC Specification and now called the Data Access Specification) resulted from the collaboration of a number of leading worldwide automation suppliers working in cooperation with Microsoft. Originally based on Microsoft's OLE COM (component object model) and DCOM (distributed component object model) technologies, the specification defined a standard set of objects, interfaces and methods for use in process control and manufacturing automation applications to facilitate interoperability. The COM/DCOM technologies provided the framework for software products to be developed by reducing the custom programming.

While broadening the audience of information users and not being too restrictive is important, so too is providing adequate security, primarily to ensure data integrity. Many MES software suites are adapting a user login security password that grants access to certain information in the MES system, depending on the user's privileged access.

1.3. HOW MES WORKS

As stated earlier, MES presents a standard suite of functions that are generic enough to apply to any type of plant foundation. However, as the production protocols in the plant become more specific, so to does the MES. There are eleven generic functions found in an MES suite, which are defined by the S95.01 [16] standard provided by the Instrumentation, Systems and Automation (ISA). These functions are not the be all and end all of MES functionality but are meant to provide a solid category of capabilities. The eleven functions are:

- Operations/Detailed Scheduling
- Resource Allocation & Status
- Dispatching Production Units
- Document Control
- Product Tracking & Genealogy
- Performance Analysis
- Labor Management
- Maintenance Management
- Process Management
- Quality Management
- Data Collection Acquisition.

There are many other specific systems depending on the type of production process that MES may need to exchange information with, such as a laboratory information system (LIMS) or an energy management system. However, for an MES to function it must integrate with the automated control systems that run the plant floor and the ERP to exchange information with management. With this, real-time data can be implemented to allow MES to report and respond to plant activities as they occur, resulting in rapid response to changing conditions. This two-way plant-wide communication enables the optimization of production activities from start to finish.

The primary goal of MES and part of the objective of this thesis is to provide the information system that can be used for optimizing production activities in a facility with focus on quick response to changing conditions. An MES can also perform some of the following subgoals based on manufacturing order execution: [13]

1. to improve communication inside a facility; for example, production activities can be rescheduled to reflect unexpected machine down time or production priority changes
2. to improve communication capability between production and other activities in a manufacturing enterprise, such as product design, process planning, resource planning, supply chain management, service and sales, and equipment control,
3. to monitor production to control operations within desired performance parameters,
4. to provide up-to-the-minute communication between the facility and facility management, and
5. to better manage production-related data, including resource data, performance data, process data, job scheduling data, equipment/device control programs, and so on.

A point of recent contention is whether MES needs to be supplied by independent software vendors, or whether it fits better as an extension of process control systems, or the corporate-wide ERP systems. The concept of Enterprise Production Systems (EPS) [13] looks at improving MES functionality by uniting its execution system applications with the plant automated control systems layer. Some of the capabilities this adds to MES are to allow for one framework for MES and the control systems layer. EPS includes all software and control systems logic used to manage the plant floor functions and processes. This includes equipment and process sensors, equipment and PLCs as well as any and all software used to control or influence production. The business and system design reasons to support this approach are [13]:

1. A more clearly divided system; one for planning (decision support) and the other for execution (transaction processing).
2. The physical separation is a better fit; one for the corporate view and the other for the local plant view. Although there will be some overlap and the systems are mutually supporting.

3. Allows for a corporate wide strategy separated from localized production facility systems.
4. Provides a better environment for system improvement and upgrade (either corporate or local).

This thesis presents the EPS because it focuses more on the goal of the research for providing more application on the control side of MES. Specifically through intelligent scheduling and decision making that feeds directly to the plant control systems layer. However, for the purpose of this thesis a division between MES and control systems layer will be maintained, and the intelligence will be assumed to reside in the MES and will be separated from the control systems layer.

1.4. PROBLEMS AND AREAS OF CONCERN

In a large extended enterprise a reliance on the web of suppliers and partners the business engages become vital to the production. This is a problem external to the plant. The reliance on this extended enterprise can be improved through a distributed MES. A paper by Huang [8] explores this need for developing distributed MESs to integrate the distributed operations that are external to the plant but vital to the production operation.

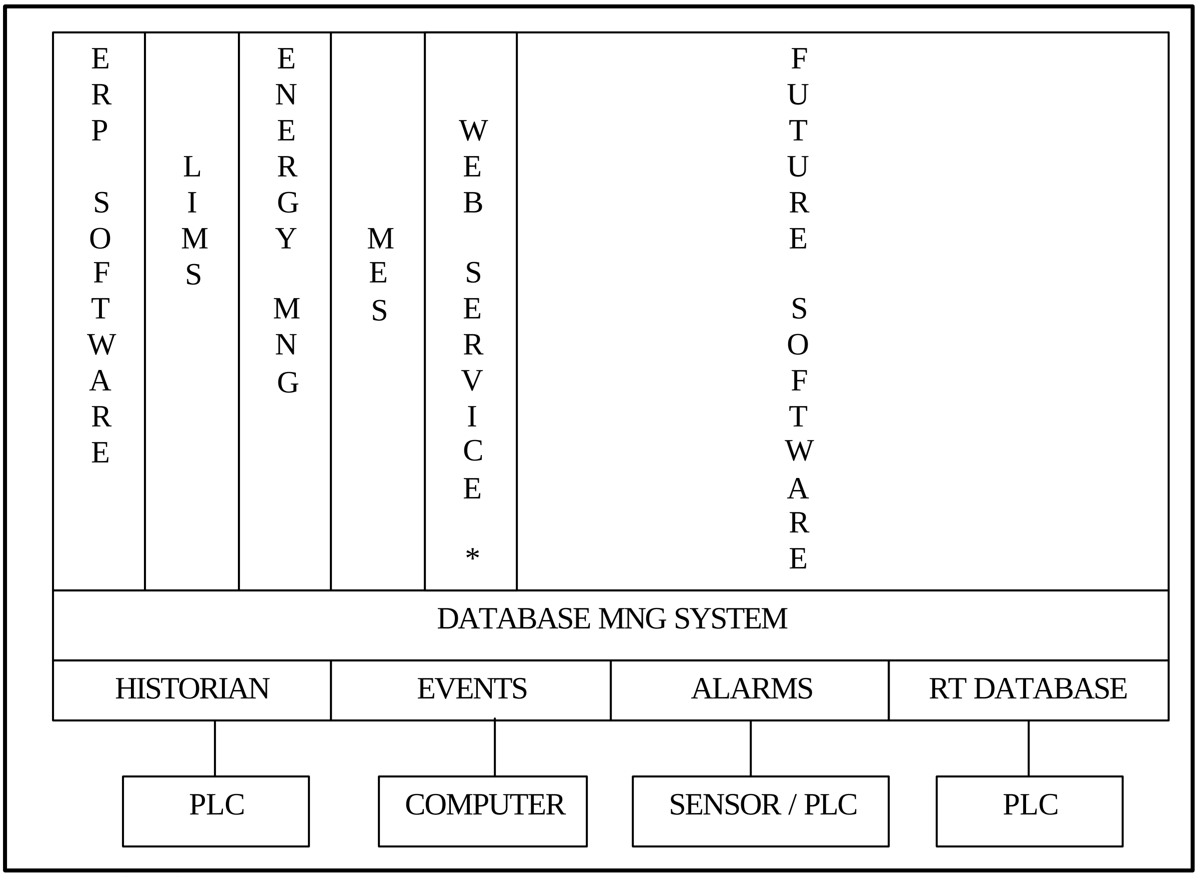
Some of the main areas of concern and problem issues that MES is faced with internally to the plant are the difficulties in interfacing with existing software and the level of detail that must be incorporated when implementing an MES from plant to plant. The first problem arises due to the different islands of automation found in a plant. These islands are sophisticated within their activities; however, difficulties arise from the portability of the information that is required from them. This is especially true when trying to incorporate MES into old existing systems that run on legacy software. The other area of concern comes from defining the details of the MES for a specific plant. What constitutes a good enterprise MES in one case may make it unsuitable in another.

With these integration and implementation challenges, an appropriate solution for a plant is to build and implement a cost-effective, integrated MES appropriate to their unique needs. First, the present state of integration and the availability of infrastructure (hardware and software) must be assessed. Then the information barrier can be handled by mapping the information needs of all the operational layers. Based on this, the framework of the MES has to be designed for the plant-specific conditions. This thesis presents an MES framework based on Feng [4] that defines the fundamental modules of an execution system and maps the specific information needed to outline the generic needs of all the operational layers. The thesis then applies this MES framework to a plant-specific case study to show its application.

With this framework a key parameter to consider is the frequency of information. As there is a wide disparity in the frequency of information polled by various systems, ranging from milliseconds to days, intermediate data storage and a transmission layer (process historian) are required. The quantity of information to be stored should be considered when determining the size of this buffer; since too small a database will make the system less effective but faster, and too large a database will make it slower and more expensive but more comprehensive. In this thesis the MES framework goes as far as breaking down the time scale for the specific data that must be exchanged in the plant.

Dealing with the complexity of integration for the various data sources and/or existing software packages in the plant is furthered here. This issue of concern can clearly be seen by considering the requirements needed to create a seamless system with the existing systems at an affordable cost based on low complexity integration. Some of these requirements consist of getting for each package the source code and having the ability to make revisions, designing and developing the drivers to enter and receive data from each package, and designing and developing the various data passing and logic conditions as well as identifying clean data issues. Aside from this, whenever any changes are made the entire system will again require a detailed analysis to ensure system functionality is not compromised. Also change-documentation and testing must always be current or there will be no way to debug or rely on revisions. To add to the matters, this problem compounds since each package requires its own revision plus the revision of all the others. In a typical plant there could be ten or more systems (PLC, control systems and system software). Revising each of these and making upgrades is very expensive, therefore integration is usually not done.

There are some new ideas and software coming to market that makes this process much easier [13]. The idea is based on a single unified data repository. Such a repository may be actual, in the form of a data warehouse, or virtual, in the sense that different software packages on various levels of the MES model make the data available as if they were part of an overall repository. The diagram in *Figure 1.1* below shows how various applications are linked.



* For Distributed MES

Figure 1.1*: A Unified Data Repository for Plant Systems*

This change for building a repository for all the data in a plant has been in process for some time but has been accelerating with the firm establishment of software standards in manufacturing such as Microsoft DNA. With systems following common standards, the ability to integrate functions has become much easier, less costly, and faster to implement.

Keep in mind that regardless of industry or plant type, failed MES implementations have a few important characteristics in common. They are: attempted with inadequate planning, based on poorly drafted specifications, not aligned with business priorities and not based on quantifiable return-on-investment [10].

1.5. BENEFITS

Aside from the difficulties presented by implementing an MES, the possible benefits of a successful MES can vastly improve not only a manufacturing process, but by streamlining and optimizing all of the individual components of a process it can allow for a company to grow more organically without the need for large overhauls of their systems. MES delivers substantial benefits by improving both sides of manufacturing. For the operations side, it reduces lead-time on new product start-ups, lowers work-in-process (WIP) inventories, improves product quality, reduces paperwork and provides speedier data-entry. MES provides real-time management of complete process definitions, production and support activities, assuring compliance to engineering plans and specifications. It helps greatly in achieving corporate goals including realizing improved return on assets, lowering operating costs, reducing capital expense and improving regulatory compliance. By improving plant efficiencies and reliability, MES boosts return on the fixed assets tied up in plant and equipment. It contributes not only to return on invested capital and return on assets, but also to overall enterprise information flows.

One of the specific benefits stated by [18] is shorter manufacturing cycle times (average 45% reduction). Other benefits that industry [10] feels MES can offer are faster inventory turnover and lower in-progress inventories, more on-time deliveries with better quality (fewer defects), and higher return on operational assets, improved gross margins and enhanced cash flow. Along with a coordinated and synchronized plant, business and supply chain processes, better real-time business decisions from automated dataflows, a clear view of how manufacturing processes are meeting business objectives, a solid foundation for continuous improvement in manufacturing plants, tighter integration of shop floor information with the supply chain and quantifiable value, through rapid return on investment.

It is without a doubt that with these industry benefits, MES is an area that deserves further research and contributions by academia and industry. Specifically, this thesis concentrates on the area of control, by providing ideas and specific algorithms that help MES in developing more autonomous and intelligent event-driven optimization.

1.6. APPLICATIONS

1.6.1. Prototypical MES Plant

An MES system can apply to any type of plant that produces a product from start of order to finished good, ranging from mid-size to large-scale manufacturing. There is no simple distinction that defines what exact type of application would need an MES. A typical plant that runs complex, nonlinear processes that require dynamic flexibility and interactive execution systems could use an MES. MES provides a powerful and practical means for users to target their greatest production challenges, and then plans solutions implementation, while minimizing risks and costs. The point to consider is the customization of what specific MES functions (listed in Section 1.3) should be incorporated into the application requirements. To meet the needs of the application, as mentioned in Section 1.4 all of the requirements must first be defined through adequate planning based on well drafted specifications that are aligned with business priorities. Most importantly the plant must determine whether the up front investment on products and integration, as well as continuous support can provide a reasonable return on investment.

MES software companies are beginning to provide a suite of standard modular products that support flexible deployment. In doing so this allows manufacturers to tackle the important problems to start with and not all the problems at once. By being built on a common framework, they allow the solution to be developed and extended easily over time at the required pace of the manufacturer. The deployment of an MES solution is a long-term project. In the world of middleware solutions there are no 'one-step fixes'. Therefore, a series of incremental steps will have to be taken by the manufacturer to overcome a handful of challenges at a time. Each manufacturer will have different needs that can be addressed at different times in different orders of priority.

1.6.2. Specific Plant Applications for MES

1.6.2.1. FABRICATION AND ASSEMBLY (AUTOMOTIVE)

On a global scale for competitiveness, automotive manufacturers have embraced a new standard. This standard is based on providing improved manufacturing speed, responsiveness and agility while still meeting the high quality and reliability consumers expect. With E-Business driving more demand-based models of build and assemble-to-order manufacturing, success must be calculated on an enterprise scale requiring multi-plant execution common architectures that enable cross-plant communication and support the application of the best practices across the entire enterprise. This only heightens the need for distributed MESs [8].

These initiatives are supported by a MES using real-time, bi-directional connections for delivering critical data to supply chain and enterprise planners. WIP visibility can be used to determine the status of every customer order in the assembling process. Data collection can help monitor vendor quality and component genealogy for liability and warranty issues. Efficiently executing build orders requires a balance between consuming assembly plants and suppliers. This is done by closely coupling suppliers to the build schedule, along with real-time order status on the production flow, material and subassembly consumption and quality issues per part.

1.6.2.2. REGULATED (FOOD AND BEVERAGE)

This Food and Beverage industry is characterized by inventory management, product quality, safety and cost control. It is faced with the demands of producing shorter supply chains, production runs and response times, along with reducing variable schedules, changing customer requirements and inventory levels. To improve control and management of production and maintain peak plant efficiency, real-time plant visibility must be incorporated. This will allow for tracking WIP and obtaining real-time production data at critical control points in the production process. By doing this, identification of HACCP (Hazard Analysis and Critical Control Points) compliance can be improved and product quality issues can be tagged before they become high-exposure incidents.

In this environment of intense change where agile manufacturers develop new products regularly, the need for MES is clear. MES can provide a real-time, interactive link from the plant floor to the ERP to help both areas in improving profitability.

1.6.2.3. LIFE SCIENCES (PHARMACEUTICAL)

Industry segments such as Pharmaceutical are defined by accuracy. Increased competition, globalization and high research costs contribute to the complexity and significance of the manufacturing plants. Current good manufacturing practices (GMP) and exact record keeping requirements are enforced by strict FDA regulations for this industry. Information systems are required to deliver associated details on product quality, efficacy and genealogy. To add to this already multifaceted manufacturing process comes the complexity of using electronic records and signatures compliance (see standard 21 CFR Part 11 of the FDA, [20]). Manufacturers are dealing with these demands by stressing product quality, shortening new product introduction cycles, and reducing unit costs. In pharmaceutical operations product tracking is used to provide real-time visibility into batch/order status, on-line release of raw materials, WIP and finished goods. Here manufacturing visibility is needed to create a basis for the analysis, reporting, and trending needed to accelerate product lifecycle management.

Manufacturing strictness is achieved using all the important real-time data obtained through the integration of internal and external supply chains. E Manufacturing is the future for this industry. Companies must realize this if they wish to compete in the industry as shorter manufacturing cycle times and improved labor and equipment utilization are becoming more widespread. For a specific application that outlines the system architecture and functional description of an MES realized in the pharmaceutical industry refer to Fisher-Rosemount Systems [5].

1.6.2.4. METAL MANUFACTURING

In the steel manufacturing industry, management is faced with a competitive customer market, where high energy costs must be considered with the demands of increased quality at lower prices. To be competitive when it comes to meeting these demands MES must be used in order to produce on-time customer orders and shorter delivery cycles. Real-Time information direct from the floor is provided through the MES for production management to reduce re-processing rate and increase first-time quality production. In this market the competitive advantage must be obtained by combining better operational management with business systems to plant floor connectivity, just as MES offers.

Improved quality, customer responsiveness and efficiency are fundamental business drivers. An industry focused on bottom line management needs execution strategies that maximize plant capacity, throughput and utilization of resources. This thesis presents methods to do just that through algorithms designed to optimize the available resources based on the customer orders and plant status.

Chapter 2

2. MES Module Interface Architecture

2.1. INTRODUCTION

Manufacturing technology is finding new applications. "These do not come from generating more data, as current systems are overflowing with data already, but through a better job of integrating the information into more usable forms for more widespread use" [13].

Some of the important factory floor information and activities are: resource allocation, dispatching production units, quality management, operations planning, detailed scheduling, labor management, product tracking, and keeping records of product genealogy. Systems that support linked sets of information flow and activities such as CAD, CAM, CAPP, ERP, control, and scheduling will be at an advantage from an integration perspective if these software systems can exchange data and messages (function calls) with each other in a wide-ranging computing environment.

To provide a better understanding of where this is going, a set of foundation modules for manufacturing execution and a set of foundation interfaces are identified through universal object modeling in Interface Definition Language (IDL). The interfaces in this report are in their first stages, and are made to set the context for Manufacturing Planning and Execution Software interfaces development. Improvement updates of the interfaces will take place in the future as the technology and practices evolve.

2.2. PROCESS PLANNING AND MANUFACTURING EXECUTION BOUNDARY

There exists an important boundary between process planning (computer aided process planning (CAPP) systems) and detailed planning activities taking place at the manufacturing execution level (decision/scheduling activities). Process planning deals with the specification process, which defines operations, their sequences, and the selection of manufacturing resources available before any product is produced. Manufacturing execution encompasses the detailed operation planning/decision making and scheduling performed at the workstation level after the work order is issued or during the production.

Only the major functions in an MES are described below, the content of interfaces described enable general software interpretation in the product manufacturing domain. The model realization depicted from *Figure 2.2 - Figure 2.9*, specific module descriptions (A1 to A6 and their sub modules), terminology and data definitions (provided in Appendix A) for facilitating the understanding of the model are based on Feng's findings [4]. The Appendix A index vocabulary includes two subsections: Terminology and ICOM (Input, Control, Output, and Mechanism) Data Definitions. Common terms referred in activity and ICOM data definitions are defined in the terminology subsection. All the data elements in ICOM are defined in the subsection of ICOM definitions.

2.3. FUNDAMENTAL ACTIVITY MODEL

The fundamental activity model of product development (in reference to a product realization model in Barkmeyer, ed. [2]) is presented below. This model is provides a standard convention to grow and build on. The top-level activity of Develop Products (A0) is indicated by its name given in the box (*Figure 2.1*). All the data that interfaces with the box is part of ICOM data. On the left side of the box with arrows pointing into the box indicate the Input data. This data is transformed by the activity to the output data. This is done using the Control data, which are used to regulate the internal process of the activity. The Control data flows in from above the box with arrows pointing into the top of the box. The Mechanism data also influences the resulting output with its method for handling and enabling the activity. The Mechanism data is presented with an input through the bottom of the box. Output data which can be transferred to other activities to be used for either Input data and/or Control data are on the right side of the box with arrows pointing out from the box.

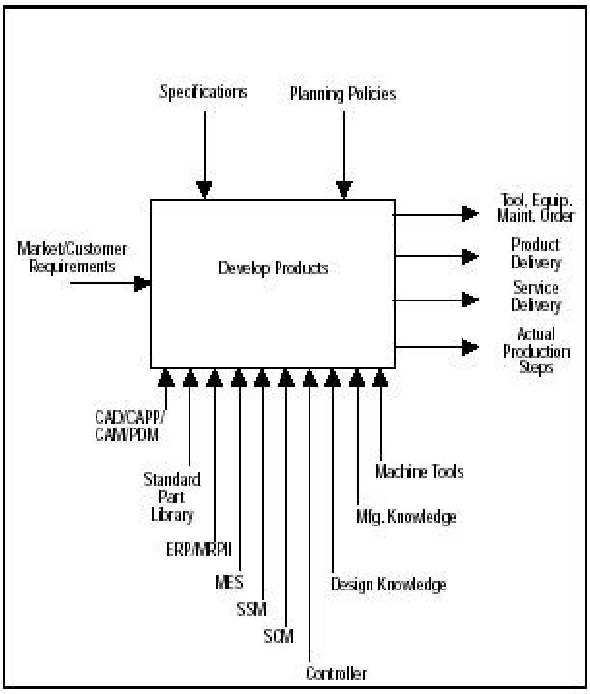


Figure 2.1: *Develop Products (Activity A0)*

2.4. MANUFACTURING EXECUTION ACTIVITY MODEL

2.4.1. Whole Plant Enterprise – Develop Product Model

The Activity A0 is decomposed into a series of six subactivities, as depicted in *Figure 2.2*. The highlighted data are primary to the A4 block. Activities A1, A2, A3, A5, and A6 are presented to show their relations to A4. The Engineer Product in A1, A2, A3, A5, and A6 are out of the scope of further definition for the MES interfaces development.

The Engineer Product and Process (A1) activity is broken down into the following two engineering groups; (1) designing products and (2) generating a manufacturing process plan along with alternative plans. The first group for designing products includes:

- functional requirements
- conceptual design
- embodiment design
- detailed design
- design analysis
- bill of material (*Product BOM*)

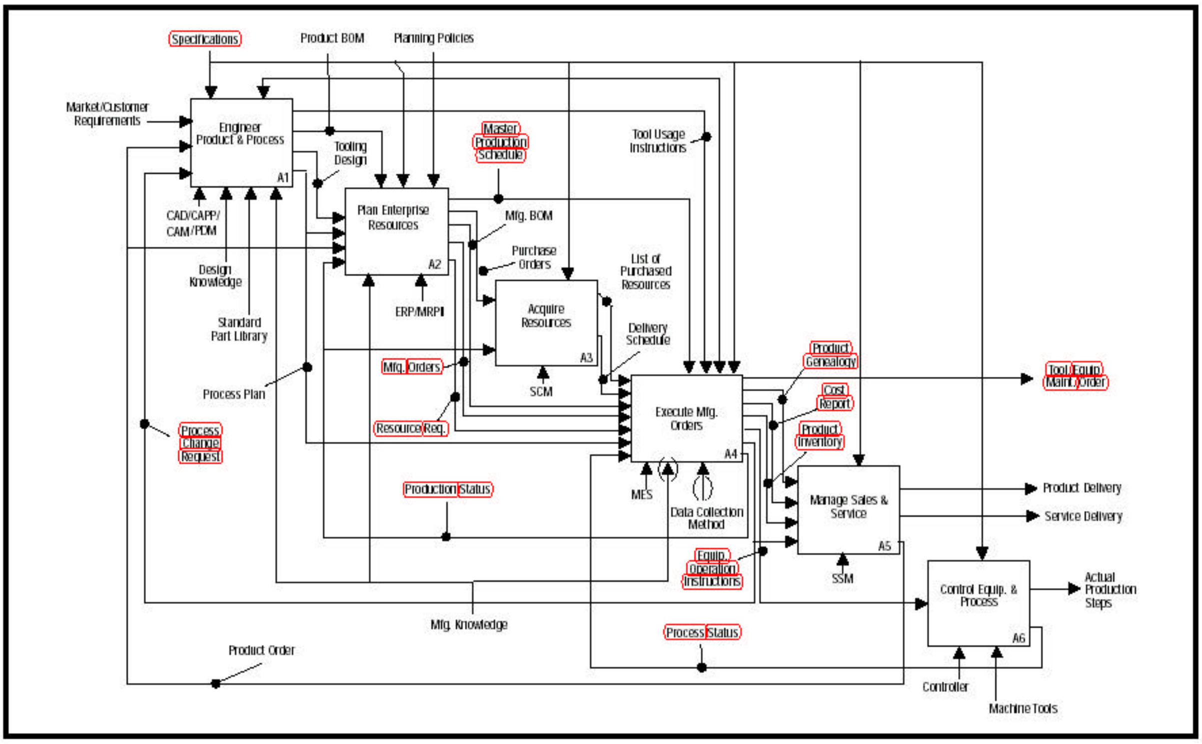


Figure 2.2: *Activity A0 – Develop Products*

The engineering group for manufacturing process engineering includes:

- process selection
- operation planning
- workpiece routing
- equipment/device control program generation

The A1 activity is important for providing design and processing information for the downstream resource planning and manufacturing execution. The decomposition of engineer process activity, which is equivalent to process planning, can be found in the NISTIR 58085 [3].

2.4.2. Enterprise Resource Planning

The Plan Enterprise Resources (A2) activity encompasses analyzing parts and performing make/buy decisions for all the parts. This involves developing a business plan and schedule to acquire the necessary resources and/or to produce products for the market. The resources include material, finished parts, equipment, and labor skills. The enterprise resource planning function includes:

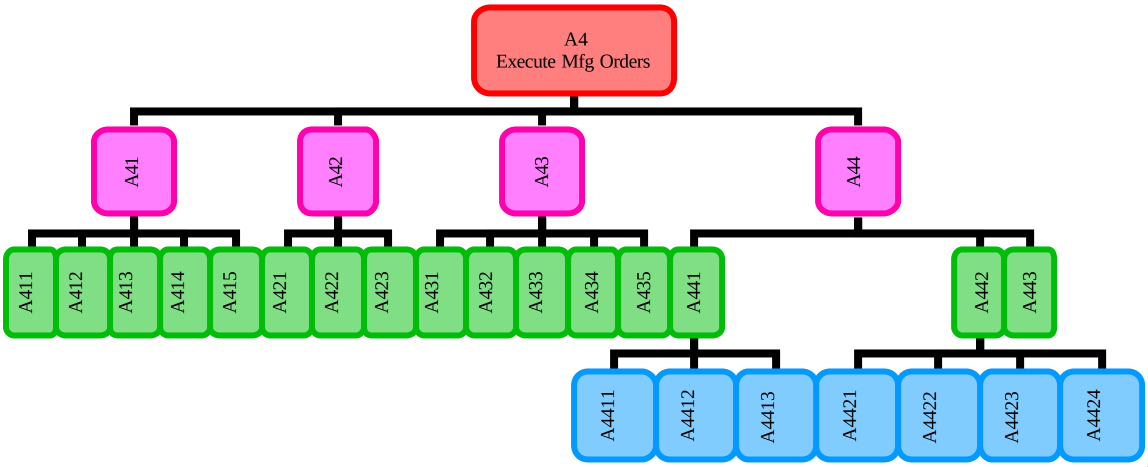
- financial and order management
- production and material planning
- *Master Production Scheduling*
- capacity requirement planning
- job definition: *Manufacturing Orders*
- business process planning
- resource requirement specification

2.4.3. Acquire Resources

Based on the resource requirement plan (*Purchase Orders*) specified in A2, the Acquire Resources (A3) activity purchases the resources from suppliers to meet the production schedule (*List of Purchased Resources*). This activity is supported by supply chain management, which includes distribution, logistics, transportation management (*Delivery Schedule*), and advanced planning.

2.4.4. Manufacturing Execution Model with Data Breakdown

The Execute Manufacturing Orders (A4) activity relates to the manufacturing execution functions and will be further decomposed to set the context for the development and use of the foundation interfaces for MES. Based on the production plan and master schedule, A4 is to initiate, guide, respond to, and report on production activities as they occur. The following is a hierarchy tree for all the sub-activities within A4.



The internal and interfacing data for the A4 block will be decomposed into the following category rating system:

Rating	Description
***	Frequent Data (timed in seconds, fixed sampling interval)
**	Moderate Data (timed in hours)
*	Long Term Data (timed in days/weeks)

This is done with the intent of:

1. Finding control points in the MES Module Interface Architecture, with the variables to optimize or use to optimize outputs.
2. Determining the data used for real-time information and feedback control.
3. Defining the fundamental functions, information for certain issues, in order to know what is taken for optimization, short and long term control.

In exploring these above issues, all the information data flow (input, output, control and mechanism) going into the MES modules is analyzed and broken down into what information is classified as real-time. Note, this is only a selected category of data and does not include all the data found in the MES Module Interface Architecture.

This rating system is a first stage breakdown of the real-time data. It still needs to be further analyzed to determine what should be taken for optimization in the short and long term control scheme.

The fundamental functions of each module are described along with definitions of all the categorized real-time data. Table 2.1 below defines this data for the A4 module. The control parameters that pertain to certain modules are also included in the tables. These are the signals sent through the top of the Module. The Mechanism is the method for handling the data, and is presented with an input through the bottom of the Module. Please note the color scheme is simply a way to clearly identify the different sub activity layers of the Module and is in accordance with the color scheme used in the hierarchy tree above.

Table 2.1: A4 Module Data

Mechanism			
Data Collection Methods: The use of data collectors to obtain information on workpieces, timing, personnel, lots, and other critical entities for production management in a timely manner. *Manufacturing Knowledge:* The information (rules, logic, examples) that a manufacturing engineer brings to bear on manufacturing engineering problems, including production techniques and implementation techniques. Many different types of manufacturing knowledge are used in different manufacturing activities, such as decomposition knowledge, assignment knowledge, consolidation knowledge, and optimization knowledge, which are used in process planning, resource planning, production planning, and scheduling. *Manufacturing Execution System (MES):* A production activity management system that initiates, guides, responds to, and reports on production activities on-line and in real time to production management people. The system aids the Execute Manufacturing Orders activity.			
Input			
***	*Process Status*	A report of the conditions of a process being monitored. The report includes alarms, process changes or shifts, workpiece throughput, and so on.	Inputted from: A6
* / **	*Master Production Schedule*	A plan that specifies starting time and finishing time of each job in the job queue that are for producing products required by customers. The plan contains job IDs, starting dates, and due dates.	Inputted from: A2
*	*Specifications*	Sets of description of standard engineering, manufacturing, and business practices that guide and control the product development process.	Inputted from: A6 (feedback)
*	*Resource Requirements*	A list of resources required supporting production jobs.	Inputted from: A2
* / **	*Manufacturing Orders*	Instructions that are sent to factories to start jobs to fulfill customer orders. The starting dates are specified in the manufacturing order according to the production plan and the master production schedule.	Inputted from: A2
Output			
**	*Production Status*	A report on the state of all scheduled operations and production units. This also includes the information on resources, process setup, job schedule,	Outputted to: A2 (feedback)

		and material routing.	
* / **	*Equipment Operation Instructions*	Specific operation steps or recipes that are used to control machine movement, such as machining, welding, assembly, material movement, and so on.	Outputted to: A6
**	*Process Change Request*	Feedback from factory-floor production requesting changes to process plan when some problems in the process plan were found. Changes can be process parameter changes, tool changes, setup changes, and so on.	Outputted to: A5, A1 (feedback)
**	*Product Inventory*	The inventory information on a product. The information is updated when finished products are sent to storage.	Outputted to: A5
*	*Cost Report*	A report on the manufacturing costs of producing a part. It contains the costs of material, labor, usage of equipment, and so on.	Outputted to: A5
***	*Product Genealogy*	One of the components in a MES that provides the visibility to where work is at all times and its disposition. Product genealogy information may include who worked on the product, components materials by supplier, lot, serial number, current production conditions, and any alarms, rework, or other exceptions related to the product. This information provides traceability of each part and component.	Outputted to: A5
* / **	*Tool, Equipment Maintenance Order*	An instruction indicating specific tools, machines, or devices that need maintenance before performing any production activities.	Outputted to:

The Manufacturing Execution A4 Module is decomposed into the following four sub-activities, as shown in *Figure 2.3*. The highlighted data found in *Figure 2.2* is re-highlighted hear in red, along with the internal data (in pink) that communicate between the A41, A42, A43 and A44 blocks.

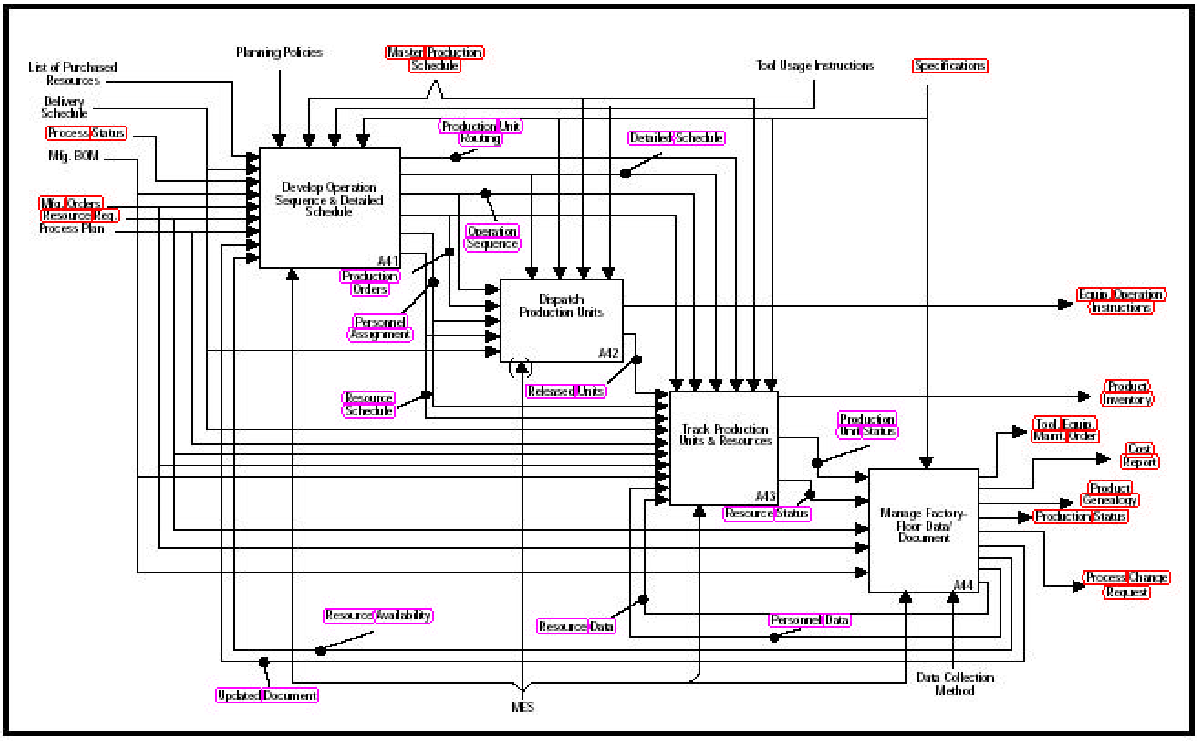


Figure 2.3: *Activity A4 – Execute Manufacturing Orders*

The sub modules are each defined, only the real-time data of the A41 module will be presented in tables, these are found in Appendix B. The A41 modules and interfacing data will be broken down in detail for the purpose of paving the way for the remainder of the thesis.

2.4.4.1. DEVELOP OPERATION SEQUENCE & DETAILED SCHEDULE

The A41 module is responsible for taking the *Production Plan* input data and the *Master Production Schedule* control data as inputs. Using this it defines, sequences, and schedules operations locally on the levels of workcells, workstations, and machines to optimize productivity. This activity considers parameters such as minimizing setup time, maximizing throughput, minimizing idle time, minimizing queue time, and adjusting shift patterns when a new priority is in effect.

The data associated with the A41 module is summarized in Table B.1 of the Appendix B. In it a detailed description of the data can be found along with the rating system assigned for the particular data signal.

2.4.4.1.1.DEVELOP OPERATIONS SEQUENCE AND DETAILED SCHEDULE FOR WORK STATIONS

In this segment of the system implementing the direction of the Work Order plan and in turn the logical configuration of the Work Stations for the day is dealt with. The planning, scheduling, and loading of each operational Work Station is done here, providing the current and total shop load by operation using routing data and time standards. Based on this plan, the system will request and manage delivery of inventory, tooling, and data in response to the *Manufacturing Bill of Material* requirements and will issue and execute commands to move the required items to the planned Work Station. The MES can and should include the direct control interface and connection with each Work Station (*Operation Sequence*).

The A41 is decomposed into five subactivities, as shown in *Figure 2.4.* If the data is carried from the above layers, then it will remain highlighted in the color defined previously. Here *Production Unit Definition* in green is the new data found internal to *Figure 2.4* below.

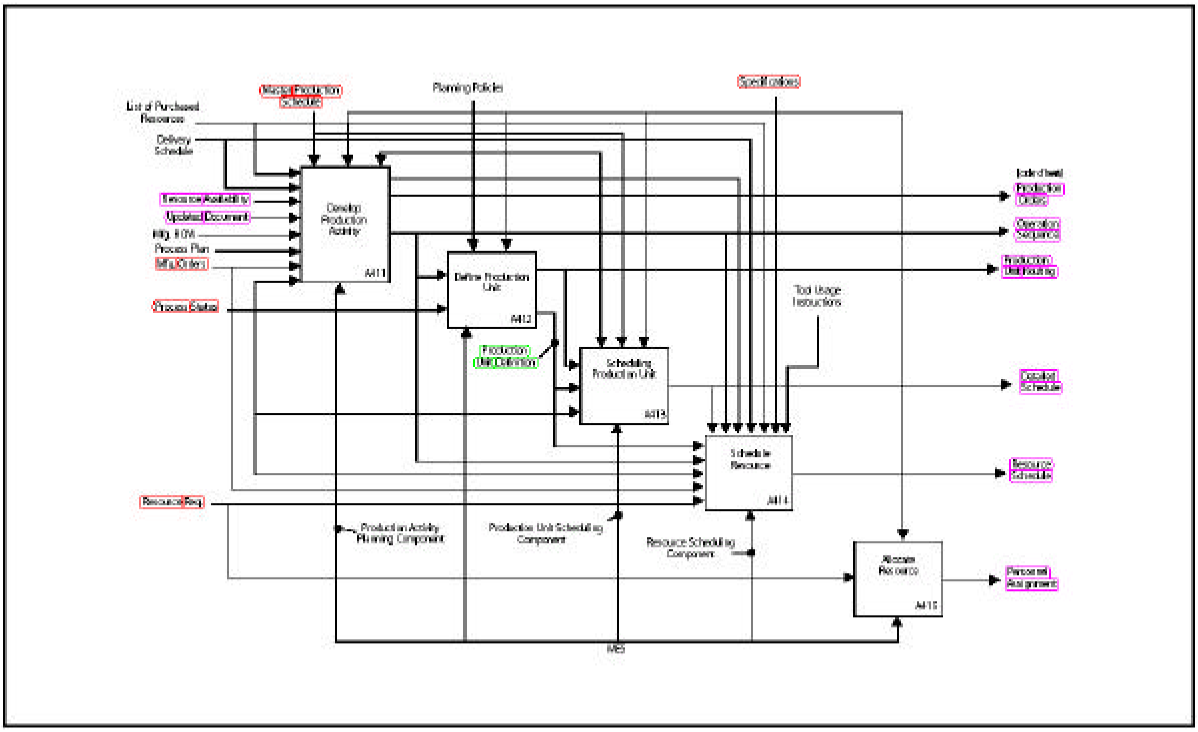


Figure 2.4: *Activity A41 – Develop Operation Sequence and Detailed Schedule*

The A411 Develop Production Activity module sequences production operations (*Operation Sequence*) based on the *Master Production Schedule*, priority (*Delivery Schedule*), setup changes/time (*Specifications*), and multi-objective optimization characteristics. It also reads as inputs *Production Status* and *Updated Documents* for the intent of generating alternative operation sequences to recognize possible production changes, such as priority changes, machine downtime, and other plant disturbances.

Table B.2 in the Appendix B provides a list of all the inputs and outputs as well as the rating system assigned for the data used in interfacing with the A411 module.

2.4.4.1.2.DEFINE PRODUCTION UNITS

The A412 module identifies a lot or batch (*Production Unit Definition*) by decomposing or collecting manufacturing orders. Each lot or batch is scheduled, processed, monitored, and tracked by the system as a unit. This is done by providing a plan that specifies the traveling route and processing times of the production unit (*Production Unit Routing*).

In Table B.3 of the Appendix B, a detailed description and rating system for the data associated with the A412 module can be found.

2.4.4.1.3.SCHEDULE PRODUCTION UNIT

The A413 activity module is concerned with adding the proper start and finish time information to a lot or batch dictated by the *Operation Sequence.* These start and stop times are provided in the outputted *Detailed Schedule.* The goal is to optimize productivity and quality and to conform to the *Master Production Schedule.*

The data associated with the A413 module is summarized in Table B.4 of the Appendix B. In it a detailed description of the data can be found along with the rating system assigned for the particular data signal.

2.4.4.1.4.SCHEDULE RESOURCE

The A414 module activity uses the start and finish time information on each resource that is used by operation(s) in the production and constantly updates its times based on the *Process Status*. The activity contains a plan for controlling resource availability and allocation in the event that a plant disturbance occurs. The activity has the ability to specify a reworked schedule based on the group of resources still available, this depends on what operations can be handled by what specific resources.

Table B.5 in Appendix B provides a list of all the inputs and outputs as well as the rating system assigned for the data used in interfacing with the A414 module.

2.4.4.1.5. ALLOCATE RESOURCE

The A415 module is responsible for assigning and preparing resources to operations that need the resource before they start. This entails properly setting up equipment. The activity also issues a *Personnel Assignment* that associates a type of resource and quantity of it to specific operation(s) that need(s) the resource for a specific time period.

In Table B.6 of Appendix B, a detailed description and rating system for the data associated with the A415 module can be found.

2.4.4.2. DISPATCH PRODUCTION UNITS

The A42 Module found in *Figure 2.3* above, determines which production unit in the queue is best processed next. The objective is to minimize the lead time and lateness. The module also contains instructions for specific equipment operation (*Equipment Operation Instructions*) steps used to control machine movement, assembly and material movement. This module also keeps track of the production units that are released (*Released Units*) for processing the manufacturing facility.

2.4.4.2.1.MATERIAL MOVEMENT

The movement of inventory or information to the needed location in the plant floor is another important area of MES. This portion of the system controls material movement in the plant, for either manual or automatic systems, by issuing requests for a manual move (printing move tickets) or issuing commands to material handling system control PLCs, such as ASRS, AGVS, conveyor systems, carrousels, robots, etc. The commands can be as simple as "Item no. must be moved from work station #1 to work station #2." Here *Equipment Operation Instructions* takes on this role.

The A42 is decomposed into three subactivities, as shown in *Figure 2.5*. Since this schematic is on the same hierarchy level as *Figure 2.4*, it also contains green highlighted data (*Resource Release Order* and *Task Assignments*).

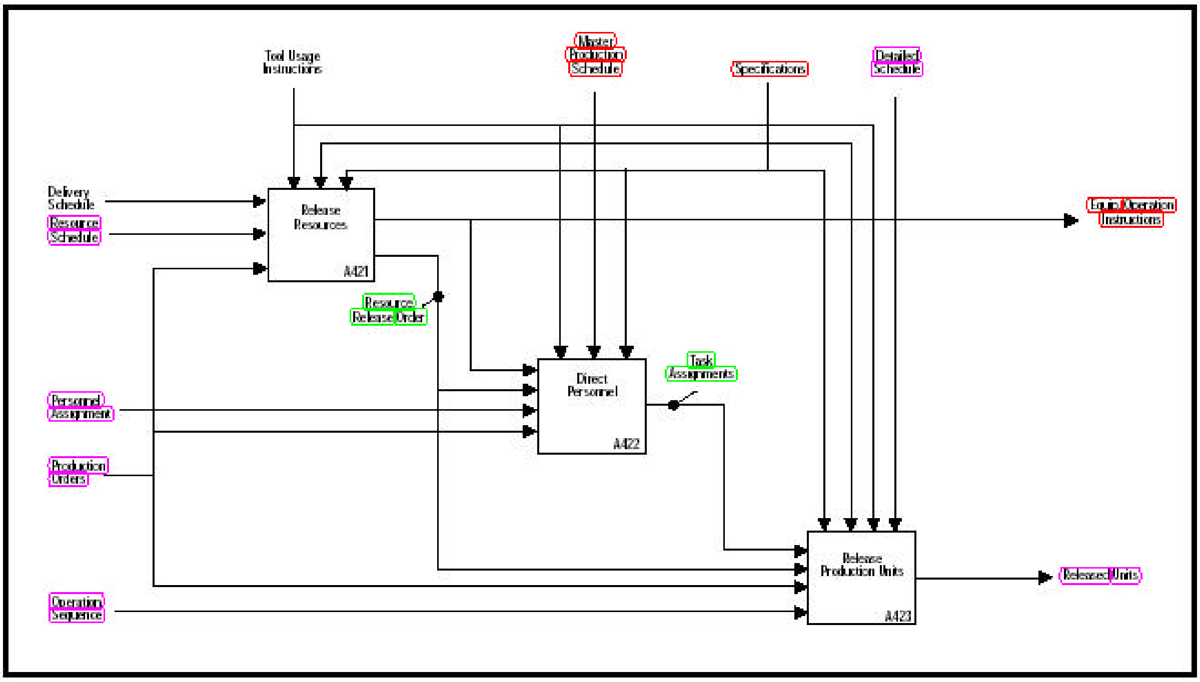


Figure 2.5: *Activity A42 – Dispatch Production Units*

The A421 Release Resources module is based on resource allocation and releasing resources (*Resource Release Order*) for the production activity in a timely manner. This activity results in physical materials, equipment and tools being moved from inventory or storage to specific production cells or workstations (*Equipment Operation Instructions*).

2.4.4.2.2.DIRECT PERSONNEL

The A422 module deals with assigning workers with adequate skills to perform specific operations according to the detailed schedule. This information of assigning tasks with due dates to workers is contained in the *Task Assignments* which is outputted by this module.

2.4.4.2.3.RELEASE PRODUCTION UNITS

Based on the *Detailed Schedule*, *Resource Release Order* and *Production Orders* the A423 module is responsible for *Releasing Units* to workcells or workstations and initiate processing the production units.

2.4.4.3. TRACK PRODUCTION UNITS AND RESOURCES

The product genealogical information is provided through the A43 module. Here the information on where any production unit is at all times (*Production Unit Status*) and its disposition is outputted to the A44 module, which outputs the *Product Genealogy* signal. The A43 module also provides information on who worked on it, current production information, component materials by supplier, lot number, serial number, any rework, measured data, or other exceptions related to the product. While at the same time providing the status information (*Resource Status*) on specified resources, such as tools, devices, machines, and stock materials, at all times.

2.4.4.3.1.INVENTORY TRACKING AND MANAGEMENT

Although the planning system has the collective data on inventory, the detail can easily reside at the local level, the MES. "Dock To Stock" operations are accomplished here with regular updates to the planning system. A current map of all inventory and storage locations, including WIP, can be maintained.

The A43 is decomposed into five subactivities, as shown in *Figure 2.6*. Here also the internal data for this schematic layer is highlighted in green.

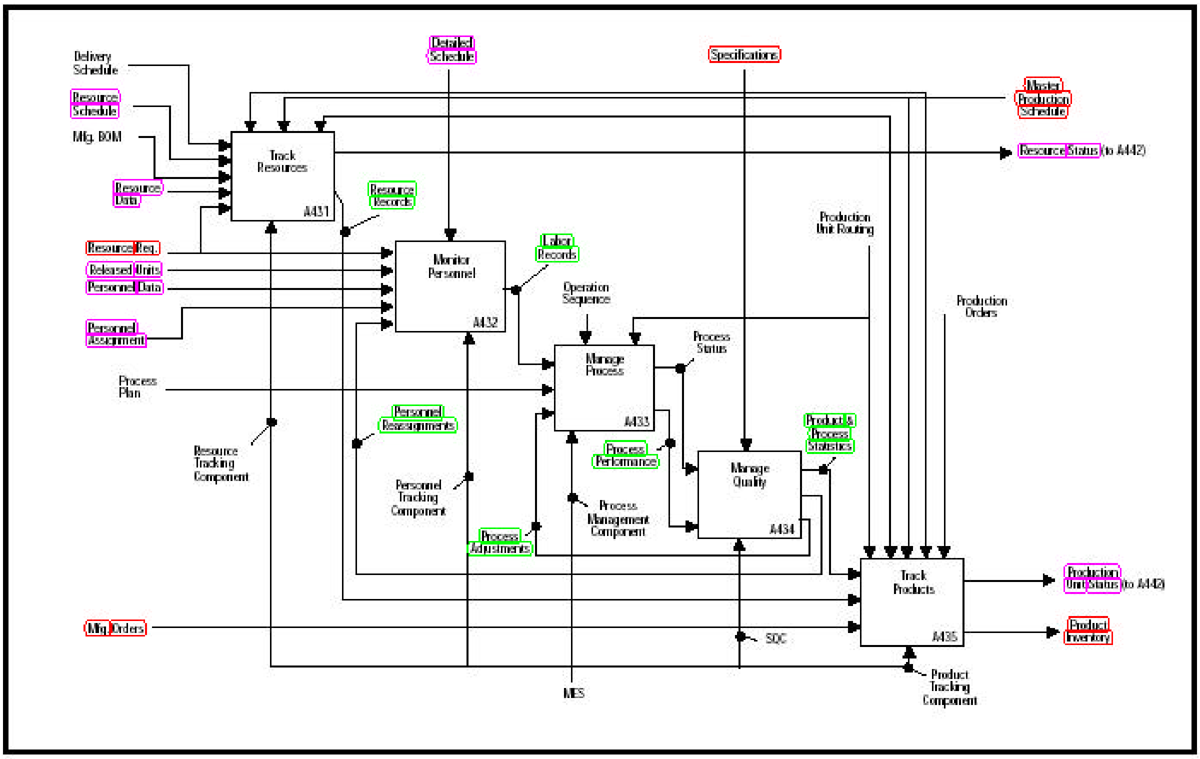


Figure 2.6: *Activity A43 – Track Production Units and Resources*

The A431 module monitors the status of resources. This is done using on-line tracking. The primary purpose is to track resources usability and consumption in the manufacturing facility. With this information the activity module creates a record of history for resources that are necessary to be traced (*Resource Records*). This piece of information indicates where the resource is located and who is using it for which operations on which production unit and for how long. If it is a piece of equipment, the record also shows whether it is functional.

2.4.4.3.2.MONITOR PERSONNEL

The A432 module tracks personnel status through *Personnel Assignment* and *Reassignments* and reports the status on the labor assigned. The report includes attendance, labor skill changes, job assignments, time performed on each assignment, and material/tool preparation time.

2.4.4.3.3.MANAGE PROCESS

The role of the A433 module is to monitor a production process and make timely decisions to adjust the detailed schedule and process plan when unexpected situations occur. These activities are requests from operators to process planners to improve process performance. Process management includes alarm management to make sure factory personnel are aware of process changes that are outside acceptable tolerances (*Process Performance*). It also includes process setup and tool preparation before production units are dispatched for processing. The activity module also maintains a history of past events or problems to aide in diagnosing problems.

2.4.4.3.4.MANAGE QUALITY

The A434 module provides timely analysis of measurements collected from products and processes to control product quality. This is done using the *Statistical Quality Control* (*SQC*) Mechanism Data. It checks the current production rate with the detailed production schedule. Identifies problems in production requiring attention, and gives recommended proper actions to correct the problems. The activity module also provides measurements and statistical analyses of process performance and product quality visible to production/business management personnel (*Product/Process Statistics*).

2.4.4.3.5.TRACK PRODUCTS

The A435 module monitors the progress of production using the *Product Tracking Component* Mechanism data and *Resource Records* Input data. With this it provides up-to-the-minute reporting on the production status (*Production Unit Status*), such as the quantity of a product made, scrap rate, rework rate, and a comparison to the production schedule.

2.4.4.4. MANAGE FACTORY-FLOOR DATA/DOCUMENT

The role of the A44 module is to output the *Production Status* mission-critical data pertinent to production activities. This is done through linking hardware/software interfaces to the machines. With this interface data can be collected from the factory and analyzed for multiple purposes, such as product throughput, quality, delivery, and equipment maintenance. The management of documents, such as cost reports, maintenance orders, inventory reports, *Process Change Request*, manuals, specifications, company policies, and so on is also handled in this module.

This activity module also controls the data collection, access, and distribution to the appropriate places. While providing versioning control of documents, such as part programs, operation instructions, *Manufacturing Orders*, *Detailed Schedules*, part drawings, *Process Change Request*, production unit records, records of communication from shift to shift, manuals, standards, company policies, safety regulations, all of which are *Updated Documents*.

The amount of information necessary to manage manufacturing grows with production, and the idea of written information and manual data entry or manual delivery of information (to and from the plant floor) is too restraining. Therefore this part of the system acts as the clearinghouse and translator for all information that is needed and/or generated on the plant floor. Present *Data Collector*, *Data Analyzer* and *Document Controller* systems are excellent tools that provide immediate and accurate information wherever needed.

The A44 is decomposed into three subactivities, as shown in *Figure 2.7*:

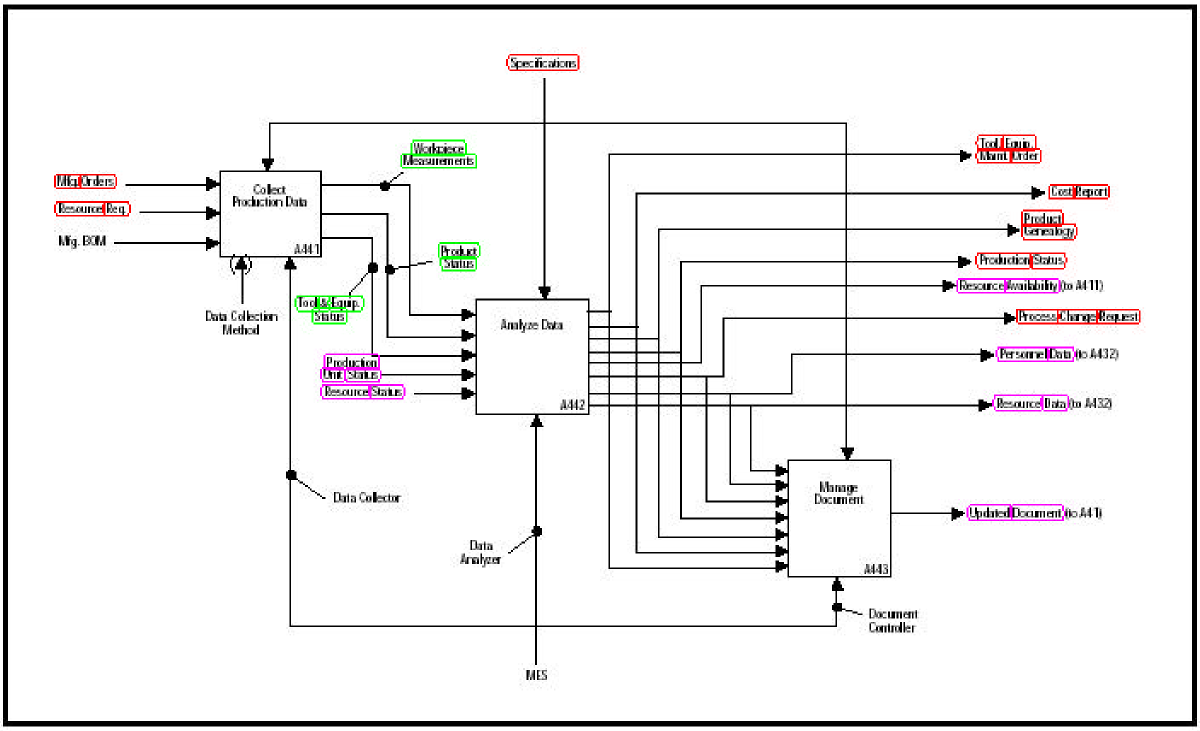


Figure 2.7: *Activity A44 – Manage Factory Floor*

2.4.4.4.1.COLLECT PRODUCT DATA

The A441 Collect Production Data Module uses data collection devices (Mechanism Data), to acquire data by measuring and sampling workpieces (*Workpiece Measurements*), products (*Product Status*), and production processes (*Tool and Equipment Status*) to support the management of product, quality, and process.

Gathering information so the system can remain current is done by this section of the MES system and serves as the eyes and ears for management. This is done with the use of various kinds of sensing devices and control interfaces. Data from the floor operations can be collected, collated, and dispersed on whatever basis is desired. This is the primary method for the MES to read from the plant. The information can be input by system operators or recognized by events electronically via direct connection through SCADA or PLCs.

The A441 is further decomposed into three subactivities, as shown in *Figure 2.8*. This takes the hierarchy to its lowest level for MES. There is no new data introduced in this layer for this schematic.

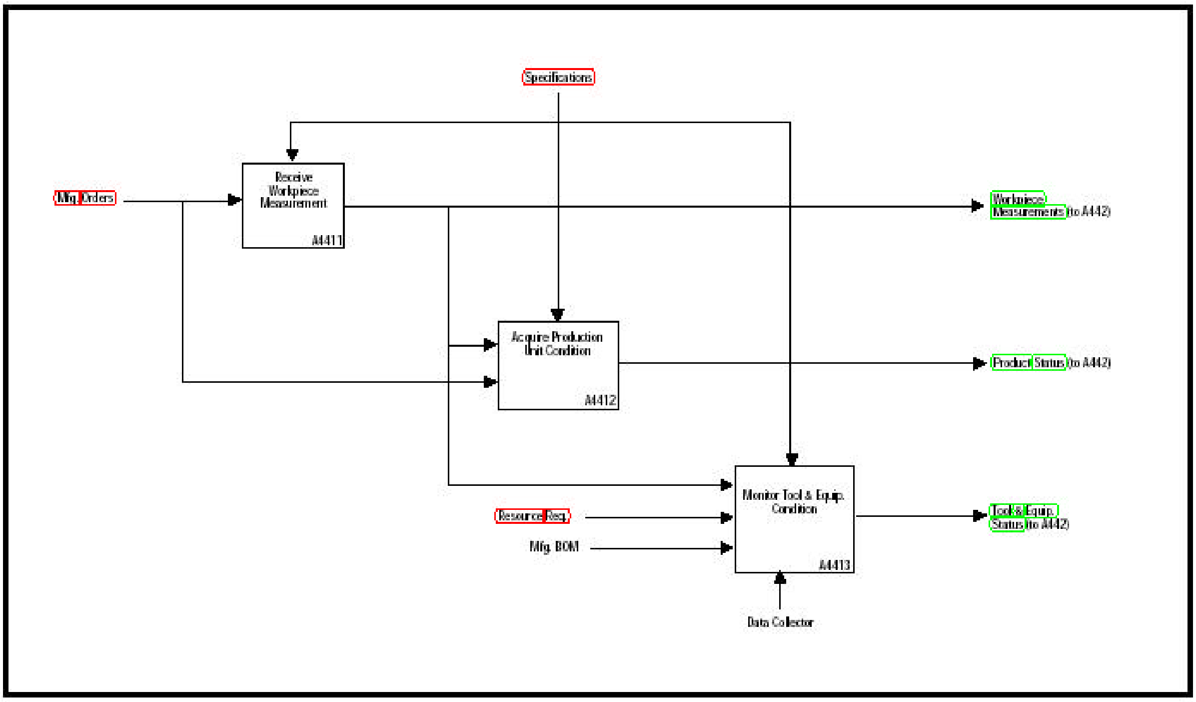


Figure 2.8: *Activity A441 – Collect Product Data*

2.4.4.4.1.1. Receive Workpiece Measurement

Using factory-floor data collection devices the A4411 module, acquires and collects measurements on workpieces (*Workpiece Measurements*), labor records, and process conditions in order to monitor the process performance and product quality.

2.4.4.4.1.2. Acquire Production Unit Condition

The A4412 Module uses data collection mechanisms, such as bar code readers or manual input devices, to acquire data on the production units. This is used to determine where the production units are and how many units of a product have been finished. The module outputs the *Product Status* information on-line and up-to-the minute. This data is made available and visible to production as well as business management.

2.4.4.4.1.3. Monitor Equipment Condition

Using the data measured from product and process, the A4413 module indicates the status of tools, devices, and machines being used in production (*Tool and Equipment Status*) to determine whether they are still proper to function or need adjustment or maintenance.

2.4.4.4.2.ANALYZE DATA

Using collected data and adequate algorithms the A442 module, analyzes the data using the *Data Analyzer* which provides up-to-the-minute reports of actual manufacturing operations results along with the comparison to past history and business expectations. With this the activity module generates records and reports and makes them available for decision making and product tracking.

The quality control method is done through the *Statistical Process Control* (*SPC*) and *Product Genealogy* by focusing on continuous monitoring of a process rather than the inspection of finished products. Achieving control of the process and eliminating defective products is the role of the *SPC*; it includes a collection of tools (*Data Analyzer*) mostly statistical, which help to understand what is going on in desired processes. The *Product Genealogy Component* keeps a record of the history of a product from its introduction into the production process through to its termination.

The A442 is also further decomposed into four subactivities, as shown in *Figure 2.9*. This schematic fits along side *Figure 2.8* in the lowest level of the MES hierarchy. The new data introduced in this level (*Quality*, *Resource* and *Production Statistics*) are highlighted blue.

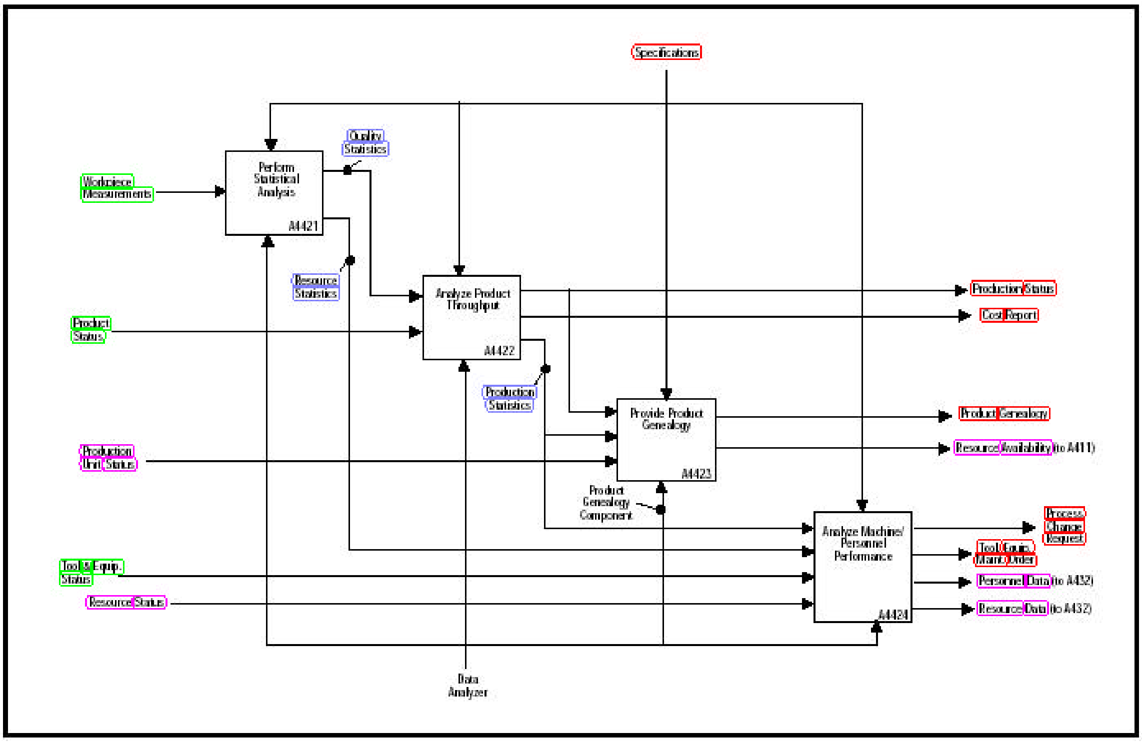


Figure 2.9: *Activity A442 – Analyze Data*

2.4.4.4.2.1. Perform Statistical Analysis

The role of the A4421 module is to conduct statistical analysis on data collected from the shop floor for tracking process performance and ensuring product quality. The outputted statistical data *Quality Statistics* and *Resource Statistics* are pertinent to the quality of the product and the state of the resources inspected respectively.

2.4.4.4.2.2. Analyze Product Throughput

The A4422 module calculates the quantity of a product completed. Then checks it against the schedule, and makes it available to authorized personnel to view. This is the outputted *Production Statistics* signal, which also provides measurements and statistical analyses of the production process quality.

2.4.4.4.2.3. Provide Product Genealogy

The A4423 module creates records on the product for traceability, including operations, process parameters, lot number, batch number, supplier, operator identifications, product measurements, and any exceptional processing conditions occurred. This data which provides visibility to where work is at all times and its disposition is the *Product Genealogy*. The module makes this record available to the authorized personnel.

2.4.4.4.2.4. Analyze Machine/Personnel Performance

Using the collected process data, the A4424 module analyzes the machine usage, production rate, capability, and estimates the maintenance schedule (*Tool, Equipment Maintenance Order*). The module also analyzes worker's performance, such as productivity, labor skill, and attendance record (*Personnel Data*). As well as indicating the condition of a resource based on inspection or measurement analysis (*Resource Data*).

2.4.4.4.3.MANAGE DOCUMENT

Collect (or generate), maintain, and distribute production-related documents and records to support the production, factory-floor decision making, and product traceability is the main function of the A443 module. This is done using the *Document Controller* mechanism data. This software, controls records and forms that support product life cycle activities, such as manuals, drawings, computer models, procedures, recipes, programs, engineering change or ders (ECO), shift-to-shift communication records, and so on. This document management activity module takes in many different data signals outputted from the A442 module and provides the appropriate information into the *Updated Document.*

2.4.5. Manage Sales and Services

The Manage Sales and Services (A5) activity consists of managing the sales of products and services to customers. This includes:

- product delivery and receipts
- product configuration
- customer orders
- quotes to customers
- product returns
- post sale service

2.4.6. Control Equipment and Process

Control Equipment and Process (A6) activity relates to the use of preprogrammed instructions (*Equipment Operation Instructions*) to control equipment motions and processes in real time. Activity A6 usually involves distributed numerical control, programmable logic control, and factory floor data collection and monitor (*Process Status*).

Chapter 3

3. MES Module Melt Shop Case Study

3.1. INTRODUCTION

The MES modules described in the previous chapter will be decomposed to illustrate a small example of its application. The case study will first be explained followed by the working model, and finally the specifics of the MES modules that will be used for interfacing with the plant will be shown based on the architecture of Chapter 2.

One question that is dealt with frequently by manufacturing management is "How is the work schedule going to be performed?" This question may seem obvious or intuitive, though the area of schedule development probably offers the best opportunity for improving the resource management process [12]. Determining Production quantities is not the concern here (most probably the manufacturing orders were done by the ERP), but instead how to prioritize a given list of jobs based on the resources (equipment resources, operators and inventory) currently available.

This Chapter will present such a scheduling tool for the following specific manufacturing plant model.

3.2. MELT SHOP MODEL

3.2.1. Short Background

In a melt-shop steel production consists of processing a heat of steel into billets. A heat of steel is the order placed by the customer and is defined by a specific grade and a specific weight. The production process involves many steps, each step taking place on a different machine unit. The processing time associated with each machine unit is dependent on the grade of steel being processed and in some cases the available power consumption allowed.

Producing billets of several different grades of steel is typical for a production order. Only one grade of steel can be processed at a time and since cost is associated with the order of production (Section 3.2.2 will expand on this), the processing order of each grade of steel must be scheduled for the melt-shop.

Many efforts have centered on obtaining optimal schedules given the logistical and economic importance of scheduling problems. Unfortunately, it has been proved that most scheduling problems of interest belong to the class of NP hard combinatorial problems, and the computational requirements for obtaining an optimal solution grow exponentially as the problem size (e.g., number of jobs and/or number of machines) increases [11]. These pure optimization methods are therefore impractical for industrial applications. For this reason we look at ways of handling optimal scheduling. This is defined as one that maximizes profit, and minimizes production time. The optimal scheduling of events within a steel production process becomes computationally more complex as the number of machines and jobs within the process increases. For this reason, an enumerative method for determining an optimal solution is not desirable. This raises a need for a method that can efficiently handle a multi-objective problem, as minimum cost and makespan are both required, and still produce an optimal schedule within a reasonable computational time. A makespan is defined as the total time needed to produce all the heats in the schedule.

3.2.2. Problem Definition:

Here we consider a simple melt shop. This consists of processing up to 6 heats of steel in one full day (12 hour shift). Each heat can be of a different grade and weight. Table 3.1 below gives details of one such production order.

Table 3.1: Heat Orders

Heat	Steel Grade	# of Billets	Billet Weight (Tons/Billet)
1	A	5	25
2	B	6	20
3	C	6	20
4	D	5	25
5	E	6	20
6	F	6	20

The production of all six heats of steel must occur in such a way, as to minimize makespan, and the amount of downgraded transition steel produced. The machine and containers involved in steel production include electric arc furnaces (EAF), ladle furnaces (LF), ladles, a tundish, and a caster. *Figure 3.1* below, provides an illustration of the machines and containers involved in the production process.

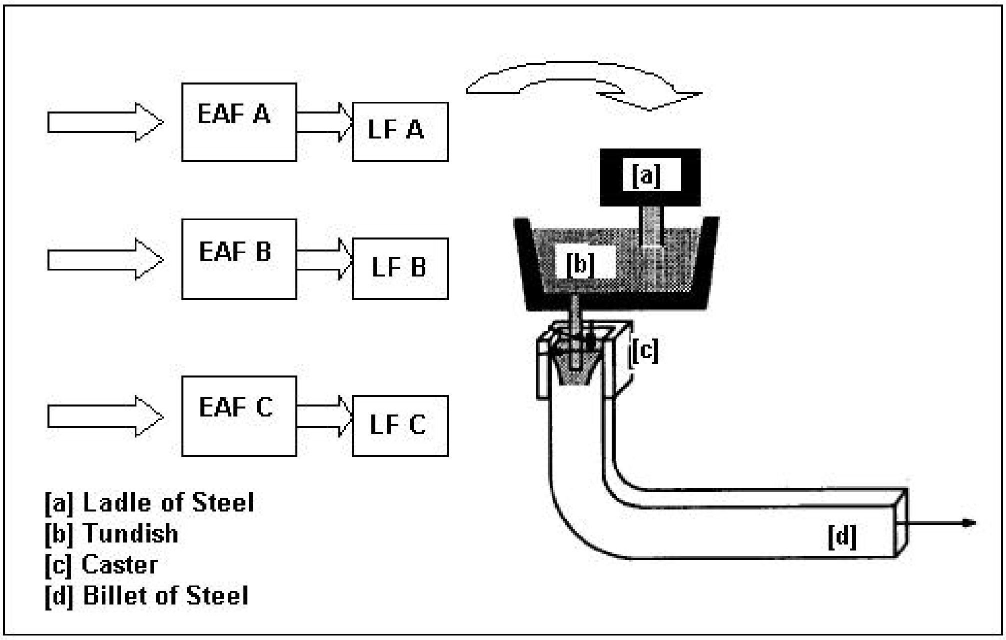


Figure 3.1: *Melt Shop Machines*

Here the plant has 3 EAFs and 3 LFs. The production scheduling of the heats should be able to handle a Disturbance Type 1: Machine Down. The furnaces are used for heating the steel to its liquid phase, and the ladles hold the molten steel. The tundish is a holding container into which the molten steel is poured. Remnants of the previous heat of steel always remain in the tundish. When a new order of steel is poured into the tundish, it is assumed that one billet of transition steel is created, before pure grade billets are produced. This transition steel has a cost for production, since the mixed grade billet might only be sold for an equal or lesser price, or might be scrapped. Note, producing a mixed billet and scrapping it has a cost associated with the heating and time on continuous caster lost, since the outputted billet is not sold. Table 3.2 below gives the cost associated with this mixed grade billet.

Table 3.2: Cost Associated with Transition Steel

From/To	Grade A	Grade B	Grade C	Grade D	Grade E	Grade F
Grade A	0	$200/ton	$100/ton	0	$200/ton	$100/ton
Grade B	0	0	0	0	0	0
Grade C	0	$100/ton	0	0	$100/ton	0
Grade D	0	$200/ton	$100/ton	0	$200/ton	$100/ton
Grade E	0	0	0	0	0	0
Grade F	0	$100/ton	0	0	$100/ton	0

Once the steel is poured into the tundish, it flows out of the tundish and into the caster. The caster is a mold, which creates long strands of steel. The steel will later be cut into billets. Some of the machines in this production process have specific characteristics, which are independent of the grade of steel they are processing. These characteristics are outlined in Table 3.3 below. It is assumed that EAFB and EAFC have the same capacity and are larger than EAFA. This must be the case, since if there is a Disturbance Type 1: Machine Down, there must be one machine left with a large capacity or else some of the heats in the schedule can not be made.

Table 3.3: Machine Characteristics

Machine	Characteristic
EAF A	Capacity = 120 tons Heating time + time to transport ladle to LF = 50 minutes
EAF B	Capacity = 130 tons Heating time + time to transport ladle to LF = 60 minutes
EAF C	Capacity = 130 tons Heating time + time to transport ladle to LF = 60 minutes
CASTER	If Caster is stopped once started, there is a delay before it can be restarted. Turnaround Delay = 20 minutes

The specific heat grade affects the time to produce billets of steel on each of the machines. These grade-specific jobs are outline in Table 3.4 below. Note in the above table, a larger capacity EAF takes longer to heat. These values are based on a standard power running day.

Steel plants suffer from an unusual disturbance, generated externally. Being such a high user of electricity, they pay dearly if they run the plant at times of peak demand. They therefore have agreements with the electricity generator, generally that they would switch off at times of known high demand. Therefore each day, the generator forecasts these peaks and the plant decides whether it is economic to close down or run only during certain hours. If a Disturbance Type 2 occurs, the power consumption levels must be changed. This has the effect of changing the heat times for the EAFs and LFs. The heat_time_table.m file stores the values to use for the plant. Presently, they are set to these default values. Note, the Continuous Caster times are fixed, since they are based on the physical fluid flowing properties of the heat being poured.

Table 3.4: Jobs Involved in Steel Production

Job	Machine	Time Needed Grade A	Time Needed Grade B	Time Needed Grade C	Time Needed Grade D	Time Needed Grade E	Time Needed Grade F
1st Heating of Steel and Transport to LF	EAFA/ EAF B/ EAF C	not dependent on grade, see Table 3.3 above					
2nd Heating of Steel and Transport to Tundish	LF A/ LF B/ LF C	20 min	30 min	20 min	20 min	30 min	20 min
Casting the Steel	Caster	2 tons/min	1.5 tons/min	2 tons/min	2 tons/min	1.5 tons/min	2 tons/min

The above problem description is a simplified version of a melt-shop steel production process. Several assumptions were made about the functioning of the system, in order to simplify the problem. These assumptions include:

1. The dynamics of the steel flow, in and out of the tundish, is simplified to Flow In = Flow Out. This is consistent with the use of a tundish level controller. No time has been associated with this process, and the continuous time behaviour of the tundish has been neglected.
2. The post-casting jobs of cutting and cooling of a billet of steel are not included as jobs in the melt shop production process.
3. The model will consist of 3 EAF/LF modules. Therefore if 1 goes down, the other 2 can still handle a complicated scheduling task (disturbance type 1). A machine being down for maintenance is not a disturbance type 1, therefore, the day schedule can be made knowing in advance what machines are available.
4. LFA is coupled to EAFA, same goes for B and C. Originally it was though that if an EAF or LF goes down, then the EAF or LF left alone could still be used along with the other 2 EAF's and LF's. However this does not improve the scheduling time in any way, since the times of the EAF and CC are much larger (double) the LF times. Therefore, if an EAF is down so is its LF and vice versa.

5. EAF/LF's may be used in parallel, if consecutive heats are scheduled on different EAF's
6. The melting times of the EAF's and LF's can be changed based on the power level to be used on the machines (disturbance type 2). A disturbance type 2 is assumed to take place prior to the day schedule since a properly operating MES will know when the power levels will affect the melting times.
7. There is no constraint on the amount of time the LF can hold the molten steel while waiting for the CC.

3.3. SYSTEM MODEL

A Discrete Event System (DES) model is designed and built to help obtain a better understanding of the interaction between the resources (machines) and entities (heats of steel), within the melt shop. The DES model was made in STATEFLOW, which is housed in SIMULINK. The SIMULINK model contains a system clock, and start switch. It also allows the GA's outputted schedule to interface nicely with the SIMULINK model, via the common workspace of MATLAB. Each STATEFLOW module consists of a series of interacting finite state machines. STATEFLOW is particularly suited to modeling discrete event systems, as events can take place within a state, and on the transition between states.

From an abstract perspective, the model consists of entities continually requesting resources, using resources, and freeing resources. The resources are either processing the entity or waiting for an entity to request their use.

The model is used to represent the DES steel plant, and test the results of a Genetic Algorithm (GA). It is also used to introduce random variables into the steel production process, and helps determine their impact on system performance. The results of the system model simulation can be found in Hadjimichael [7].

3.3.1. Overview of the System Model

The SIMULINK model of the melt-shop is shown in *Figure 3.2* below. The main things to notice are the 5 blocks labeled INPUT QUEUE, EAF-LFA, EAF-LFB, EAF-LFC, and CASTER. Internally within each block is a number of interacting STATEFLOW modules, representing the block's behavior. The wires connecting the blocks carry the request and grant handshaking signals as well as the characteristics of the heats of steel. The 4 input boxes labeled ELA_STATUS, ELB_STATUS, ELC_STATUS and SCHEDULE provide the interface from MATLAB's workspace into the SIMULINK model.

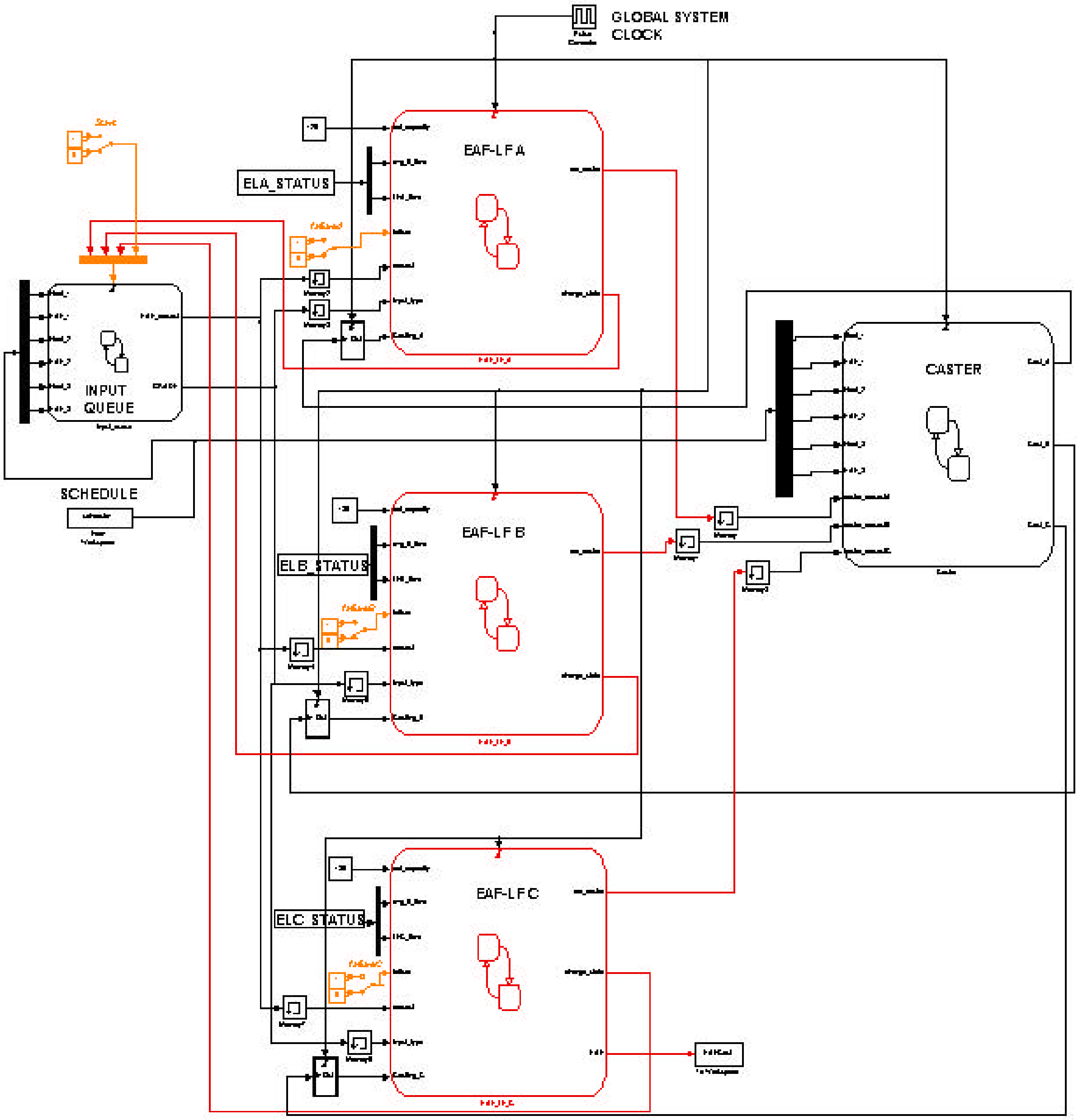


***Figure 3.2**: SIMULINK Model of the Melt-shop*

3.3.2. Input Module

The INPUT QUEUE module is used to model the order for the schedule of heats of steel. Each state in the INPUT QUEUE represents a heat in the schedule, and a final state represents that all heats have entered the production process. The inputs to each state are the grade of steel, and the starting furnace of each heat in the schedule. The outputs of each state are a request for an EAF, and the grade of steel requesting the EAF. A state transition occurs when the EAF requested by a heat, is servicing that heat ("change state" signal sent to INPUT QUEUE by appropriate EAF-LF). The SCHEDULE block in *Figure 3.2* above takes the schedule outputted from the GA and via the workspace loads it into the INPUT QUEUE module.

The key functions of the INPUT QUEUE are illustrated in *Figure 3.3* below. For a more detailed description and diagram refer to Appendix C.

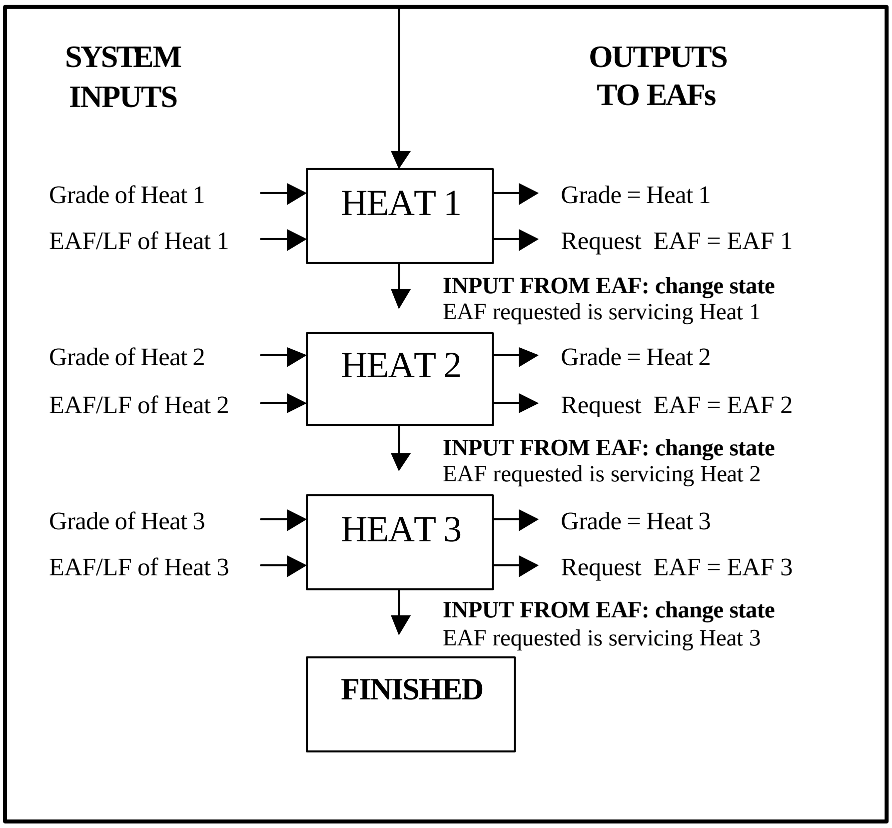


Figure 3.3*: Input Queue Model*

3.3.3. Resource Modules

The resources of the system consist of the EAF-LFA, EAF-LFB, EAF-LFC and CASTER blocks shown in *Figure 3.2* above. They all contain internal timers, which are usually dependent on the grade of steel they are servicing. Note the variable 't' used in the MATLAB workspace is the time symbol. This means the variable 't' in the workspace is directly coupled to the SIMULINK simulation target time. Generally, all the resources are either processing steel, or waiting for a heat of steel to process. The stuck in down state due to Disturbance Type 1 is also a possible state for EAF-LFA, EAF-LFB and EAF-LFC. The Caster can fail only by the Turnaround Delay.

3.3.3.1. THE EAF MODULE

The blocks EAF-LFA, EAF-LFB and EAF-LFC consist of an EAF coupled with an LF. All three blocks are essentially the same. The unique capacities and processing times is what categorizes the characteristics of the three blocks. Since these values are inputted in to each block, the internal workings of all three blocks are identical. The EAF-LF blocks also receive as inputs, requests from heats of steel from the INPUT QUEUE, and the grade of the requesting heat. The EAF-LF blocks output a caster request, and the grade of the heat requesting the caster. The EAF-LF blocks contain three finite state machines, an EAF, and EAF timer, and an LF timer. All three state machines interact with each other in parallel.

The EAF's have been modeled by two main states, IDLE and BUSY. The EAF stays in the IDLE state, until a heat requests its use. As the EAF enters the BUSY state, it triggers the EAF timer to begin counting. It also sends a "request being processed" signal to the heat (in the INPUT QUEUE), which requested it. When this signal is sent the heat of steel is being processed by the EAF. The EAF will enter the IDLE state again, after the EAF timer indicates the processing time has finished, and once the LF begins to process the heat of steel. The EAF will then be able to service another heat of steel.

The EAF timer has two main states, DISABLED and COUNTING. The timer represents the amount of time required to process a given heat of steel. Once the timer has finished counting (simulated minutes are inputted from ELA_STATUS and are heat dependent), it will send a signal to the LF timer, stating a heat is ready to be heated in LF. It will then enter the DISABLED state again. Note ELA_STATUS is for EAF-LFA, similarly ELB_STATUS is for EAF-LFB and ELC_STATUS for EAF-LFC.

The EAF and associated timer are illustrated with the key features below in *Figure 3.4*. For a more detailed description and diagram refer to Appendix D.

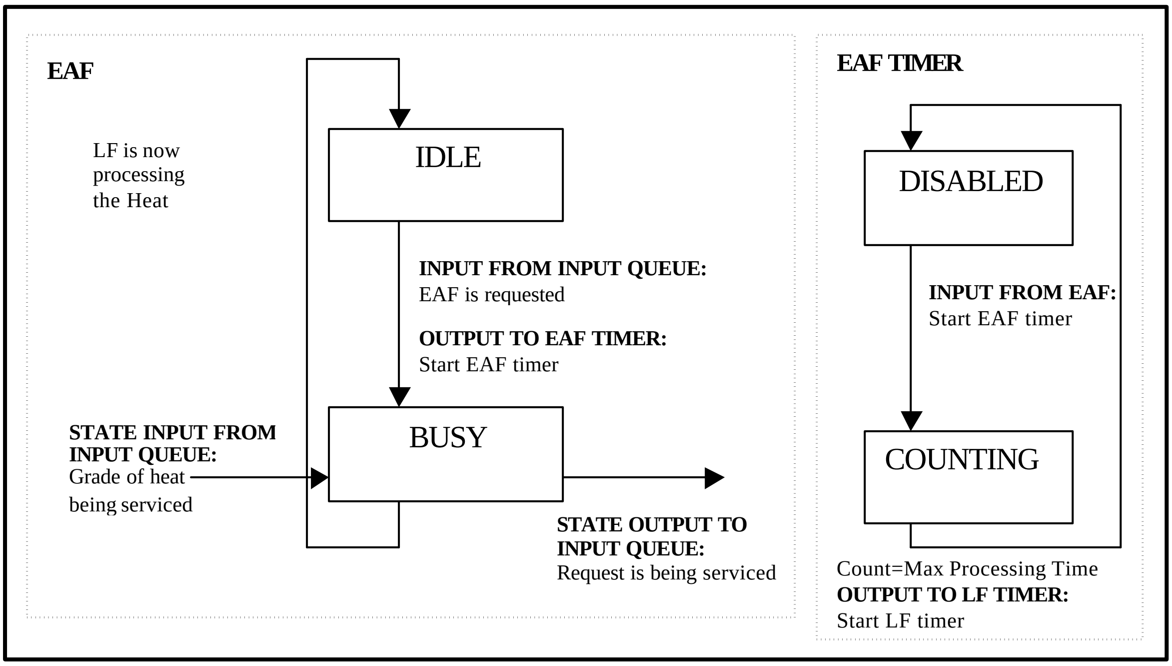


Figure 3.4: *Model of EAF and EAF Timer*

3.3.3.2. THE LF MODULE

The LF has been modeled as a timer. Since it is directly coupled to an EAF, it is always used once the EAF has finished processing the heat of steel, and does not have to be requested. The LF timer consists of three main states, DISABLED, COUNTING, and CASTER_WAIT. Upon completion of the EAF processing of the heat, the EAF timer signals to the LF timer that a heat is ready for it to process. This causes the LF timer to move into the COUNTING state. The EAF passes the grade of the heat to the LF timer. The LF timer will stay in the COUNTING state, until the grade-specific processing time has expired (inputted from ELA_STATUS for EAF-LF A, similarly for B and C). It will then send a "caster request" signal to the caster, and move to the CASTER_WAIT state. The "caster request" signal carries with it, the grade of steel requesting the caster. The LF timer will stay in the CASTER_WAIT state, until the caster is available to service the heat of steel. Once a "processing request" signal has been received from the caster, the LF timer returns to the DISABLED state, and can process another heat of steel.

The LF timer is illustrated with the key features below in *Figure 3.5*. For a more detailed description and diagram refer to Appendix D.

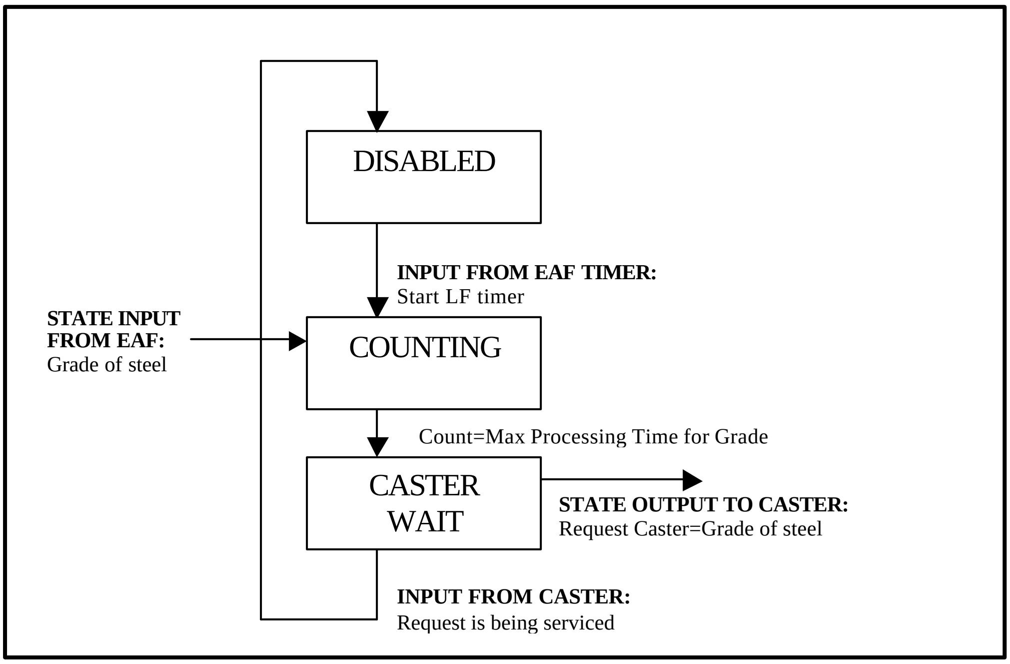


Figure 3.5: *Model of LF Timer*

3.3.3.3. THE CASTER MODULE

The Caster block consists of three interacting finite state machines; the Caster, Caster Timer, and Turnaround Timer. The Caster receives as input, the schedule of heats, so it will know which resource request it should service first. The "caster request" input signal carries information about the grade of steel requesting the caster. The Caster is broken down into four states; IDLE, BUSY, TURNAROUND and DONE. When a request to use the caster is received from an LF, the Caster enters the BUSY state, and signals to the Caster Timer, to begin counting. Upon completion of the casting time, the Caster Timer signals to the Caster that the casting is complete. The Caster checks again to see if there is a new request from an LF. If there is, it briefly enters the IDLE state, and then returns to the BUSY state again. If there is not another request, this indicates the Caster has to stop casting, and therefore, enter the TURNAROUND state. During the transition to the TURNAROUND state the Caster signals to the Turnaround Timer to start counting. If there are no more requests to service (this occurs when the schedule is at half day or finished completely), the Caster enters the DONE state.

The Caster Timer consists of two states; DISABLED and COUNTING. The timer represents the amount of time needed for casting the given heat of steel. The Caster signals when to enter the COUNTING state. Once the timer has finished counting, it will signal to the Caster, that the heat has finished being cast, and it will then enter the DISABLED state again. Note the Caster Timers are fixed values that depend on the type of heat.

The Turnaround Timer also consists of two states; DISABLED and COUNTING. The timer represents the amount of time needed to turn the caster around. The Caster signals a "turnaround" signal when the Turnaround Timer should enter the COUNTING state. Once the timer has finished counting (20 minutes), it will signal to the Caster that the Caster has been turned around. After this, the Turnaround Timer will re-enter the DISABLED state.

The Caster, Caster Timer and Turnaround Timer are illustrated with the key features below in *Figure 3.6*. For a more detailed description and diagram refer to Appendix E.

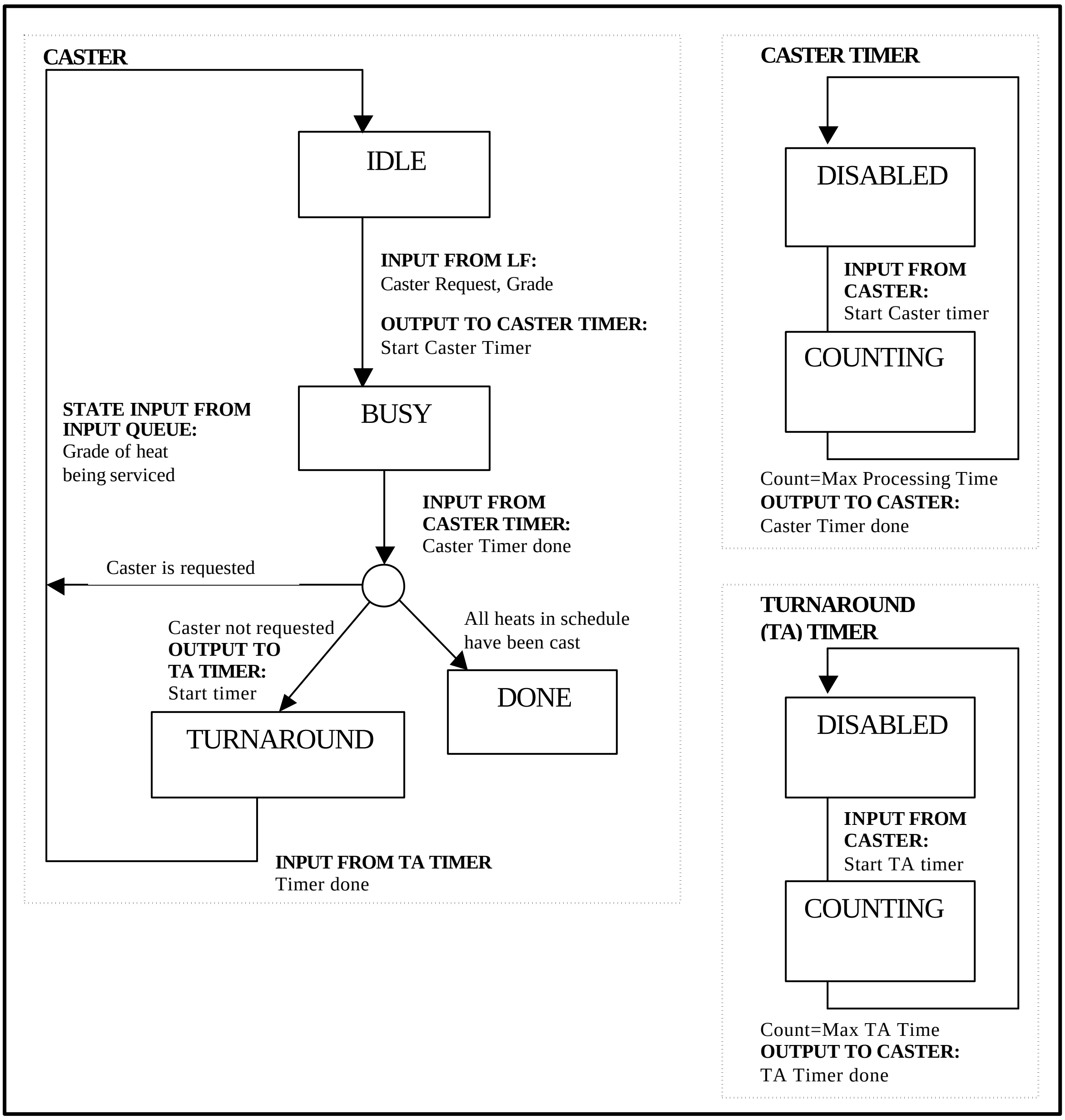


Figure 3.6: *Model of Caster, Caster Timer and Turnaround Timer*

3.3.4. Machine Coupling in the Model

In the model the coupling between the EAF, LF and Caster are as follows:

The EAF can start a heat once an EAF_request is made by the input queue. Before the EAF begins the heat it sends a signal back to the input queue to change states. This command tells the input queue that it is allowed to request for the next EAF in the schedule. Note the EAF will not return to its original "idle" state until it is finished heating the previous heat in the EAF.

The EAF will send a command to the LF to start the LF heat "LF_Start_timer", once it has completed its heating (this occurs when the EAF_Timer has exited the Counting state and is back at the Disabled state.

The LF will finish the heat once the LF_Time >= LFA_time, this will cause the system to leave the LFTime state and enter the Caster Wait state. Upon exiting the CountingLF state a "req_caster" signal is sent to the Caster. In the Caster Wait state the system is waiting for the Caster to accept the heat. Once the heat is accepted the Caster sends the Cast_A signal to the EAF_LF_A module seen in Appendix D. At this point the LF will return to its "Disabled" state, and wait for the next heat to be sent.

The STATEFLOW model serves the purpose of verifying that the schedule meets the hard constraints of the plant. Mainly, not allowing LF's to be filled with a heat until the previous heat has been delivered to the caster.

3.4. MANUFACTURING EXECUTION ACTIVITY MODEL

In this section the development of the model, scope of the algorithms and interfacing data will be defined based on the plant system requirements defined above. Applying the MES Architecture Modules to help determine the real-time signals and data needed, as well as the sampling periods required can do this. Some of the areas for MES defined through business rules are the response to certain events or actions that occur within the system and define the actions or responses to system stimuli. Some of the possible applications for improved integration that can be offered through the MES specifically the subgoals outlined in Chapter 1 – Section 3 are revisited and applied to a specific plant example. These are done using custom built algorithms that reside inside the modules found in *Figure 2.4* of Chapter 2, based on the situation that may arise in the case study outlined above.

3.4.1. Whole Plant Enterprise – Develop Steel Billets

The Activities depicted in *Figure 2.2* of Chapter 2 will be revisited and geared towards the case study defined above. In *Figure 3.7* the highlighted data are primary to the A4 block and activities A1, A2, A3, A5, and A6 are presented to show their relations to A4 in the context of this case study.

The Engineer Product and Process (A1) activity is broken down into the following two engineering groups; designing products and generating a manufacturing process plan along with alternative plans. The first group for designing products includes:

Generic	Specific
• functional requirements:	Mechanical properties of finished billets
• detailed design:	Billet weight, exact length of finished billets
• bill of material (*Product BOM*):	This could contain the materials and alloys required for the different heats in the schedule

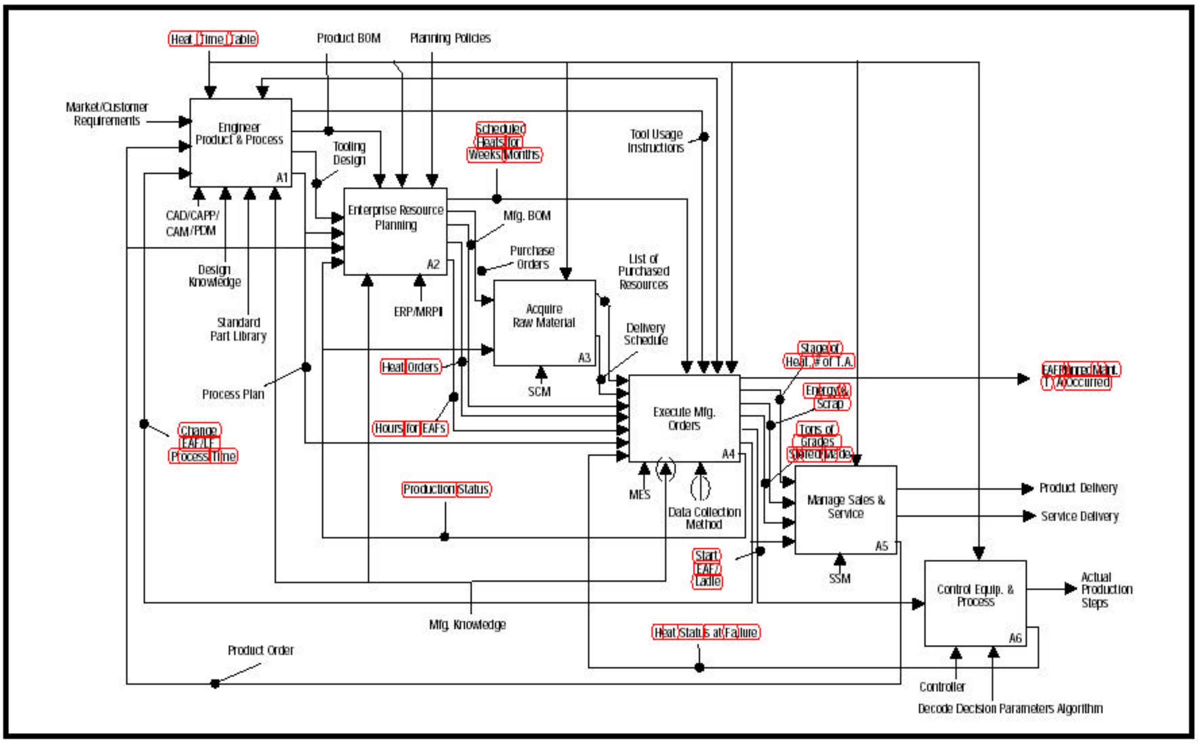


Figure 3.7: *Activity A0 – Develop Products*

The engineering group for manufacturing process engineering includes:

Generic	Specific
• process selection:	EAF temperature boundaries for heats
• operation planning:	material engineering group for defining transition steel cost
• workpiece routing:	crane design group
• equipment/device control program generation:	EAF engineering group

The A1 activity is important for providing design and processing information such as the information found in Table 3.1, Table 3.2, Table 3.3 and Table 3.4 above. This information is used for the downstream resource planning and manufacturing execution.

3.4.2. Enterprise Resource Planning

The Plan Enterprise Resources (A2) activity encompasses analyzing parts and performing make/buy decisions for all the parts. This involves developing a business plan and schedule to acquire the necessary resources (raw material or scrap) and/or to produce products for the market (finished cast products or billets). The resources include material, finished parts, equipment, and labor skills. The enterprise resource planning function includes:

Generic	Specific
• financial and order management:	*Purchase Orders* for production and material planning
• production and material planning:	assigning raw material and alloys needed for master production schedule
• master production scheduling:	schedule of heats to be produced for the week/month
• job definition:	*Manufacturing Orders (Heat Orders)*
• resource requirement specification:	*Resource Requirements (EAFs, LFs, Work Hours)*

3.4.3. Acquire Raw Material

Based on the resource requirement plan (Purchase Orders) specified in A2, the Acquire Resources (A3) activity purchases the resources from suppliers to meet the production schedule (List of Purchased Resources – raw materials: limestone, iron ore, steel scrap, etc.) and provides the Delivery Schedule for the purchased resources.

3.4.4. Manufacturing Execution Model with Data Breakdown

The key interfacing signals for the A4 module were summarized in Chapter 3. Here they are revisited with reference to the case study. Table 3.5 below summarizes all the interfacing data signals relevant to the A4 module, these signals are shown in *Figure 3.7* above. In Table 3.5 the column on the right defines the specific case study data name, which is representative of the generic one named in the second column.

Table 3.5: A4 Module Specific Case Study Data

Mechanism			
Data Collection Methods: The use of data collectors to obtain information on tundish level and EAF temperature readings, timing, personnel, lots, and other critical entities for production management in a timely manner. *Manufacturing Knowledge:* The information (rules, logic, examples) that a manufacturing engineer brings to bear on manufacturing engineering problems, including production techniques and implementation techniques. Many different types of manufacturing knowledge are used in different manufacturing activities, such as assigning heats out of specification to lesser grade heat, adding a certain composition of scrap metal and raw material to create a heat (assignment knowledge), and using power level grids to determine times of production (optimization knowledge), which are used in process planning, resource planning, production planning, and scheduling. *Manufacturing Execution System (MES):* A production activity management system that initiates, guides, responds to, and reports on production activities on-line and in real time to production management people. The system aids the Execute Manufacturing Orders activity.			
	Input Data	Specific Example	Plant-Specific Data
***	*Process Status*	Retrieve temperature monitoring information from each machine PLC and append to the genealogy file of each order every 2 minutes. Therefore know what is done in temperature spectrum, this helps verify that the times given by the *Detailed Schedule* are met. If predicted power level changes sound Alarm. A report of the conditions of a process being monitored. The report includes failed heats, failed machines, process changes or shifts, heat throughput, and so on.	Given the instant_ftime and EAF_failed variables. Determine current status of plant; heats that are active on the different machines and times needed before the processing of the heat is finished for the active machines (*Heat Status at Failure*). See Decode_Decision _Parameters (Chap. 4)
* / **	*Master Production Schedule*	A plan that specifies starting time and finishing time of each job in the job queue that are for producing products required by customers. The plan is made by the A2 module and contains	*Scheduled Heats for Weeks/Months*

		the scheduled heats for a time span of weeks to months.	
*	*Specifications*	Sets of time standards for each heat based on a regular operating power grid day (*Heat_Time_Table*). Data taken from Table 3.3 and Table 3.4 above. Similar to *Process Plan*- process parameters for manufacturing a product.	*Heat_Time_Table*
*	*Resource Requirements*	The times for resources required supporting production heats. Taken from *Heat_Time_Table*.	*Hours of EAFs*
* / **	*Manufacturing Orders*	The Heats to be done for the day according to the production plan and the *Master Production Schedule*.	*Heat Orders*
	Output Data	Specific Example	Plant-Specific Data
**	*Production Status*	A report on the state of all scheduled operations and production units. This includes the status of all active heats, finished heats and heats that are to start (information on resources), process setup, job schedule, and the "Caster_Wait" for Ladle (material routing).	Outputted to: A2 (feedback)
* / **	*Equipment Operation Instructions*	Commands to start EAF and Ladle. In the model everything is pre-sequenced in order. "Start EAF Timer" *Figure 3.4* is the instruction; "Start LF Timer" will follow *Figure 3.5* above.	*Start EAF/Ladle*
**	*Process Change Request*	Changes to process plan come through adjusting the "avg_tt_time", "LF_Time" if the EAF/Ladle are not running on full power.	*Change EAF/LF Process Time*
**	*Product Inventory*	The warehouse is not included in the model, but inputs to this model would be tons of Grades produced and tons stored after production.	*Tons of Grades Stored/Made*
*	*Cost Report*	Good for analysis: given operating times and energy input, as well as scrap lost due to failed heats and finished products produced. Note finished products could go in *Product Inventory*.	*Energy & Scrap*
***	*Product Genealogy*	Status of a heat, what stage it is in (given in real-time) for EAF, Ladle and Continuous Caster (CC). Rework is measured by the number	*Stage of Heat, # of T.A.*

		of times the "Turnaround" (T.A) Time of the CC expires.	
* / **	*Tool, Equipment Maintenance Order*	In the model there exists two types of maintenance; planned and immediate. The planned is scheduled maintenance on EAF machines. This is checked through *EAF Planned Status* before making the schedule for the day. The immediate is the rework needed on the Continuous Caster if the "Turnaround" (T.A) time expires. Disturbance Type 1: Machine Failing is not immediate since the schedule will re-work around it.	*EAF Planned Maint., T.A. Occurred*

The Manufacturing Execution A4 Module is decomposed into the following four sub-activities, as shown in *Figure 3.8*. The highlighted data found in *Figure 3.7* is re-highlighted hear in red, along with the internal data (in pink) that communicate between the A41, A42, A43 and A44 blocks.

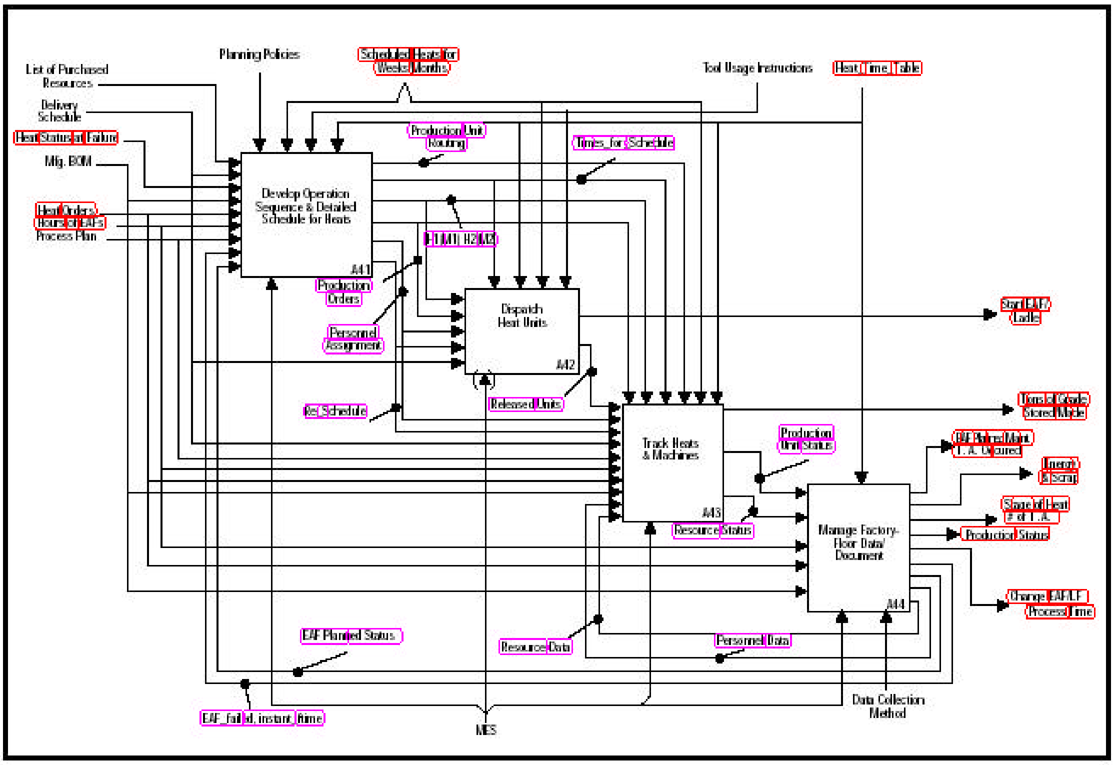


Figure 3.8: *Activity A4 – Execute Manufacturing Orders*

The sub modules are each defined however only the real-time data of module A41 and its sub modules will be presented with tables for this case study (Appendix F). The A41 module will be used as a blueprint schematic of where the algorithms should rest in the system and what inputs and outputs are needed for these algorithms. Chapter 4 describes the algorithms.

3.4.4.1. DEVELOP OPERATIONS SEQUENCE AND DETAILED SCHEDULE FOR HEAT STATIONS

The A41 module is responsible for taking the *Heat Orders*, *Hours of EAFs*, EAF *Planned Status* as input data, and the *Master Production Schedule* as control data. Using this it defines sequences for scheduled heats and machines to optimize productivity. This activity considers parameters such as minimizing makespan time and transition steel cost, and adjusting *Operation Sequences* when a machine has failed.

The data associated with the A41 module is summarized in Table F.1 of the Appendix F. In it a detailed description of the data can be found along with the rating system assigned for the particular data signal.

3.4.4.1.1.OPTIMIZE SCHEDULE FOR HEATS DEFINED TO BE DONE

The MES serves as the middle layer and therefore there still exists a higher level for Enterprise Resource Planning (ERP) A2 module where long term scheduling is done, Energy Management, Inventory Ordering and so on. As described in the previous chapter the A41 module of the MES architecture will handle the detailed day to day scheduling. Looking at A41 in more detail (*Figure 3.9* below), the following inputs and outputs needed and specific data associated with them are defined.

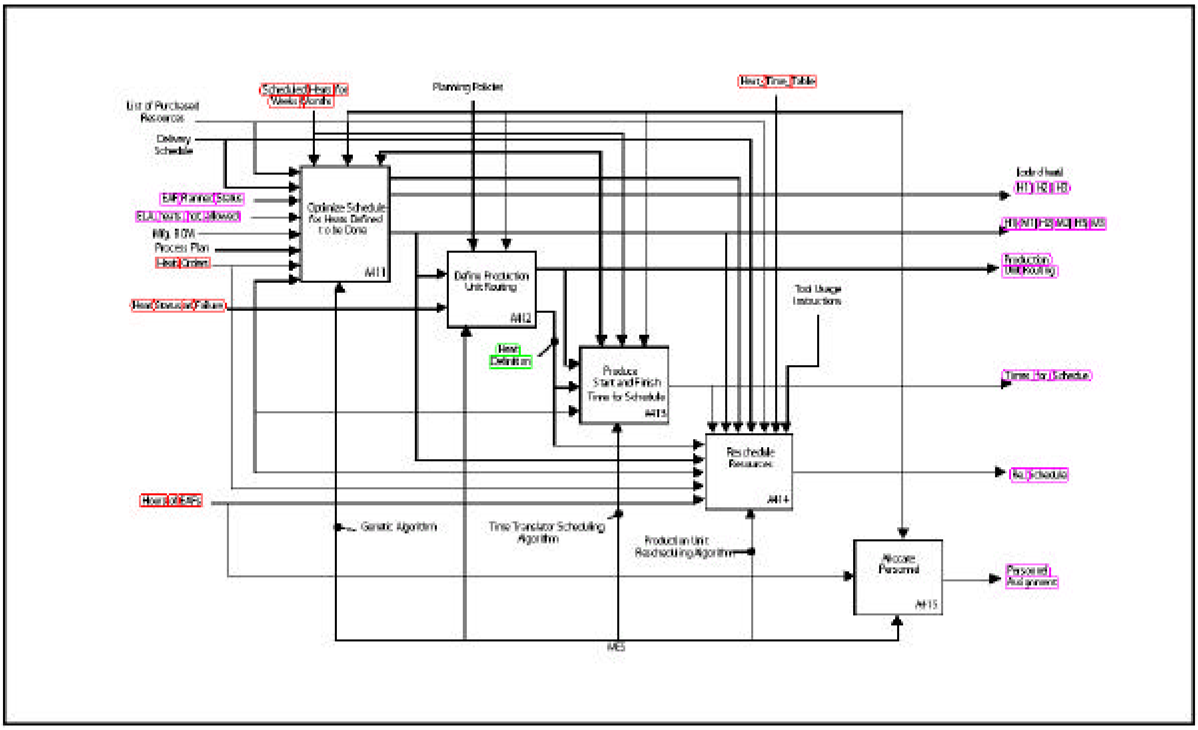


Figure 3.9: *Activity A41 – Develop Operation Sequence and Detailed Schedule for Heat Stations*

Module (A411)- Sequence production operations based on constraint priority characteristics. Given the heats needed (*Heat Orders*) to be produced for the day taken from the *Master Production Schedule* and the *EAF Planned Status* specifying the available machines for the day, find best sequence for cost and makespan to produce them in. This algorithm also considers the ELA_heat_not_allowed (*Updated Document*), which tells the scheduler which heats EAFA and LFA can handle based on capacity limits. This scheduler can be done using a GA (See Chapter 4)

The data associated with the A411 module is summarized in Table F.2 of the Appendix F.

3.4.4.1.2.DEFINE PRODUCTION UNIT ROUTING

The necessary interfacing done between the simulation model and the scheduling algorithm are communicated through this module.

Module -A412 outputs the *Production Unit Routing*. This is done by identifying the *Operation Sequence* and decomposing each heat in the schedule, with the proper processing time for each machine in the process.

3.4.4.1.3.PRODUCE START AND FINISH TIMES FOR SCHEDULE

Module -A413 outputs the *Detailed Schedule*. This is done by adding the start and finish time information to the heats dictated by the *Heat_Time_Table* and *Production Unit Definition.*

The data associated with the A413 module is summarized in Table F.3 of the Appendix F.

3.4.4.1.4.RESCHEDULE RESOURCES (FAILURE RECOVERY)

Module - A414 contains the algorithm for failure recovery after a machine has failed during processing (see Sections 5.4.3, 5.4.4 and 5.5.4). Data is provided by the *Process Status* alarm, along with the *Detailed Schedule* as an input (A413 output – *Times_for_Schedule*) and outputs the *Resource Schedule.*

The data associated with the A414 module is summarized in Table F.4 of the Appendix F.

3.4.4.1.5.ALLOCATE PERSONNEL

A415 –The case study does not detail this information, therefore no algorithm has been written for this module. A possible application for this module might be a personnel data base with a list of jobs that can potentially be assigned for this person. Therefore, a human machine interface (HMI) exists for resources, which controls which operator is allowed to sign in. The database information is also provided to the A422 module (see Section 3.4.4.2.2 below)

As we can see the A41 Module can be used as a blueprint schematic of where the algorithms should rest in the system and what inputs and outputs are needed for these algorithms. Chapter 4 describes the algorithms.

3.4.4.2. DISPATCH HEAT UNITS

The A42 Module found in *Figure 3.8* above, contains instructions for specific equipment operation (*Start EAF/Ladle*) signals used to control machine execution. This module can also keep track of the production units that are completed (*Released Units*) from the melt shop. This is done by noting the control input *Times_for_Schedule* and instantaneous time since the start of the schedule, and taking all heats whose finishing time on the CC is before the instantaneous time.

If the model included the rolling stages of steel manufacturing then this module would also handle which production unit in the queue is best processed next. The objective is to minimize the lead time and lateness. Assume the next stage after the melt shop is the rolling stage which has three machines A,B and C. Each steel piece must be processed first on A then on B followed by C, with a unique time on each machine. The A42 module could have an algorithm to handle this; it can be a search tree which looks at the times of production for different units on the downstream machines (B and C), and optimizes for fastest completion of the units. This method tries to minimize the waiting time for machines.

3.4.4.2.1.HEAT MATERIAL MOVEMENT

The control of material movement for this case study is all automatic. This means no discrete signal is needed to tell the crane when to move a heat from one workstation to the next. If this command was incorporated, then the *Equipment Operation Instructions* (*Start EAF/Ladle*) would contain this signal.

Activity A42 is decomposed into three subactivities, as shown in *Figure 3.10*.

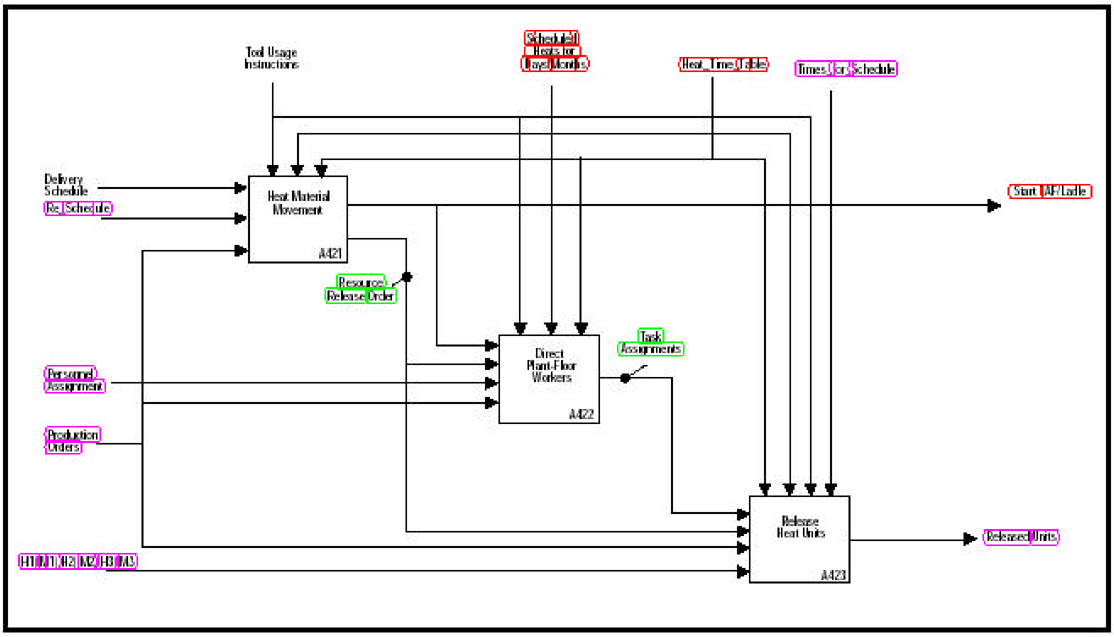


Figure 3.10: *Activity A42 – Dispatch Heat Units*

The A421 Release Resources module is based on resource allocation and releasing resources (*Resource Release Order*) for the production activity in a timely manner. This activity could be as simple as telling inventory or storage that raw material or scrap metal of a certain quantity or type should be ready for processing at a specific EAF at a certain time based on the schedule.

3.4.4.2.2.DIRECT PLANT-FLOOR WORKERS

The A422 module deals with assigning workers with adequate skills to perform specific operations according to the jobs for the day. This information of assigning tasks with due dates to workers comes from *Personnel Assignment* outputted by the A415 module.

3.4.4.2.3.RELEASE HEAT UNITS

Based on the *Resource Release Order* output of A421, the A423 module is responsible for *Releasing Units*, which simply triggers when the *Resource Release Order* command is executed. This means when the raw material or scrap is released into production the *Released Units* specifies the details of when and what was released for where.

3.4.4.3. TRACK HEATS AND MACHINES

The status information of a heat in production (*Production Unit Status*) is provided through the A43 module (see *Figure 3.8* above). The A43 module also provides information on who worked on it (*Personnel Assignment*) and component materials by supplier (*Released Units*). While at the same time providing the status information (*Resource Status*) on specified furnaces (Electric Arc or Ladle), such as number of heats completed by specific furnace machine.

3.4.4.3.1.HEAT INVENTORY TRACKING AND MANAGEMENT

The A43 is decomposed into five subactivities, as shown in *Figure 3.11*.

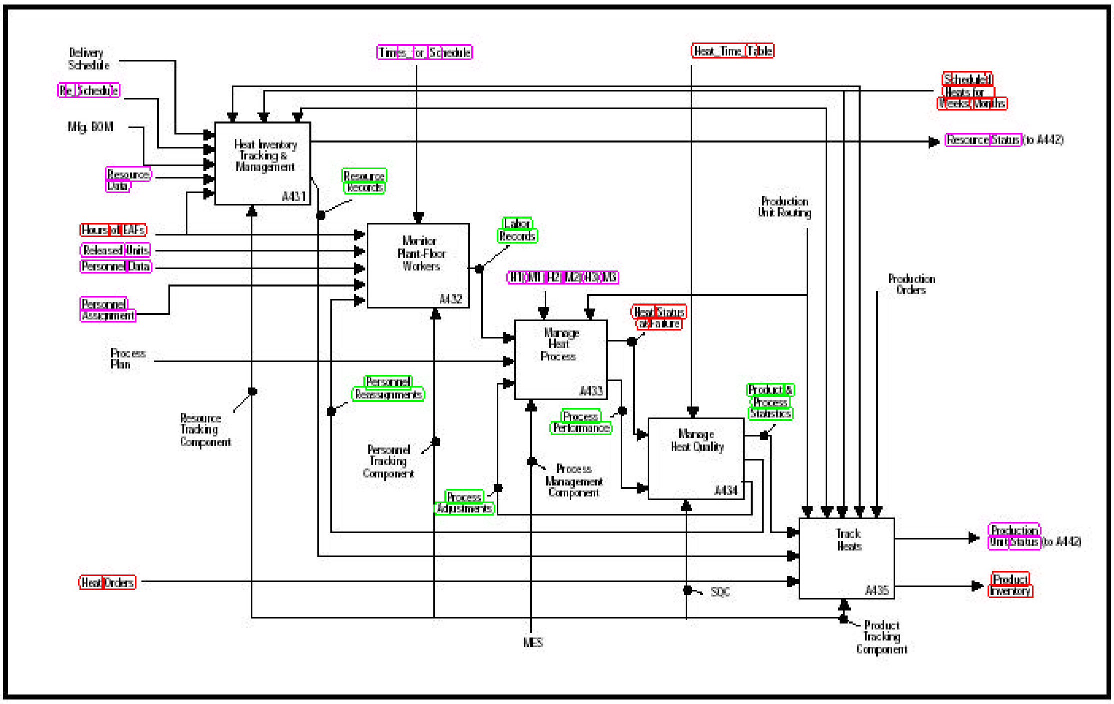


Figure 3.11: *Activity A43 – Track Heats and Machines*

The A431 module monitors the status of machines. This is done by knowing the *Times_for_Schedule* and tracking what has been done in the production process. The primary purpose is to track furnace usability, energy and material consumption in the manufacturing facility (*Resource Status*). With this information the activity module creates a record of history for machines that are necessary to be traced (*Resource Records*). This piece of information indicates the personnel using the machine and the heat being used, along with the time spent on the machine. The time spent on a machine is fixed by the schedule unless a disturbance occurs.

3.4.4.3.2.MONITOR PLANT-FLOOR WORKERS

The A432 module tracks personnel status through *Personnel Assignment* and *Reassignments* and reports the status on the labor assigned. The report could include information that could help in determining the level of quality performed by the operator. Such information may be; response time to power fluctuations in verifying heat is ready and time needed to verify proper heat composition by response of lab group Quality Assurance Metallurgists.

This last point is not included in the case study model but comes from the situation when a sample is sent to the lab, and a fast time response (minutes) is needed to verify whether to cast the LF heat as defined originally or change the heat to a different grade if the sample does not meet recipe requirements. The QA is done in A434, by outputting the *Process Adjustments* into the A433, which updates the *Process Status*. Here *Process Status* is renamed as *Heat Status at Failure*, since the model only handles disturbance of failed machines and not quality disturbances. If quality was modeled then *Process Status* would contain *Heat Status at Failure* and *Heat Quality*. Note *Heat Status at Failure* is updated by the A6 module (Section 3.6)

This is where the scheduler (A414) can be incorporated through the *Process Status* (*Heat Status*) which is outputted by the A433 module to the scheduling module. This information may tell the scheduler that the planned schedule will not complete a certain heat planned to be done for the day, since the quality is not met. This might mean that the heat could be downgraded to another heat planned in the schedule. The scheduler must find a new schedule at this point in the day in order to try and meet the *Heat Orders* for the day.

3.4.4.3.3.MANAGE HEAT PROCESS

The role of the A433 module is to monitor a production process and make timely decisions to adjust the detailed schedule and process plan when unexpected situations occur. This includes the request from QA to make *Process Adjustments* and reschedule since a heat does not meet the specifications (not included in the model). *Heat Status at Failure* is the *Process Status* signal sent to A414 to reschedule the heats since a machine has failed. This signal is updated by the A6 module (see Section 3.6 below). Therefore, the A433 module updates the *Process Status* for QA, and the A6 module updates *Process Status* for machine failures.

3.4.4.3.4.MANAGE HEAT QUALITY

The A434 module provides timely analysis of samples collected from heats to control product quality. This is explained in Section 4.4.4.3.2. Note when a heat does not meet the specification it is the job of this module to recommend which class of heats to categorize this heat under (*Process Adjustments*), so that it may be used for a different customer order.

3.4.4.3.5.TRACK HEATS

The A435 module monitors the progress of production using the *Product Tracking Component* Mechanism data and *Resource Records* Input data. With this it provides up-to-the-minute reporting on the production status (*Production Unit Status*). This signal includes the *Heat Status* if a heat does not meet quality standards and status of machines. This information goes into A442 (Section 3.4.4.4.2). A442 outputs the *Resource Availability* which includes information on Turnaround (T.A.) occurring, furnaces failing (*Scrap*), heats not meeting quality and stages of heats (for energy consumption) to A443 (Section 3.4.4.4.3) which gives the *Updated Document.* The *Updated Document* contains for this case model the ELA_heat_not_allowed, which tells the A411 scheduler which heats can not be handled by ELA and LFA based on their capacity constraints. This is updated only if a *Process Adjustments* is called by the QA (A4434) module to redefine heats. So at the beginning of the day when the A411 schedules the original *Operation Sequence (H1|M1|H2|M2|H3|M3)* the *Updated Document* already contains the restrictions of EAFA/LFA for the heats defined in *Heat Order* for the day.

3.4.4.4. MANAGE PLANT-FLOOR DATA/DOCUMENT

The role of the A44 module is to output the *Production Status* mission-critical data pertinent to production activities. This information is sent to the A2 and A3 modules to be analyzed for multiple purposes, such as heat throughput, heat quality, delivery, and machine maintenance. The management of documents, such as *Energy & Scrap* cost reports based on stages of heats and failed heats, machine maintenance orders (*EAF Planned Status*), inventory reports, *Change EAF/LF Process Time*, *Heat_Time_Table*, and so on is also handled in this module.

This activity module also manages the data collection, access, and distribution to the appropriate places. While providing versioning control of documents, such as *Heat Orders*, *Times for Heats*, production unit heat records, records of communication from shift to shift, heat standards, safety regulations, all of which can be included in *Updated Documents*. Here ELA_heats_not_allowed is the only parameter used by the algorithm and therefore the only signal defined by the *Updated Document*.

The A44 is decomposed into three subactivities, as shown in *Figure 3.12*:

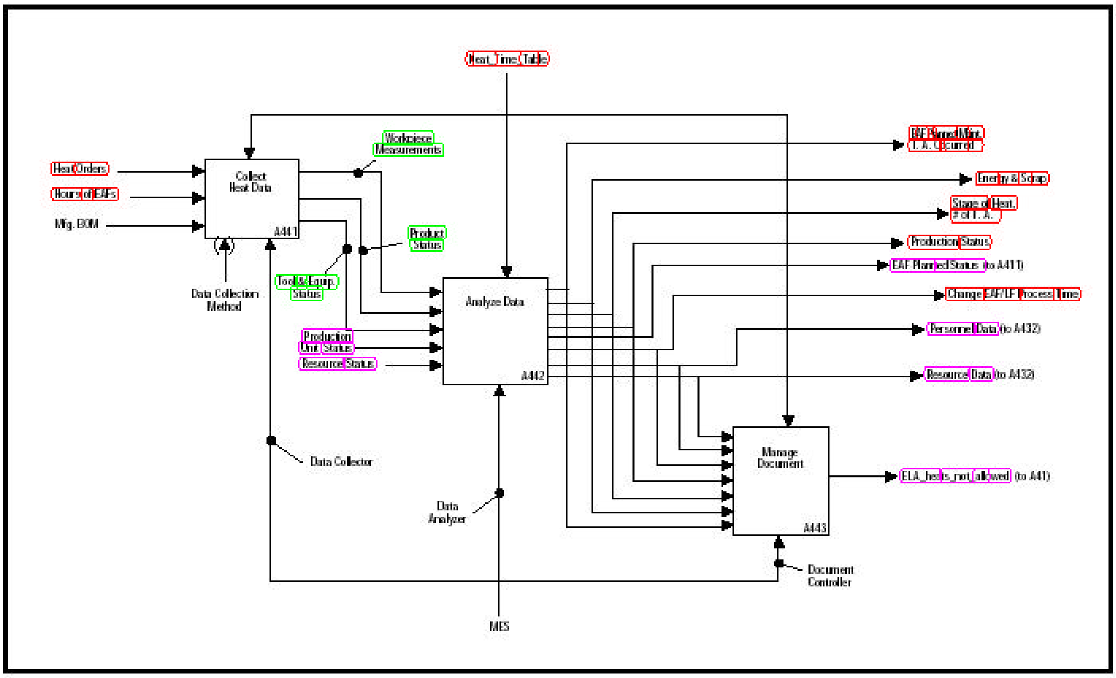


Figure 3.12: *Activity A44 – Manage Plant-Floor Data/Document*

3.4.4.4.1.COLLECT HEAT DATA

The A441 Collect Production Data Module uses data collection devices (Mechanism Data), to acquire data by measuring and sampling heats (*Workpiece Measurements* and *Product Status*), and production processes (*Tool and Equipment Status*) to provide support for analyzing the data (inputted into A442).

The A441 is further decomposed into three subactivities, as shown in *Figure 3.13.*

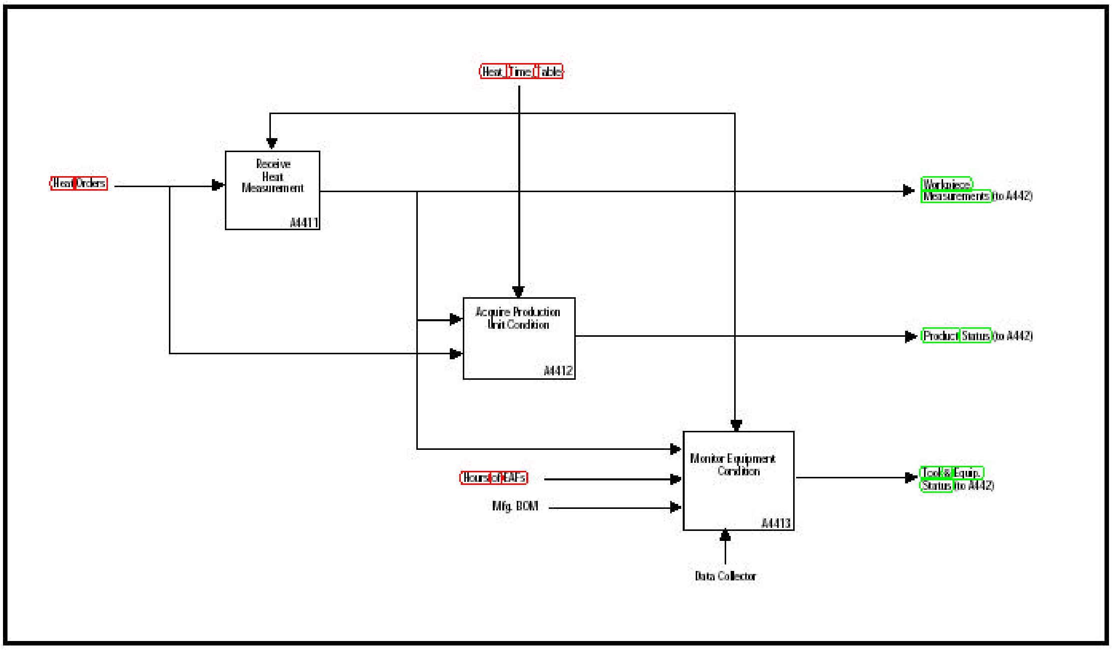


Figure 3.13: *Activity A441 – Collect Heat Data*

3.4.4.4.1.1. Receive Heat Measurement

Using factory-floor data collection devices the A4411 module, acquires and collects measurements on heat composition (*Workpiece Measurements*), labor records, and process conditions (heating times) in order to monitor the process performance and product quality.

3.4.4.4.1.2. Acquire Production Unit Condition

The A4412 module uses data collection mechanisms to acquire data on the heats. This is used to determine where the heats are and what heats have been finished. The module outputs the *Product Status* information on-line and up-to-the minute. This data is made available and visible to production as well as business management (through *Production Status* to A2 and A3).

3.4.4.4.1.3. Monitor Equipment Condition

Using the data measured from heats and machines, the A4413 module indicates the status of machines being used in production. The *Tool and Equipment Status* represents the Electric Arc Furnace, Ladle Furnace and Continuous Caster status. This information is inputted into A4424 to determine whether the machines are still proper to function or need adjustments to their heat times (*Change EAF/LF Process Time*), and if a planned or immediate maintenance is required (*EAF Planned Maintenance, T.A. Occurred*).

3.4.4.4.2.ANALYZE DATA

The A442 module uses the Data Analyzer Mechanism to provide up-to-the-minute reports of heat operation results along with the comparison to past history and business expectations. This allows the activity module to generate records and reports and to make them available for decision making and product tracking.

To perform a quality control method that focuses on continuous monitoring of a process rather than the inspection of finished products, a Neural Network can be implemented which monitors and learns best practice methods for future business expectations.

The A442 is also further decomposed into four subactivities, as shown in *Figure 3.14*.

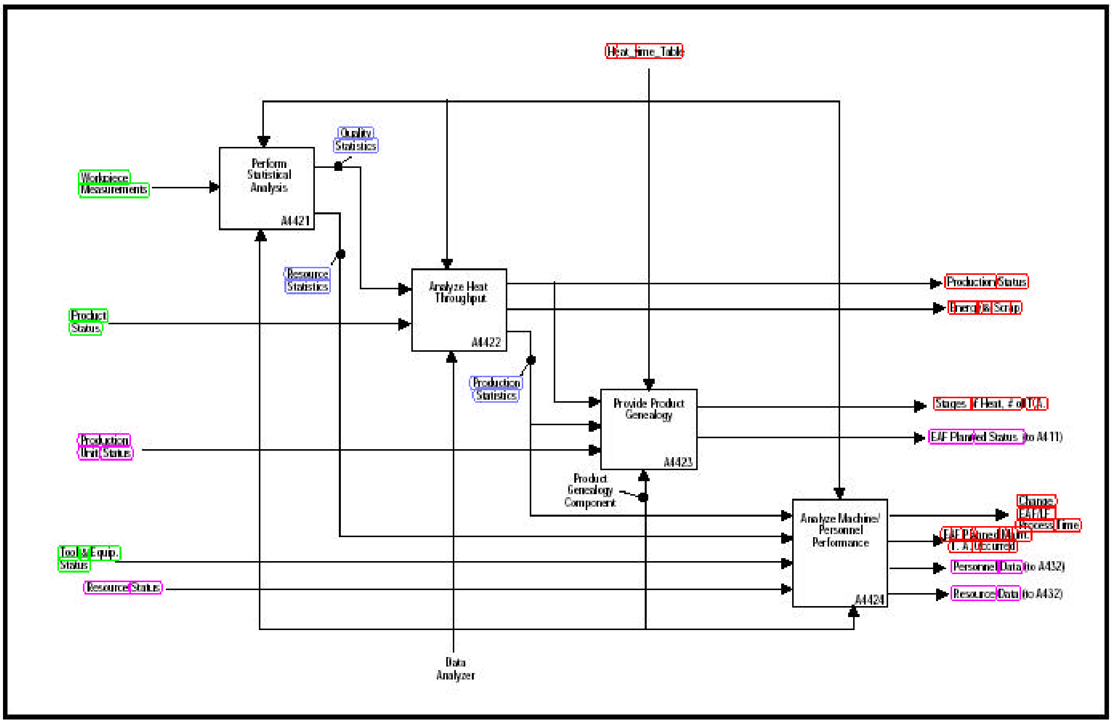


Figure 3.14: *Activity A442 – Analyze Data*

3.4.4.4.2.1. Perform Statistical Analysis

The A4421 module conducts statistical analysis on heat quality and machine resources. The outputted statistical data *Quality Statistics* and *Resource Statistics* are pertinent to the quality of the heat. This is important for measuring the *Energy & Scrap* as well as the *Change EAF/LF Process Time* and *EAF Planned Maintenance*.

3.4.4.4.2.2. Analyze Heat Throughput

The A4422 module provides the *Production Status* and calculates the quantity of energy needed for a product to be completed. It also measures scrap produced when a heat has failed during processing on an EAF. This is the outputted (*Energy & Scrap*) signal. The *Production Statistics* signal which also provides measurements and statistical analysis of the production process quality is also outputted by this module and made available for A4423 and A4424.

3.4.4.4.2.3. Provide Product Genealogy

The A4423 module creates records on the heats for traceability, including machines used (*Stages of Heat*), heat times, material supplier, operator identifications, heat measurements, and any exceptional processing conditions such as number of Turnarounds (*# of T.A*) occurring. The module gives the *EAF Planned Status* to the A411 scheduler, which needs to know when furnaces are available for production during specified time periods

3.4.4.4.2.4. Analyze Machine/Personnel Performance

Using the collected process data, the A4424 module analyzes the EAF machines status (*Tool and Equipment Status*) given by A4413 and estimates the maintenance schedule (*EAF Planned Maintenance, T.A. Occurred*). The module also analyzes worker's performance, such as heats completed with proper composition (*Personnel Data*). As well as indicating the condition of a furnace based on inspection or measurement analysis (*Resource Data*). This also includes adjustments to the machine heating times if the *Resource Statistics* (given by A4421) indicates the machines should not be running on full power (*Change EAF/LF Process Time*).

3.4.4.4.3.MANAGE DOCUMENT

The main function of the A443 module is to collect (or generate), maintain, and distribute production-related documents and records to support the production, factory-floor decision making, and product traceability. This is done using the *Document Controller* Mechanism. This software, controls records and forms that support the heat life cycle activities, such as procedures, recipes, programs, engineering change orders (ECO), shift-to-shift communication records, and so on. This document management activity module takes in many different data signals outputted from the A442 module and provides the appropriate information into the *Updated Document*.

3.4.5. Manage Sales and Services

The Manage Sales and Services (A5) activity consists of managing the sales of products and services to customers. This includes:

• product delivery dates and receipts:	note the cost and delivery dates are affected by plant operation and MES performance
• product composition:	heat recipes quality
• customer orders:	number of heats and type of heats
• product returns:	returned heats for reuse in another heat

3.4.6. Control Equipment and Process

Using factory floor data collection and monitoring, output the *Heat Status at Failure* (*Process Status*). The A6 module is fed the *Start EAF/Ladle* commands, which is taken from the *Times for Schedule*. The A6 module has the ability to detect when a machine has failed. This means the (*EAF_failed*) and failed time (*instant_ftime*) is known. With this information the Decode_Decision_Parameters Algorithm (see Chapter 4) can be executed by an interrupt when the failure occurs to provide the *Heat Status at Failure* with the following information:

- *LF_machine* – array size 3, component 1,2, and 3 contain the active heats in LFA, B and C respectively at time of failure. If a component has a zero, there is no heat in the associated machine at time of failure.
- *EAF_machine* - similar to LF_machine, components 1,2 and 3 represent the active heats in EAF A,B and C respectively.
- *time_left_in_LF* – array size 3, for the heats in LF_machine this array gives the time still needed to finish processing the heats in LFA,B, and C with reference starting at the time of failure.
- *time_left_in_EAF* – array size 3, similar to time_left_in_LF, but for heats in EAF_machine.
- *heat_in_CC* – the active heat being cast at time of failure.
- *time_left_in_CC* – the time left to finish casting the heat after the time of failure.
- *heats_left_in_schedule* – the heats that have not begun processing since the failure occurred.

This information provided by the *Heat Status at Failure* can be used in the Resource Rescheduling Algorithm (A414) to determine if Turnaround can be avoided and find a reworked schedule (*Re_Schedule*) for the instantaneous heat activity solution to meet least cost based on the availability of resources (see Algorithm Chapter 4).

Chapter 4

4. Algorithms

The algorithms mentioned in Chapter 4 are revisited in this chapter. Here the details on the functions used and the design methodology that went into making them are dealt with. The first algorithm explained belongs to the A411 module of Chapter 4 and uses a Genetic Algorithm (GA) [1] to solve the scheduling problem. Disturbances to the schedule are handled by the A414 module, which uses the detailed schedule outputted by the A413 module. The details of the A413 algorithm will not be explained in this chapter, although the code is provided in Hadjimichael [7], along with all other code used in the thesis. Finally the A412 module algorithm, which sends the equipment instructions to the STATEFLOW model, will be looked at in this chapter.

4.1. GENETIC ALGORITHM DESIGN (A411 MODULE)

GAs are inspired by Darwin's Theory. Specifically the way nature uses evolution to allow the fittest to survive. In this approach the solution will be found through a simulated evolution. The algorithm is started with a set of possible solutions. The set of possible solutions is named as the population. Each possible solution within the population is also called a chromosome. A fitness value is assigned to each chromosome based on a fitness function. Fitness is a measure of how well the solution solves the defined problem. Members of the population are selected to produce a mating pool. Two parent chromosomes are chosen for the mating pool, they are crossed over to produce new children chromosomes, which are hopefully better solutions than their parents. Mutation slightly alters the newly formed solutions. The mutation yields a possibility that a mutated solution will give a better solution, that crossover could not have produced. Solutions from one population (the children chromosomes) are taken and used to construct a new population. The whole process of selection, crossover, and mutation is repeated with the new generation of solutions. This is motivated by the hope that the new population will be fitter than the old one. After the process occurs over several generations, the result will be a final population that has converged to an optimal solution.

The optimal schedule for the given heats in the steel production process is determined by building the GA in MATLAB. The full code with comments can be found in Hadjimichael [7]. Following will be a description of how the GA was developed and details for running the algorithm.

4.1.1. Encoding the problem through a chromosome

The first step in the GA is to "translate" the real problem into "biological terms". Formatting the chromosome is referred to as encoding.

The task here is to find a representation of a schedule through a chromosome. Recall the driving mechanism for the algorithm is to encode a solution to the problem in the chromosome, and based on how good the solution is, choose this solution (chromosome) for a future solution until the optimal solution is found.

As mentioned in Chapter 3 (3.2.2) the schedule is based on running between 2 to 6 heats of steel depending on the production for the day and each heat can be of a different grade and weight. Table 3.1 of Chapter 3 defines the grade and weight used for this specific case. The melt shop production process has 3 stages, these are; the EAF, the LF and the CC. When making the schedule, there are 3 EAFs (EAFA, EAFB and EAFC) to choose from and these EAFs are coupled to the 3 LFs (LFA, LFB and LFC) respectively. This means there are 3 choices for a heat: EAFA and LFA, or EAFB and LFB and finally EAFC and LFC. Depending on the sequence of the order, a transition cost can occur (see Table 3.2 of Chapter 3). Also the makespan defines how long the production run will take in order to complete the scheduled sequence of heats. This time also is affected by the scheduled sequence, since queuing can occur between stages for the different heats.

The following approach was chosen to encode the schedule into a chromosome. There are up to 6 grades of steel (A, B, C, D, E and F), and 3 possible starting machines in the process (EAFA, EAFB and EAFC) for the first heat in the sequence. The next heat in the sequence can also choose from any of the 3 choices of EAFs. This creates a solution space of up to $6\times3\times5\times3\times4\times3\times3\times3\times2\times1\times3 = 524880$. This number will be smaller since some heats are too large for EAF A's capacity, as defined by Table 3.3 of Chapter 3. The convention used for encrypting a chromosome is shown in *Figure 4.1* below. Note this figure is for a chromosome with only 3 heats (in *Figure 4.1* these are the odd number slots from the left). Each heat has 1 of 3 coupled machines (in *Figure 4.1* these are the even number slots from left) too choose from. The coupled machines are labeled as 1 for EAFA and LFA, 2 for EAFB and LFB and 3 for EAFC and LFC.

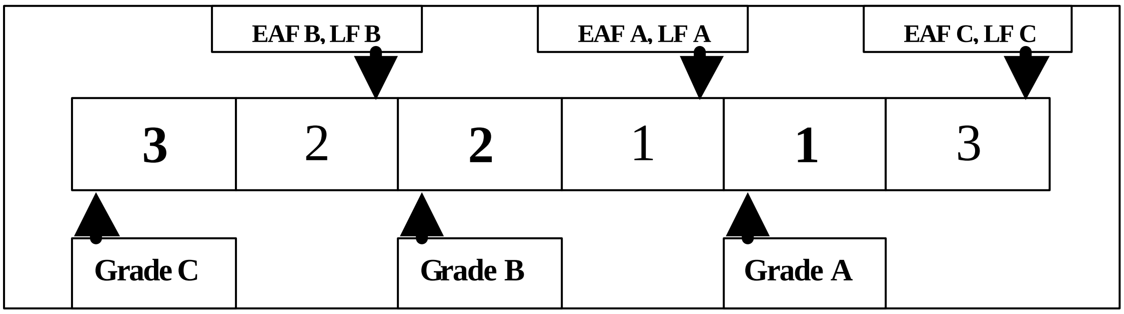


Figure 4.1: *Solution Encoded as a Chromosome*

The above sequence, 32 21 13, represent the schedule explained in Table 4.1 below.

Table 4.1: Schedule represent by 32 21 13 Encoding

MACHINE	TIME				
EAF A	GRADE B				
EAF B	GRADE C				
EAF C	GRADE A				
LF A		GRADE B			
LF B		GRADE C			
LF C		GRADE A			
CASTER			GRADE C	GRADE B	GRADE A

4.1.2. Population of Chromosomes

The initial population of chromosomes is randomly generated. The population consists of a subset of all the possible solutions. As a default a value of 24 chromosomes was chosen. This can be changed in the 'launch_parameters.m' file, which must be run before executing the GA (see Hadjimichael [7]). The GA can handle only even numbers for the initial population size.

4.1.3. Penalty Factor and Objective Fitness Function

4.1.3.1. MAKESPAN AND TOTAL COST OF A SCHEDULE

In order to determine the optimal schedule, the makespan and the total cost of transition steel produced by a given schedule must both be minimized. Table 4.2 below illustrates how to determine the makespan for a given schedule. For illustration purposes assume the encoding of a given schedule is 12 22 32. By observing Table 4.2, we can see that at 142.5 minutes, the caster is ready to cast the next sequence, but the 2nd heat of Grade B steel is still being heated in the ladle furnace. Recall from Table 3.3 (Section 3.2.2), that a turnaround delay of 20 minutes is introduced into the makespan if casting does not occur continuously. The makespan for this schedule is 302.5 minutes. This is can be verified using the STATEFLOW model in SIMULINK (Section 3.3.1)

Table 4.2: Makespan for Schedule 1-2 2-2 3-2

MACHINE	**TIME**
(min)	
EAF A	
(min)	60 120 180
EAF B	**A B C**
(min)	
EAF C	
(min)	
LF A	
(min)	80 150 200
LF B	**A B C**
(min)	
LF C	
(min)	142.5 162.5 242.5 302.5
CASTER	**A B C**

LEGEND:

	Heat of Grade A
	Heat of Grade B
	Heat of Grade C
	Turnaround Delay
	Machine not used

The cost associated with transition steel produced will be illustrated through analyzing the schedule given in Table 4.2 (12 22 32). The total cost for this schedule is determined by first referring back to Table 3.2 (Section 3.2.2). There a transition cost of $200/ton is associated with processing Grade A steel, then Grade B steel. This transition will produce a mixed grade billet of 25 tons. There is no cost associated with the transition steel produced when processing Grade B steel then Grade C, as the transition billet will be of equal value of a Grade C billet. Therefore, the total cost associated with this schedule is ($\$200/ton \times 25tons = \$5000$) five thousand dollars.

4.1.3.2. DEFINING THE PENALTY FACTOR

It is important in steel production to keep the Caster continuously flowing with molten steel. There is a downtime of 20 minutes needed, if the flow of molten steel has stopped after the Caster has begun being fed molten steel through the tundish. If the Caster is ready to cast the next heat of steel, and the next heat has not finished being heated in the Ladle Furnace, the Caster must be shut down and restarted. This introduces an extra 20 minutes of delay in the schedule, as seen in Table 4.2 with the vertical line cell between casting Grade A and B. This situation results in a sub-optimal schedule, since there is a time and cost associated with turning the caster around. In order to label this schedule as a poor solution, a Penalty Factor is associated with it. The Penalty Factor is a discrete value, which indicates whether the Caster has to be turned around, and how many times it had to be turned around. If the Penalty Factor is represented by P. Then a schedule with a Penalty Factor of 2P, indicates the caster had to be shut down and restarted twice. The following section will explain where we introduce the Penalty Factor into the GA.

4.1.3.3. DERIVING THE MULTI-OBJECTIVE AND FITNESS FUNCTION

Direct multi-objective optimization may be particularly useful when several conflicting requirements have to be satisfied at the same time. The basic feature of GAs is the multiple directional and global search capability [6]. In fact, assigning arbitrary weights to combine different design objectives into a single function optimizes the given set of objectives. A multi-objective optimization problem is to find the set or to approximate it with a representative subset. Here the multi-objective function, $f(x)$, is an equation which includes all the objectives that need to be considered in our schedule. These are minimum makespan and minimum cost. The schedule which yields the smallest $f(x)$ value is considered the optimal schedule.

The objective function is defined as follows:

$$f(x) = a \cdot C + b \cdot M' + P$$

where:
a is a weighting factor for C
C is the Cost
b is a weighting factor for M'
M' is the makespan subtracted from the adjust_makespan
P is a Penalty factor

The weights a and b are used to express the relative importance of cost and makespan. These values were assigned 0.1 for a and 40 for b. In the function these are inputs that can be adjusted. Note these inputs are internal to the algorithm and do not come from other modules of the MES architecture. The same goes for all other inputs into the function (genetic_algo.m) except for the *Heat Orders* (Heats), *Heat_Time_Table*, *EAF Planned Status* (ELA_DOWN, ELB_DOWN and ELC_DOWN) and ELA_heat_not_allowed. The weighting values were chosen through trial and error, and state that a minimal makespan is considered more important than a minimal cost. The values are defined in the file 'launch_parameters.m', which must be run before executing the GA, see Hadjimichael [7] on procedures for running the algorithms.

The total cost C of the transition steel produced as a result of the schedule ranges between $0 to depending on how many heat types are being used. For this problem the most costly transitions are defined by Grade A to Grade B/E, or D to B/E, and each of these transitions cost $5000. M' represents the makespan (M) of a schedule subtracted from the adjusted makespan (270 minutes). The makespans of all possible schedules depend on the values assigned in the heat_time_table.m file. These values are made to represent the availability of power consumption allowed for the given shift. If power consumption must be held to a minimum then high values should be assigned to the machines in the *Heat_Time_Table*. Therefore, it is important that M' can not be negative, this is done by guaranteeing that the minimum makespan with the heats and heat times chosen is larger than 270 minutes. If not the value of adjusted makespan of 270 minutes must be reduced in the fitness function algorithm (generate_f_fit.m). Note the CC's times are fixed and cannot be changed due to the power consumption allowed. The following values have been defined for a regular day's power consumption shift (see Table 4.3 below). Note these are similar to the values given in Table 3.3 and Table 3.4 (Section 3.2.2). For more detail on the reasoning behind this feature please refer to Section 4.5 POWER LEVEL FLUCTATIONS.

Table 4.3: Heat Time Values for Machines

	Heat 1	Heat 2	Heat 3	Heat 4	Heat 5	Heat 6
EAF A	0	50	50	0	50	50
EAF B	60	60	60	60	60	60
EAF C	60	60	60	60	60	60
LF A	20	30	20	20	30	20
LF B	20	30	20	20	30	20
LF C	20	30	20	20	30	20
CC	62.5	80	60	62.5	80	60

The value for the makespan has been modified to M' to emphasize the difference in values. The penalty factor, P, as described above was assigned a discrete value of 400. This factor is used to increase the value of $f(x)$ when the caster is turned around.

Returning to our example schedule 12 22 32,

$$f(x) = 0.1 \cdot (\$5000) + 40 \cdot (302.5\text{min} - 270\text{min}) + 400 = 2200$$

The fitness function $F(x)$ describes how "fit" the chromosomes are. That is, how well the schedule minimizes both cost and makespan. The higher the value of $F(x)$ the better the solution. Therefore, the relationship between the fitness function $F(x)$ and the objective function $f(x)$, can be described as follows:

$$F(x) = \frac{1}{f(x)} \cdot 100$$

Which means a minimum value of $f(x)$ will result in a maximum value of $F(x)$. To clarify this further, refer back to the example schedule 12 22 32, here

$$F(x) = \frac{1}{2200} \cdot 100 = 0.045$$

One could have chosen to use say $F(x) = A - f(x)$, where A is some constant. This would avoid any non-linearity that may arise from taking the inverse of the linear $f(x)$ function. However, in this case the non-linearity is not a concern since the path to the optimal solution does not follow a linear path, but does converge to an optimal heuristic solution.

4.1.4. The Selection Method

In order to reproduce offspring, parents need to be selected. One of the most commonly used methods is roulette wheel selection. The key for roulette wheel selection is fitness. The fitter the chromosomes are, the more chances they will have to be selected. This selection method assigns a higher probability of selection to chromosomes with a higher fitness. *Figure 4.2* below helps illustrate this.

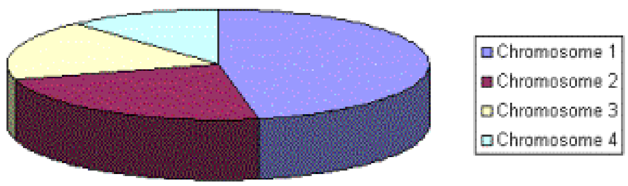


Figure 4.2: *Roulette Wheel Selection*

The Roulette Wheel method can be summarized in the following 6 steps [9]:

1. For each chromosome in the population, determine the objective function $f(x)$, and the fitness function $F(x)$.

2. Determine the average $F(x)$ for the entire population. Recall that the population size chosen was 24.

$$F(x)_{avg} = \frac{\sum F(x)}{Population \quad Size}$$

3. Determine the expected number of copies of each chromosome in the mating pool. A chromosome with a higher $F(x)$ is expected to have a larger number of copies of itself in the mating pool.

$$Expected\_Count(x) = \frac{F(x)}{F(x)_{avg}}$$

4. Determine the probability of selection for each chromosome.

$$Selection\_Probability(x) = \frac{Expected\_Count(x)}{Population \quad Size}$$

5. Determine the cumulative probability for each chromosome by summing the previous chromosome's cumulative probability with the current chromosome's selection probability. This process will produce cumulative probabilities ranging between 0 and 1.

$$Cumulative\_Probability(x) = Selection\_Probability(x) \\ +Cumulative\_Probability(x-1)$$

6. Choose a random number between 0 and 1. Determine between which two chromosomes' cumulative probability the random number lies. The chromosome with the cumulative probability just greater than the random number will be selected for the mating pool. In total, 24 random numbers are chosen, to place 24 chromosomes in the mating pool.

4.1.5. Crossover and Mutation Methods

4.1.5.1. CROSSOVER ALGORITHM

The driving force for evolution is reproduction. Crossover and mutation perform this task. Pairing chromosomes sequentially within the mating pool does the crossover process of producing new solutions from existing solutions. The 1st chromosome in the mating pool is paired with the 2nd, and so forth, producing 12 pairs from the 24 total chromosome population. The crossover process is performed twice. The first time, one parent will be the "dominant" parent; the second time the other parent will be the "dominant" one. An 80% probability of crossover was chosen. This value along with the number of crossover points to be done per mating parents can be modified in the launch_parameters.m file. It should be noted these values will affect the number of generations before convergence. An analysis of results for different number of crossovers and crossover probabilities was not performed since the intent of the thesis is not to learn how different factors affect the GA. If the pair does not crossover, a copy of the dominant parent will enter the next generation of solutions. The crossover process is illustrated in *Figure 4.3*, and Steps 1 to 4 below. This particular crossover scheme preserves the uniqueness of the heats, by making sure to always carry all the heats to the children. This guarantees a child chromosome in the solution space.

Elitism is the term used when the fittest member of the population is automatically copied to the next generation without any changes made to it. If elitism is used then the fitness function will at least be as good as the fittest member from the previous population.

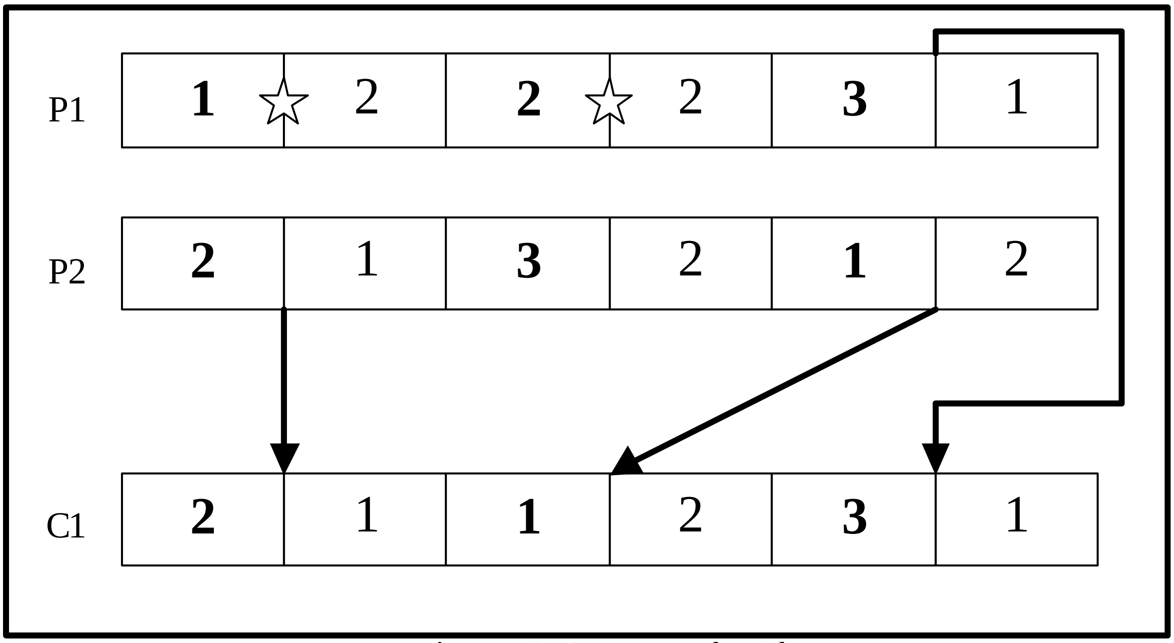


Figure 4.3: *Crossover Algorithm*

1. Two parent chromosomes (P1, P2) are selected to produce one child chromosome (C1). The dominant parent is P1.
2. Two random heats are chosen in P1 (12 and 22).

3. C1 inherits both the value and location of the unchosen heat (31).
4. C1 inherits the order and value of the two chosen heats from P2. In P2 Grade B is scheduled before Grade A, and Grade B is processed on EAFA. Therefore, C1 inherits (21 and 12) from P2.

The above illustration was for 2 crossovers. A similar method is applied for up to as many crossovers as there are heats. If there are 4 crossovers chosen for 6 heats, the 2 positions that are not chosen are automatically copied from parent 1 to the offspring. The four jobs in parent 2 that correspond to the jobs in the cross-sites location in parent 1, are copied to the offspring with the order maintained from parent 2.

4.1.5.2. MUTATION ALGORITHM

When a child chromosome is produced it is important to allow for a small probability that mutation can occur. This sometimes helps the GA find a solution faster, or jump out of local optimal solutions to the global solution. Here a 1% probability is assigned to mutation. This can be modified in the launch_parameters.m file. Mutation involves randomly selecting a heat in a schedule, and changing its starting furnace. In *Figure 4.4* below, heat 2 is randomly chosen. Mutation causes the heat to use EAF B now instead of EAF A. When creating the original population (createPopulation.m), the status of the plant must be considered. If an EAF and LF are down (planned maintenance) they must be removed from the initial guess of the schedule and cannot be mutated in later. This was a strict constraint that must hold in the chromosome for all its generations or else the schedule will not work for the current plant status.

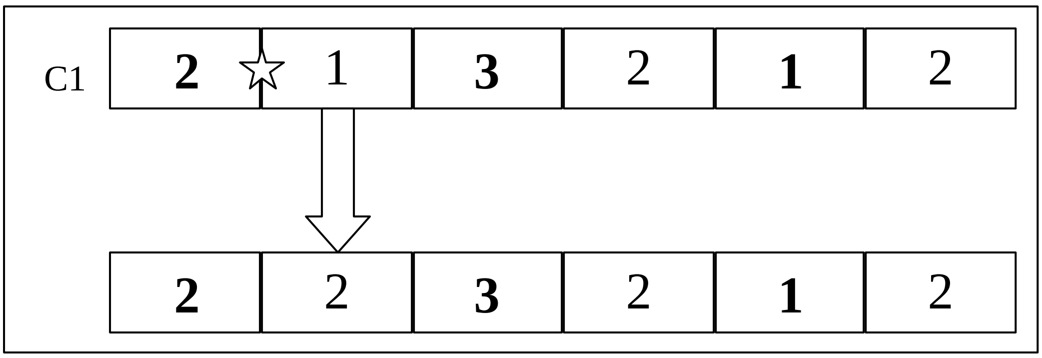


Figure 4.4: *Mutation Algorithm*

4.1.6. Regeneration

Finally, all the stages in the GA are complete, for one generation. The last step is to replace the previous generation with the resulting children from the crossover and mutation procedures. After running the GA for several generations (max_gen), it is anticipated that the population will consist of a majority of the fittest chromosomes. The dominant chromosome pattern in the final generation will be considered to be the optimal solution. If the solution did not converge for fitness, cost and makespan, then an optimal solution can not be concluded as found. The results are explained in Chapter 6. Refer to Hadjimichael [7] for details on the code architecture, function variables and the steps for running this algorithm.

4.2. GA FOR OTHER APPLICATIONS

The GA is good for disturbance scheduling, since it can take what is available and find an optimal solution for these current conditions. This algorithm is also applied to Disturbance Handling (Section 4.3). More details on disturbances types found in steel manufacturing can be found in [19]. The GA is meant to optimize the day to day schedule. In the STATEFLOW model only 3 heats can be inputted and run consecutively in a half day. After this the new 3 heats and associated machines must be redefined in through the schedule input block for the second half of the day. Therefore, the GA will take up to 6 heats and find the optimal schedule for the full day based on the availability of the 3 machines. The GA has the flexibility of giving a schedule for 2-6 heats.

4.3. DISTURBANCE HANDLING

The most custom portion of the MES is exception management. Specifically how a company responds to plant disturbances, e.g., what decisions should be made when a WorkStation suddenly goes down or when the available power consumption is changed. The next two Sections 4.4 and 4.5 look at how the MES should be able to take these changes in stride and respond with alternative actions.

4.4. MACHINE FAILURE RECOVERY

Failure recovery can only be effective with a detailed understanding of the current situation and a control system that allows immediate change of resource assignment. With a properly designed MES system, the information necessary to view all the options is readily available. This information is provided by the A6 module (*Heat Status at Failure*). The system information is based on the idea of using monitoring and data collection. When certain events occur they are recognized as triggers causing a corresponding business rule to be followed within the integration layer. The MES system is made to respond to this alarm. The goal is to determine a recovery strategy to bring the execution to the best state. The heuristic to be used here is created as a tool for solving this disturbance, but is not necessarily optimal.

The A414 module provides the ability to respond to unanticipated events that affect the production plan. The order rework requirements and routings are generated and reinserted into the schedule when an EAF fails during a heating. Here the re-assignment should not cause more disturbances. It is assumed that an LF can not fail during a heating. If an EAF fails the heat it was processing must be reprocessed from the start through another available EAF. The LF will finish heating the heat it is currently processing and then it will not be used for rest of the schedule, since it is coupled to the EAF that failed.

4.4.1. Machine Monitoring

As mentioned in the above section the A6 module provides the information necessary to allow the rescheduler to make a decision. The A6 module contains the Decode Decision Parameters Algorithm. This algorithm is best explained by first understanding what information is needed to interpret the status of the melt shop.

4.4.2. Decoding the Status of the Melt Shop

The following schedule found in Table 4.4 below will help in portraying the status of the plant and action of events that can take place once an EAF machine fails. This schedule is the expected daily schedule for the day without any disturbances occurring. The information in Table 4.4 for start and stop times on the different machines can be formed given the outputted *Detailed Schedule* (*Times_for_Schedule*) by the A413 module. The A413 module produces this data by running its Time Translator Scheduling Algorithm.

Table 4.4: Optimal Schedule: **61 31 22 51 42 12**

MACHINE	**TIME**
(min)	50 100 150
EAF A	**6** **3** **5**
(min)	60 200
EAF B	**2** **4** **1**
(min)	
EAF C	
(min)	50 80 100 140 150 280
LF A	**6** **3** **5**
(min)	60 200 360 422.5
LF B	**2** **4** **1**
(min)	
LF C	
(min)	80 140 200 280 360 422.5 485
CASTER	**6** **3** **2** **5** **4** **1**

For an instantaneous failure time of 110 minutes the following data is extracted from Table 4.4 by the A6 module algorithm to be inputted into the A414 Production Unit Rescheduling Algorithm. The term 'active heat' refers to a heat that is currently being processed by a machine at the instantaneous time of failure. For the LF's it can be seen that the active heats are 3 for LFA, 2 for LFB and 0 (or none) for LFC, as defined in "*LF_machine = [3,2,0]*". Each has a time left of 30 minutes and 90 minutes respectively. These are defined in "*time_left_in_LF*". For the EAF's, there are two active heats, however one of them heat 4 cannot be completed since EAFB has failed. Hence only heat 5 is active since the failed one (heat 4) must be reprocessed again at a later time. Thus the status for the active heats in the EAF machines is "*EAF_machine = [5, 0, 0]*" with times defined as "*time_left_in_EAF = [40, 0, 0]*". The continuous caster has heat 6 active at time of failure ("*heat_in_CC = 6*") and the time left is 30 minutes ("*time_left_in_CC = 30*"). Lastly, the heats not yet active at time of failure is 1, with a scheduled process on EAF B (machine 2), and the failed heat 4 which was being processed on EAFB, hence *heats_left_in_schedule = [4, 2, 1, 2]*. Where the odd positions are the heats and the even positions are the machines (recall EAFB-LFB are represented by a 2).

With the above data inputted into the A414 module its Production Unit Rescheduling Algorithm will perform the analysis. The function's objective is to send heats to the CC at a cost effective sequence when considering transition steel produced and with the main priority of avoiding Turnaround.

The pseudo code and architecture for the Production Unit Rescheduling algorithm can be found in Appendix G.

4.4.3. First Stage (LF) Rescheduling

The first stage of rescheduling which deals with the first active heats to be sent to the CC, after the current heat in the CC is finished processing at time of failure, is divided into two categories. The first looks at the algorithm under the case where Turnaround is avoided. The second looks at the case where Turnaround is present.

4.4.3.1. FIRST STAGE (LF) RESCHEDULING WITHOUT TURNAROUND

The method in deciding which order to send the active heats to the CC is done by first choosing the active heats in the LFs with the primary intent of no Turnaround occurring. If this is the case, the algorithm proceeds by choosing the sequence of active LFs for least transition cost between it and the current heat in the CC at time of failure. Here, since a heat was found to be ready without Turnaround occurring, it can be assumed that all remaining heats in the LFs will not have a problem with Turnaround, due to the large time difference in processing the heat in the LF and waiting for the casting to finish. Therefore, the remaining active heats in LFs are only chosen for least transition cost based on following the one chosen for no Turnaround and cheapest transition cost with the current heat in the CC.

4.4.3.2. FIRST STAGE (LF) RESCHEDULING WITH TURNAROUND

When rescheduling after failure and Turnaround cannot be avoided, the calculated best sequence for lowest transition cost is done without considering the heat in the CC at time of failure. This means the sequence is calculated without using the heat in the CC as the first fixed heat of the sequence. This is the case since Turnaround calls for having to start with a fresh cast, and starting with a new cast means there will be no transition with the current heat in the CC.

4.4.4. Second Stage (EAF) Rescheduling

After the ordering of the active heats in the LFs is completed as described by the methodology in First Stage Rescheduling above, the next step is to choose the ordering for the active heats in the EAFs. This is done with a similar methodology to that of the LFs. First the algorithm considers if the active EAF heat can be poured without Turnaround occurring. This depends if there was any heats in the LF's at time of failure and on the processing time still needed for the EAF heat plus the next stage time needed to be processed by the LF. If this is the case then the sequence is chosen for least transition cost after fixing the first heat to go. If this is not the case then the sequence for lowest cost with all active heats in EAFs is looked at while considering all the possible sequences to order them in.

4.4.5. Final Stage of Rescheduling

The above algorithm performs an accelerated simulation to determine if Turnaround can be avoided for the active heats and if so, what is the least cost. The algorithm also gives a solution for least cost, if turnaround cannot be avoided for the active heats. The main improvement is the active heats are still completed in a best possible sequence after the failure. Whether Turnaround occurred or not, these heats would still need to be cast.

Once the second stage is completed and all the active heats in the EAFs have been given a sequence to follow, the final stage calls on the GA. Here the GA is a modified version of the one explained above, since the population is fixed by the first heat (this being the last heat from the sequence of active EAFs). The GA finds the optimal sequence of the remaining heats that had not started at time of failure along with the heat that failed and must be redone from the beginning.

In the case where turnaround can not be avoided it might be better to optimize starting from after turnaround and using a full fitness function (i.e., GA). A comparison can be made if the cost of losing the active heats has lower weight than redoing the GA for optimality from where the Turnaround first occurred. This however would mean that the active heats would be lost, and since the cost and time of redoing the active heats outweighs having a new optimal schedule from time of first turnaround, it is therefore not done. Note the architecture for the Production Unit Rescheduling algorithm is made in such a way that if at time of failure there are no active heats, then the GA will still be performed for all inactive heats.

4.5. POWER LEVEL FLUCTATIONS

In a melt shop, the process of creating molten steel is a highly energy intensive process. About 7% of the world energy consumption is taken by the Iron and Steel Industry [15]. This being said, energy cost and availability are a matter of concern for corporate survival. Many steel manufactures must consider the price and availability of power when creating the time schedule. Power level restrictions are a reality and must be incorporated into the scheduler when defining heat times for machines.

The heat times for the various machines in the melt shop on a regular power consumption day are defined in Table 4.3 found in Section 4.1.3.3 above. These values are based on the model parameters given in Table 3.3 and 3.4 of Chapter 3. The details of using the heat_time_table.m in the GA are explained in Hadjimichael [7].

For understanding more about the types of issues of energy monitoring, modeling and optimization, the reader should refer to [15]. This is out of the scope of MES for this thesis, but this area could overlap into MES for interfacing the data needed about adequate supply of energy for meeting the production schedules. Also data from MES will have overlay to the Energy Management System (EMS), where all the energy management and control decisions are made. The EMS needs to receive on-line information on the energy scenario of the plant from field sensors, therefore, the operation status of machines would have to be communicated to the EMS and tentative schedules outputted by the scheduler of the MES would need to be given to the EMS for calculating energy consumption.

4.6. STATEFLOW AND MATLAB GA INTERFACING

The A412 module provides the *Production Unit Routing* output signal. This signal contains the information needed by the machines (EAFs and LFs) to define the routing time needed (heat times) for the day. This means based on the defined schedule for the day, determine the time needed for each heat at each machine, which defines the heating time on a machine. Since the machines in STATEFLOW run on a global clock, the times given to the machines must be in reference to this clock. The ga2sf.m algorithm determines these times and provides them with the schedule to the plant machines. This is needed to maintain priority to the defined sequence. The ELAstatus.m algorithm gives the time EAFA and LF A should process each heat for the scheduled day. The ELBstatus.m and ELCstatus.m algorithms do the same for EAF and LF B and C, respectively.

More detail on this can be found in Hadjimichael [7].

4.7. CONCLUDING REMARKS ABOUT APPLICATION

Our process began with a planned or sequenced list of heat orders, methods to schedule those into heat stations, control of heat assignment based on capacity to machines, and management of power available. Along with data collection to keep the system current and a way to handle exceptions, we have the ability to execute the manufacturing plan-truly a Manufacturing Execution System.

Chapter 5

5. Algorithm Results

In this chapter the results are included to illustrate how the algorithms perform given different plant configurations and situations. Only the algorithms of the GA and Production Unit Rescheduling will be looked at here.

5.1. PERFORMANCE OF THE GENETIC ALGORITHM

In the MATLAB file “launch_parameters.m” the following GA parameters were set:

```
GA Paramters
inpt_pop_size = 24;
Penalty = 400;
alpha = 0.1;
beta = 40;
cross_prob = 0.80;
num_crosses = 1;
mut_prob = 0.01;
max_gen = 75;
ELF_data;
elitism = 1; %use elitism

PLANT Parameters
first_shift = 1; (morning shift)
Heats = 6;
ELA_DOWN = 0; %not down
ELB_DOWN = 0; %not down
ELC_DOWN = 0; %not down
```

See Hadjimichael [7] for instructions on how to run the GA.

The restriction of heats allowed for EAF A, is set through the ELA_heat_not_allowed variable found in the ELF_data.m file. In Table 3.1 of Chapter 3 it can be calculated that heats 1 and 4 require a capacity of 125 tons. In Table 3.3 of Chapter 3, the capacity of EAF A is 120 tons. Therefore heats 1 and 4 can not be handled by EAF A. This is set by defining a 1 in positions 1 and 4, corresponding to the heats that can not be handled by

EAFA. Therefore the vector ELA_heat_not_allowed = [1, 0, 0, 1, 0, 0] is defined in ELA_data.m file.

Table 5.1 below defines the new heat time values for the various machines (this can be modified in the heat_time_table.m file). Here the LFA, B, and C times have been modified from that found in Table 4.3 (Section 4.1.3.3).

Table 5.1: Heat Time Values for Machines

	Heat 1	Heat 2	Heat 3	Heat 4	Heat 5	Heat 6
EAF A	0	50	50	0	50	50
EAF B	60	60	60	60	60	60
EAF C	60	60	60	60	60	60
LF A	30	30	30	30	30	30
LF B	20	20	20	20	20	20
LF C	20	20	20	20	20	20
CC	62.5	80	60	62.5	80	60

The average values for the original and final populations are compared in Table 5.2 below:

Table 5.2: Comparison of Original and Final Populations

Population	Penalty	Makespan (min)	Cost ($)	Average Fitness (F(x))
Original	**33.3**	**485.4**	**6187.5**	**0.010820**
Final	**0**	**475**	**2000**	**0.011905**

Note: Raw data can be found in first_pop1 and last_pop1 of Appendix H.

From Table 5.2, it is observed that the average penalty, makespan and cost of the population decrease substantially over time. It is not surprising, that the average fitness of the final population has increased.

Figure 5.1 below shows the average fitness of the population. The convergence to 0.011905 occurs at 13 generations, however just after 40 there is slight drop, this is attributed to the changes in *Figure 5.2* and *Figure 5.3*, where the makespan and penalty spike up before converging again to 475 and 0 values, respectively. The occurrence of this situation is due to the dynamic nature of the population. Meaning through mutations, crossovers and pure roulette selection there is possibilities that the fittest member might leave the population since it has either changed or has not been picked. To avoid this from happening elitism must be incorporated into the GA. Elitism is where the fittest member is automatically carried to the next population without any crossover or mutation (Section 5.1.2).

Figure 5.3 and *Figure 5.4* (below) show the penalty and cost, each of which converges to values of 0 and $2000, respectively.

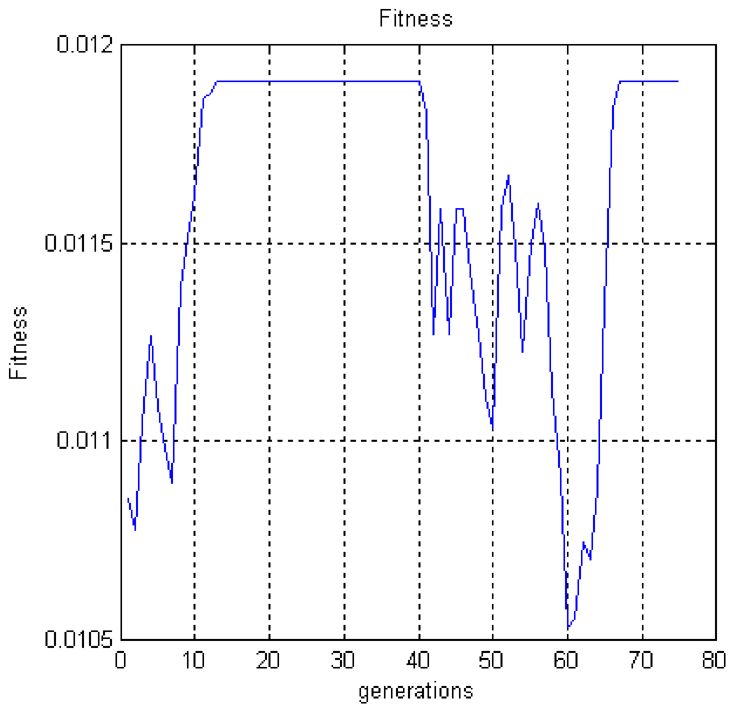


Figure 5.1: *Convergence of Fitness Function*

Note in *Figure 5.1* above the average fitness function is being shown.

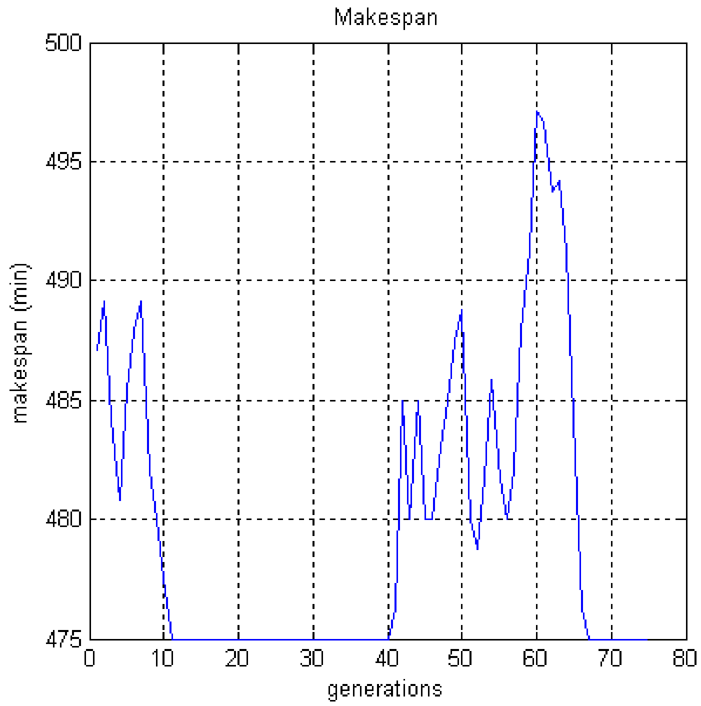


Figure 5.2: *Convergence of Makespan*

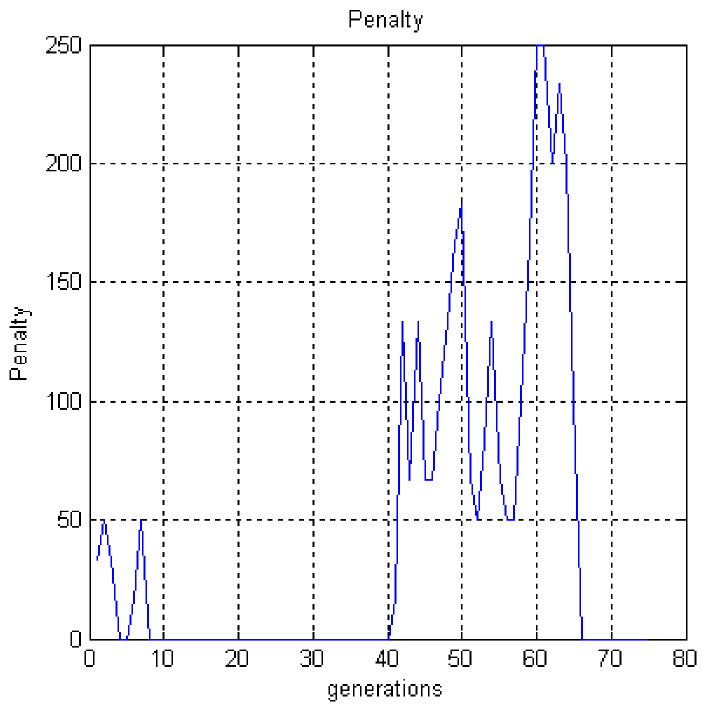


Figure 5.3: *Convergence of Penalty*

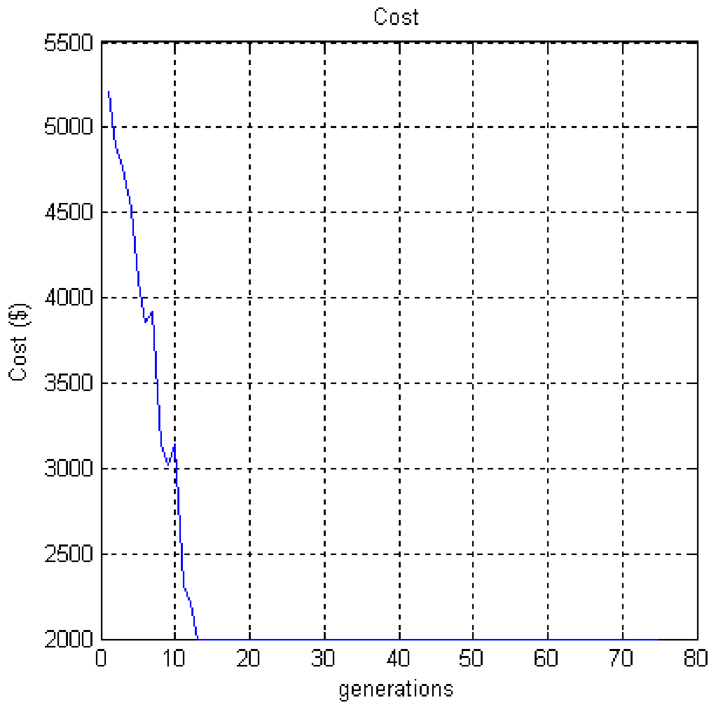


Figure 5.4: *Convergence of Cost*

Note the minimum cost for the solution is 0. However, this is not realized in *Figure 5.4* above due to the compromise between the cost and the makespan. Had the weight on cost been greater than the weight on makespan, through choosing different alpha and beta values in the objective function (as discussed in Section 4.1.3.3), then cost would have reached zero but at the expense of a longer makespan.

During the last generation, the majority of the chromosomes represented the schedule:

6	1	3	1	2	2	5	1	4	2	1	2/3

Note the schedules below also give the same fitness:

6	1	3	3	2	3	5	2	1	3	4	2
3	1	5	2	2	3	6	2	4	3	1	2
3	1	5	2	2	1	6	1	1	2	4	3

5.1.1. Analyzing the Schedule of the Genetic Algorithm

The optimal schedule is outlined in Table 5.3 below and uses the heat times defined in Table 5.1. The start and stop times of each machine is generated by the A413 module by running its Time Translator Scheduling Algorithm.

Table 5.3: Optimal Schedule: **61 31 22 51 42 12**

MACHINE	TIME
(min)	50 100 150
EAF A	**6** **3** **5**
(min)	60 200
EAF B	**2** **4** **1**
(min)	
EAF C	
(min)	50 80 100 140 150 280
LF A	**6** **3** **5**
(min)	60 200 360 422.5
LF B	**2** **4** **1**
(min)	
LF C	
(min)	80 140 200 280 360 422.5 485
CASTER	**6** **3** **2** **5** **4** **1**

It is important to understand how the melt shop system works when integrated together. Specifically the heat times defined in Table 5.1 above are not exactly the times seen in the chart (Table 5.3). This is due to the restrictions caused by downstream processes (bottlenecking). The CC will always follow the values in Table 5.1 since this is a continuous process dictated by fluid flow, a physical property. However the times on the EAFs and LFs is not always fixed, since the LF must wait for the CC to accept its heat, and the EAF must wait for its LF to be freed up before passing along the heat. For illustration, heat 5 is accepted for LFA after heat 3 has been given to the CC, it is assumed that both transitions occur simultaneously with no delay. Heat 4 in LFB is not accepted until its previous heat 2 has been given to the Caster. This causes EAFB to hold heat 4 for an extra 80 minutes until heat 2 is taken away from LFB at time 200. This causes the times on the machines to be longer than the required heating time. It is assumed that this does not affect the physical properties since the heats are fully melted and simply maintaining that state.

5.1.2. Using Elitism in the Genetic Algorithm

The initial population will affect the time of convergence if elitism is incorporated. There are cases when the GA reaches a stagnant population as seen in *Figure 5.1* and *Figure 5.2* above at intervals 13-40 generations. There are times when the fitness function does not settle to its optimal of 0.011905. The momentum that is carried by passing the fittest member to each population causes this. Since roulette wheel is the selection method and since the fittest member has the most chance of being chosen, the population can quickly converge to this fitness value, even if it is not the optimal one. If regenerations goes on for long enough a possibility of a mutation could diverge the population out of a local maximum to the global maximum (fitness function =0.011905).

In analyzing the GA with elitism it is found that the best the fitness can do is only as good as its initial populations best, unless a crossover or mutation can find a better solution than the original best. After a few attempts the best solution found was:

6	1	5	1	2	1	3	2	4	2	1	2

see last_pop1.xls in Appendix H.

This still gives the same final results for cost, makespan, penalty and fitness as the optimal solution found above (Section 5.1). *Figure 5.5* below is included to show how the average fitness value converges to the best fitness value. Note the best fitness value (dotted line) was found originally in the first population.

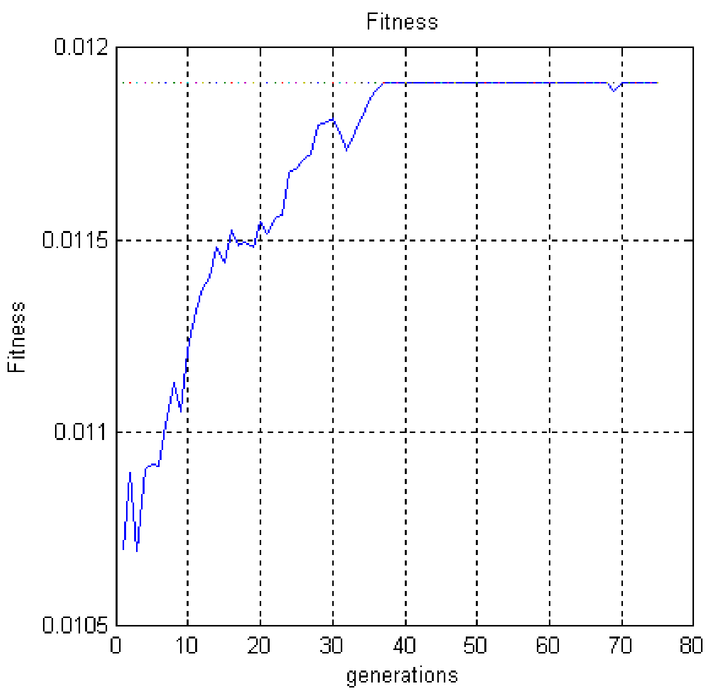


Figure 5.5: *Fitness with Elitism*

A problem without using elitism is seen in *Figure 5.6* below where the average fitness function diminishing. However if the fittest chromosome from all generations is stored, then the optimal solution will not be lost.

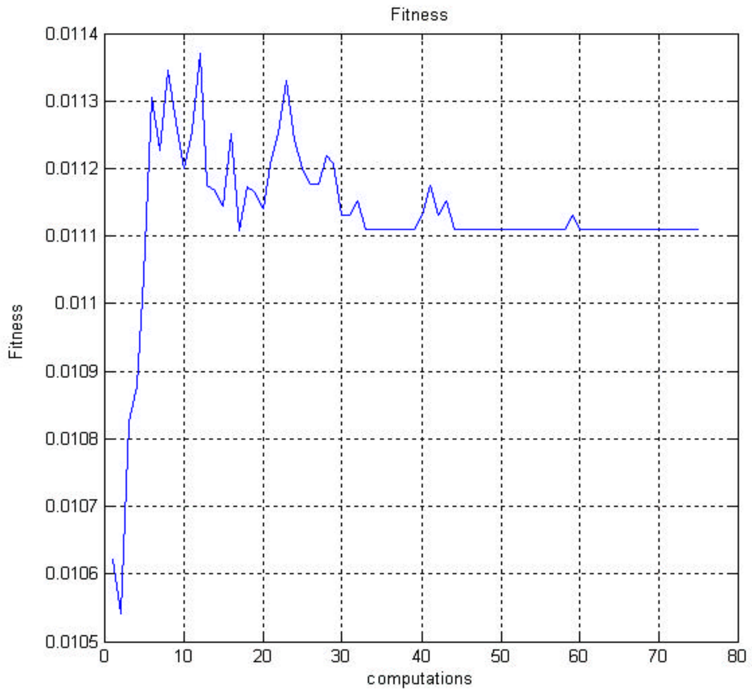


Figure 5.6: *Problem in the GA without Elitism*

5.1.3. Concluding Remarks about the GA

It is concluded that using a larger population will give a better result. With an initial population size of 124 the optimal solution is nearly almost found. However, due to the very large population the convergence to the optimal value is not seen in all the chromosomes with the number of generations used. This is interpreted as not allowing evolution to take place for long enough so that all the population has had time to take on the same chromosomes as the fittest member. Using a larger population size and more generations means more computation time. Therefore, it is better to run the algorithm with a smaller population size and keep track of the fittest member for all generations, this chromosome will have the best solution regardless of what the rest of the population chromosomes evolve into.

5.2. THE PERFORMANCE OF PRODUCTION UNIT RESCHEDULING ALGORITHM

This section will use the melt shop status values defined in Section 4.4.2 given the instantaneous failure of machine EAF B at a time of 110 minutes. Using this data produced by the A6 module's Decode Decision Parameters Algorithm the Production Unit Rescheduling Algorithm will output the following results.

3	4	2	5	5	1	1	3	4	3

The above schedule is interpreted by first noting if a zero exists as a first element. This means Turnaround can not be avoided with this failure. Here this is not the case, since a zero is not present in the first element. Therefore, this schedule says the active LF heats are dealt with by casting them in the order given in the schedule. First cast heat 3 in LF A (machine 4), then cast heat 2 from LF B (machine 5). The next active heats are found in the EAFs. In this case only heat 5 in EAF A (machine 1) was active at time of failure and should be cast next following heat 2. After this the inactive heats are processed heat 1 then heat 4 in EAFC (machine 3), this is coupled to LFC (machine 6). Notice heat 4, the failed heat, is completely reheated by EAF C from the beginning.

The Production Unit Rescheduling Algorithm also outputs the following charts; these are presented in *Figure 5.7* below.

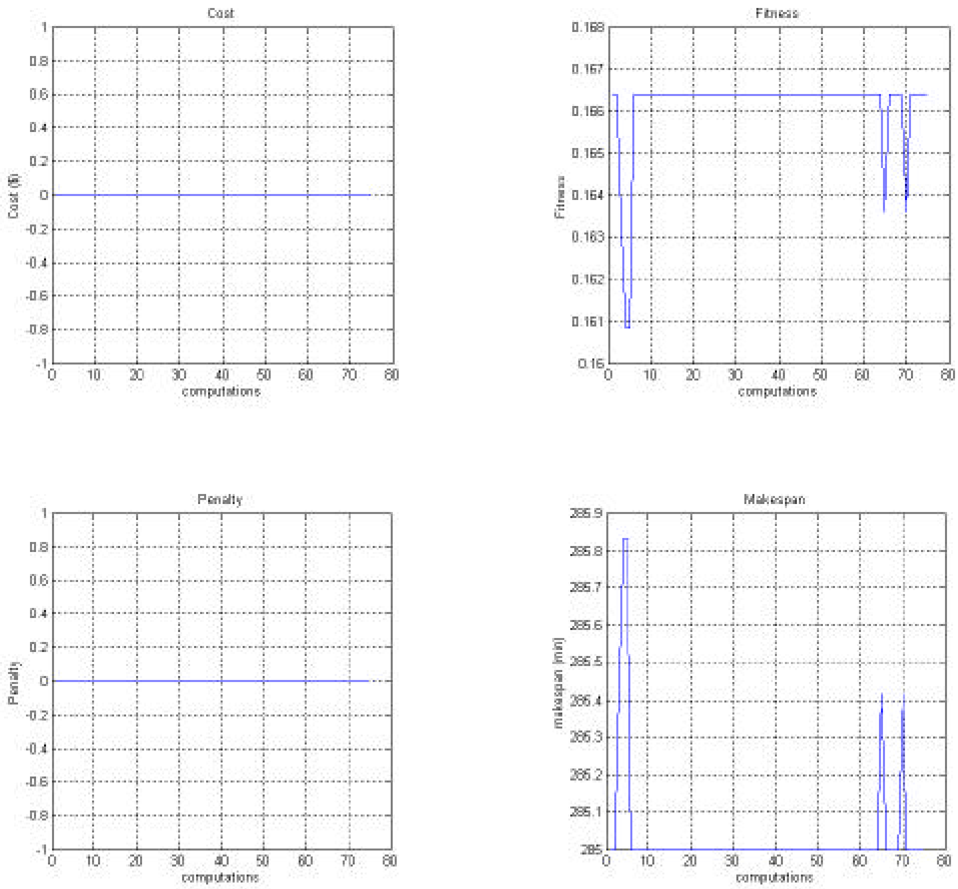


Figure 5.7: *GA Outputs using Production Unit Rescheduling Algorithm*

The four charts above are the performance of the modified GA, used in solving the sequence for the inactive heats starting with the first heat fixed. The first heat fixed is taken as the last cast from the active heats, for this case it will be heat 5 processed through EAF A and LFA, note heat 4 and 1 are the inactive heats. For a more complete interpretation and explanation of all the stages of this algorithm and results obtained along with the modifications for this new GA please refer to Hadjimichael [7].

Chapter 6

6. Final Remarks and Conclusions

6.1. RECOMMENDATIONS AND FUTURE WORK

The standard general modules found in the Develop Operation Sequence and Detailed Schedule module, these being the Develop Production Activity, Define Production Unit, Scheduling Production Unit and Schedule Resources have been worked on and can be applied to a real world situation. We have shown that they can improve the manufacturing performance as far as optimizing scheduling and handling power level and resource failure disturbances goes.

The MES designed for the melt-shop works in theory. However, an improvement may be incorporated to alter the Decision Tree Algorithm by using the entire final population generated and applying a checking function that stores the alternative schedules found in the final population. This data contains alternative machines for certain heats in the sequence, but still maintains the same fitness function. This information can be included as alternatives depending on the plant status at failure for a more informative rescheduling of the machine failure disturbance problem.

Due to the ever increasing pace of business, customers can change their priorities on an almost daily basis. The effect is that manufacturing priorities are shifted and production is lost through extended changeovers between processes. The ability to handle these event driven customers and quality disturbances is something that is lacking in the existing specific case study MES of this thesis.

It is recommended that customer disturbance handling be incorporated into the A41 Develop Operation Sequence and Detailed Schedule for Heats module. Customer disturbances occur on a day-to-day basis as purchasing requirements are altered. The idea is to reach a compromise between the customers' requirements and the economics of production. Here the MES may be required to change the daily schedule on the fly, if a decision is made by management. The MES must consider where in the schedule the new heat order can fit to maintain a continuous caster while using available resources, and minimizing the cost of transition steel. The shortest makespan should also be considered in this optimization problem, since adding a new heat to the schedule means adding more time to process for the day schedule.

It is also recommended that quality disturbance handling be incorporated into the A433 Manage Heat Process module and A434 Manage Heat Quality. The A433 module provides the status at failure using the A6 module's Decode Decision Parameters Algorithm; however the A434 module needs an algorithm for QA Process Adjustments. Quality disturbances occur when chemical problems arise mainly from not meeting the correct specification. This could be anything from too much carbon in the steel to the presence of an impurity such as copper. This information is provided when the lab QA system has tested the steel, usually some time before entering the casting stage. "Therefore, the immediate response is to keep running but adjust the operating parameters. There are a number of well defined and documented actions that can be taken. These range from adjusting the levels of additives to resolve a chemical imbalance to changing the pressure in water-cooling sprays at the caster, for a physical problem. [19]" The MES should have the capability to take the proper actions. In some cases the diagnostics are not so simple and require intelligent re-scheduling and optimization. If the heat is downgraded to a different quality then the MES must look at its future schedule and see where the heat not passed can be done and if the downgraded heat can replace a similar heat that was up-coming in the schedule for the day.

6.2. CONCLUSIONS

Manufacturing Execution Systems (MES) are needed to provide a software solution that can improve all areas of the manufacturing process including better performance, higher consistencies and quicker response to adapting needs from customers, suppliers and internal processes. The traditional tools such as Enterprise Resource Planning (ERP), supply chain, customer relationship, and product life-cycle management systems are not sufficient to achieve the efficiencies required by today's low margin, quick response and scalable market. These systems simply author what MES should execute while still allowing MES to maintain and manage the production orders, and provide intelligence for optimizing operations with fast, accurate and transparent data in near real-time.

To meet its goal, MES must conform to becoming a standard reusable application. To achieve an integrated flexible manufacturing environment at affordable costs it is evident that the work in the area has to be synchronized and data information must be well defined and organized. This research starts with a standard (National Institute of Standard and Technology, NIST) framework that defines a set of foundation modules for manufacturing execution with a set of interfacing data needed to properly define how MES fits in the overall architecture of plant information and functions. The information given to the MES by the authoring systems and status data from the plant floor systems is clearly defined to help illustrate the general role of the MES functions and interfacing data.

By taking this framework the research properly applies the general application to a specific steel melt-shop plant case study. This is done to clarify the aim of the research in enabling the MES to feedback set points and control operations to the plant floor systems and provide scalability for growing the functionality of the MES. Both aims are accomplished by starting with the standard framework and adding specific intelligence to some of the modules in order to improve the MES control functionality.

The control functionality portion of the research was intended to provide first a generic scheduling algorithm. This was done using a Genetic Algorithm (GA), and made specific to the needs of the steel melt-shop by defining the chromosomes to represent the heat schedule. The fitness function was also defined specifically to optimize for lowest cost in transitioning steel produced and shortest makespan for the heats defined to be done for the day. The second functionality for improving the MES's near real-time plant floor control operations focused on the capability for handling disturbances. The disturbances handled by the MES focused on power level fluctuations and plant furnaces failing. The machine failure disturbance was handled by taking the outputted GA's schedule and assigning specific working time durations for the different resources in the plant using the

Time Translating Algorithm. That way the information of the active heats and machines could be known at anytime the failure occurred. This was derived from the MES functionality of having real-time plant floor status data. Power level fluctuations are handled within the GA by defining the times that will be needed for heating based on the power availability for the scheduling day.

The MES's capabilities in handling these disturbances were defined within the MES's framework through labeling the generic data presented in Chapter 2 to specific data found in Chapter 3. The specific data includes inputs, outputs and control to certain modules. The specific modules that use this data have built-in algorithms (mechanism data) designed to handle decoding the melt-shop's current status (Decode Decision Parameters Algorithm) and providing this information to the Production Unit Rescheduling Algorithm for re-optimizing the heats. This is an attempt to try and avoid turnaround if possible and provide a new resource allocation for the heats based on what is available, while still trying to optimize for lowest cost for transition steel produced during casting.

The second aim for allowing scalability and improvements to the MES's abilities of providing feedback through control algorithms is supportable through the standard framework. In fact some of the improvement ideas that can be incorporated are furthered by the recommendations of applying customer and quality disturbance handling capability.

7. REFERENCES

[1] Bagchi Tapan P., *Multi-Objective Scheduling by Genetic Algorithms*. Boston, Massachusetts: Kluwar Academic Publishers, (1999).

[2] Barkmeyer E., ed., NISTIR 5939 (Gaithersburg, MD: NIST, 1996), *SIMA Reference Architecture Part 1: Activity Models*.

[3] Feng S, NISTIR 5808 (Gaithersburg, MD: NIST, 1996), A Machining Process Planning Activity Model for Systems Integration.

[4] Feng S, Manufacturing Planning and Execution Objects Foundation Interfaces, *Journal of Manufacturing Systems*, Vol. 19, No.1, 2000, pp. 1-17

[5] Fisher-Rosemount Systems, Inc. *"Emerson Process Management Manufacturing Execution Systems Capabilities."* White Paper, January 2002. 1996—2003

[6] Gen Mitsuo. *Genetic algorithms and engineering optimization.* Chichester, England: Wiley, (2000).

[7] Hadjimichael Basil, *Manufacturing Execution Systems for Integration and Intelligence: Guidelines and Details for Executing the Intelligent MES Algorithms* (TR-CIM-04.03). May 2004

[8] Huang C.Y, Distributed manufacturing execution systems: A workflow perspective, *Journal of Intelligent Manufacturing*, 12, pp 485-497, 2002.

[9] Institution of Electrical Engineers. *Genetic algorithms in engineering systems.* London, England, (1997).

[10] Interwave Technology, *MES Enterprise summary web page* (http://www.interwavetech.com/r_about_cmes.asp)

[11] Lenstra, J.K., A.H.G. Rinnooy Kan, P. Bruckner, Complexity of Machine Scheduling Problems, *Annals of Discrete Mathematics*, Vol. 7, 1977, pp. 343-362.

[12] McClellan Michael, *Applying Manufacturing Execution Systems*. St. Lucie Press: Boca Raton, FL, (1997), pp 179.

[13] McClellan Michael, *Evolving to the Enterprise Production System (EPS).* MES Solutions Inc.

[14] *"MES Explained: A High Level Vision"*, MESA Organization. September 1997

[15] Rominus Valsalam S., Muralidharan V. and Krishnan N. Implementation of Energy Management Systems for an Integrated Steel Plant, *IEEE Catalogue* No: 98EX137. 1998.

[16] S95.01 (2000). ANSI/ISA-95.00.01-2000 Enterprise-Control System Integration Part 1: Models and Terminology.

[17] SP88 (1994). SP88 MES Task Force Europe – Position Document v2. 7 February, 1994

[18] TATA CONSULTANCY SERVICES, *MANUFACTURING EXECUTION SYSTEMS: A Concept*. February 2002.

[19] Valckenaers Paul, *ESPRIT LTR 22728,* Manufacturing control systems capable of managing production change and disturbances (MASCADA) – Analysis and evaluation of change and disturbances in industrial plants. Katholieke Universiteit Leuven, 2000.

[20] 21 CFR Part 11 (2000). 21 Code of Federal Regulations Electronic Records; Electronic Signatures. Issued: March, 2000

APPENDICES

Appendix A: Terminology

Appendix B: Module Data Breakdown

Appendix C: STATEFLOW Model and Detailed Description of Input Queue

Appendix D: STATEFLOW Model and Detailed Description of EAF-LF

Appendix E: STATEFLOW Model and Detailed Description of Caster

Appendix F: Detailed Case Study Data for A4 Modules

Appendix G: Pseudo Code and Architecture for the Production Unit Rescheduling Algorithm

Appendix H: Raw Data for first_pop and last_pop generated by the GA

APPENDIX A

TERMINOLOGY

AGVS. Abbreviation for Automatic Guided Vehicle System.

AS/RS. Acronym for Automatic Storage/Retrieval Systems, a rack storage system with automatic loading (storage) and unloading (retrieval).

CAM. Computer-Aided Manufacturing

CIM. Computer-Integrated Manufacturing

COM. Component Object Model

DCOM. Distributed Component Object Model

EPS. Enterprise Production System

ERP. Enterprise Resource Planning

LIMS. Laboratory Information Systems

MRP. Material Requirements Planning

PLC. Programmable Logic Controllers

QA. Quality Assurance

SCADA. Supervisory Control and Data Acquisition

WIP. Work-In-Process

The following terms are used in activity and ICOM definitions.

Batch:
One or more lots are identified as one group that is treated by some processes as a unit. After the batch is treated, the batch can be dissolved and individual lots are routed separately.

Job:
A batch or lot that is scheduled to be released for production. (A more detailed description is given in Ref. 4.)

Lot/Load:
One or more parts/components in a group that travel through the production process as a unit. Lot and load are synonymous. A lot has a release date and time, a due date/time, and an actual finished date/time.

Production Unit:
In a manufacturing facility (factory or shop), a production unit can be a batch, lot, or single part. Each production unit has a unique identification, maintained by the system.

Resources:
Skills and physical entities that are required for performing production activities. Examples are stock materials, equipment (machining centers, automatically guided vehicle, robots, measuring machines, and so on), tools (fixtures, cutters, adapters, hand tools, gages, computer programs, tapes, and more), labor skills, and energy.

Input, Control, Output, and Mechanism (ICOM) Data (information flow) Definitions

Actual Production Steps:
An actual production step is a detailed instruction to equipment or workers to execute production activity, such as load tools to a machine, start a milling cycle, drill a hole, check the actual dimension of a feature, and so on.

CAD/CAPP/CAM/PDM:
The computer systems for product design and modeling (CAD), engineering, computer aided process planning (CAPP), machining numerical control programming (CAM), product data/process management (PDM). A system includes interface, data repository, and data process software.

Controller:
Usually hybrid hardware/software systems. Examples are distributed control systems (DCS), programmable logic controllers (PLC), distributed numerical control (DNC), and supervisory control and data acquisition (SCADA) systems.

Cost Report:
A report on the manufacturing costs of producing a part. It contains the costs of material, labor, usage of equipment, and so on.

Data Analyzer:
A software component that provides up-to-the-minute report of actual manufacturing operations results along with the comparison to past history and business expectations. The results include such measurements as resource utilization, resource availability, product cycle time, conformance to schedule, and performance to standards.

Data Collection Methods:
The use of data collectors to obtain information on workpieces, timing, personnel, lots, and other critical entities for production management in a timely manner.

Data Collector:
A collection of devices with control software that are linked to factory-floor production equipment to gather data either manually or automatically from the manufacturing facility in an up-to-the-minute timeframe.

Delivery Schedule:
The schedule of delivery of purchased resources.

Design Knowledge:
The information (rules, logic, or examples) that a human designer brings to bear on design problems, including design techniques and implementation techniques. Many different types

of design knowledge are used in different design activities, such as decomposition knowledge, assignment knowledge, consolidation knowledge, and optimization knowledge.

Detailed Schedule:
A plan that specifies starting time and finished time of each production unit in the queue locally to an area in the manufacturing facility, such as a workcell, a workstation, or a machine.

Document Controller:
A mechanism, usually software, that controls records and forms that support product life cycle activities, such as manuals, drawings, computer models, procedures, recipes, programs, engineering change orders (ECO), shift-to-shift communication records, and so on.

ERP/MRPII:
Enterprise Resource Planning (ERP) and Manufacturing Resource Planning (MRP) II are the systems that provide financial, order management, productions and materials planning, and related functions. The modern systems focus on global planning, business processes, and execution across the whole enterprise (intra-enterprise systems), with an accrued recent importance of aspects like supply chain planning and the whole supply chain management aspects and extending to include the whole inter-enterprise supply chain.

Equipment Operation Instructions:
Specific operation steps or recipes that are used to control machine movement, such as machining, welding, assembly, material movement, and so on.

Labor Records:
A labor record is a piece of data that records the time, attendance, tasks performed, tasks assigned, skill level, and certificates of a worker.

List of Purchased Resources:
A list of resources that are purchased from suppliers. For each resource item, the list contains resource number, description, purchased date, quantity, cost, and other related information that is company specific.

Machine Tools:
A machine tool is a machine with accessories that provides the capability of machining, such as milling, turning, drilling, and grinding.

Manufacturing Bill of Material (BOM):
A list of parts that are scheduled to be manufactured in the factory. For each part, the BOM contains part number, description, quantity description, and so on. Manufacturing BOM is the manufacturing version of product structure and part list in a corresponding production system, known as "as-built configuration," which support manufacturing engineers to consider additional information when planning how to manufacture the product, such as manufacturing capabilities, physical assembly possibilities, and the availability of parts

Manufacturing Execution System (MES):
A production activity management system that initiates, guides, responds to, and reports on production activities on-line and in real time to production management people. The system aids the Execute Manufacturing Orders activity.

Manufacturing Knowledge:
The information (rules, logic, examples) that a manufacturing engineer brings to bear on manufacturing engineering problems, including production techniques and implementation techniques. Many different types of manufacturing knowledge are used in different manufacturing activities, such as decomposition knowledge, assignment knowledge, consolidation knowledge, and optimization knowledge, which are used in process planning, resource planning, production planning, and scheduling.

Manufacturing Orders:
Instructions that are sent to factories to start jobs to fulfill customer orders. The starting dates are specified in the manufacturing order according to the production plan and the master production schedule.

Market/Customer Requirements:
A list of customer needs based on market studies, detailed evaluation of the competition, and review of all available literature. It includes the description on product performance, appearance, delivery time, target price, volume, safety, and environment.

Master Production Schedule:
A plan that specifies starting time and finishing time of each job in the job queue that are for producing products required by customers. The plan contains job IDs, starting dates, and due dates.

Operation Sequence:
A set of step-by-step instructions that specify how to perform tasks to process a workpiece in a local area, such as a machine, a workstation, a workcell.

Personnel Assignments:
A list of workers who are assigned to perform specific operations in the production plan. Each worker is assigned to perform or monitor one or more operations, usually, with due dates.

Personnel Data:
A record of personnel assigned to perform production activities. It provides work hours, on-station time, skills, certificates, and so on.

Personnel Reassignments:
Requests to reassign workers to new tasks.

Personnel Tracking Component:
A software component in a MES that aids users to track workers in a manufacturing facility.

Planning Policies:
Rules, regulations, strategies to plan business, engineering, and production activities.

Process Adjustments:
Requests from operators to process planners to modify the process plan or to adjust certain predefined parameters to improve process performance.

Process Change Request:
Feedback from factory-floor production requesting changes to process plan when some problems in the process plan were found. Changes can be process parameter changes, tool changes, setup changes, and so on.

Process Management Component:
A software component in a MES that aids users to manage processes.

Process Performance:
Measures of how good parts, components, and products are produced by a process. Process performance include production rate, product quality, and process capability.

Process Plan:
A plan that specifies operation sequences, equipment, and process parameters for manufacturing a product.

Process Status:
A report of the conditions of a process being monitored. The report includes alarms, process changes or shifts, workpiece throughput, and so on.

Product Bill of Material (BOM):
An index to illustrate the structure and detailed information of product, component and part, known as "as-designed configuration" or "Engineering BOM." It includes the item number of letter, the part number, the quantity needed in the assembly, the name or description of the component, the material from which the component is made, and the source of the component.

Product Delivery:
Move finished products to customers who requested the products.

Product/Process Statistics:
Measurements and statistical analyses of process performance and quantities and the quality of products.

Product Genealogy Component:
One of the components in a MES that provides the visibility to where work is at all times and its disposition. Product genealogy information may include who worked on the product, components materials by supplier, lot, serial number, current production conditions, and any alarms, rework, or other exceptions related to the product. This information provides traceability of each part and component.

Product Inventory:
The inventory information on a product. The information is updated when finished products are sent to storage.

Product Order:
Quantities of parts or products to be produced, usually with nominal delivery dates, as specified by enterprise sources external to the manufacturing facility.

Product Status:
Current conditions of a product, including the quantity of the product made, checked against the schedule, measurement and test results, and any exceptional process conditions that occurred.

Product Tracking Component:
A software component in a MES that aids users to track products in a manufacturing facility.

Production Activity Planning Component:
A software component in a MES that aids users to plan production activities.

Production Orders:
Instructions that are sent to a local area of a factory to start processing a production unit with the starting date and time and the ending date and time.

Production Statistics:
Measurements and statistical analyses of the production process and the quantity and quality of products being produced.

Production Status:
A report on the state of all scheduled operations and production units. This also includes the information on resources, process setup, job schedule, and material routing.

Production Unit Definition:
Definition of a lot or a batch. It includes an ID, number of workpieces, and the descriptions of the workpieces. Each workpiece may have a serial number. In the product record, workpiece ID and production unit ID are associated.

Production Unit Routing:
A plan that specifies the traveling route of a production unit in a manufacturing facility. The plan also specifies stops for processing and queuing.

Production Unit Scheduling Component:
A software component in a MES that aids users to schedule production unit to be processed locally in a manufacturing facility.

Production Unit Status:
A snapshot of a product unit being processed. The status includes the quantity of finished product, scrap rate, rework rate, product measurements analyses, and a check of the status with the master production schedule.

Purchase Orders:
A purchase order is an instruction to buy certain resources (material, parts, components, tools, machines, and so on) from a supplier.

Quality Statistics:
The statistical data pertinent to the quality of the product measured in-process or post-process based on the design specifications.

Released Units:
Production units that are released for processing in the manufacturing facility.

Resource Availability:
A report on whether needed resources are available for production during specified time periods.

Resource Data:
The data that indicate the condition of a resource based on inspection or measurement analysis.

Resource Records:
A resource record is a piece of information that indicates where the resource is located and who is using it for which operations on which production unit for how long. If it is a piece of equipment, the record should also show whether it is functional.

Resource Release Orders:
A resource release order is an instruction that requests to release resources from storage or from current user to a new user.

Resource Requirements:
A list of resources required supporting production jobs.

Resource Schedule:
A plan of control resource availability and allocation. It specifies a group of resources that each resource is assigned to which operation or transferred from one place to another in a specific time period. Only resources that are used/shared by multiple workcells or workstations are on the resource schedule.

Resource Scheduling Component:
A software component in a MES that aids users to schedule the release of resources to workcells, workstations, and/or machines.

Resource Statistics:
The statistical data pertinent to the state of resources inspected or measured in-process or post-process.

Resource Status:
A snapshot of a resource used in production. The conditions, location, and service time of the resource are reported. If it needs maintenance, replacement, or disposition, the resource is marked accordingly.

Resource Tracking Component:
A software component in a MES that aids users to track resources being used in a manufacturing facility.

Service Delivery:
The delivery of post-sale service to customers.

Specifications:
Sets of description of standard engineering, manufacturing, and business practices that guide and control the product development process.

Statistical Quality Control (SQC):
A software component in a MES that aids users to analyze and control product quality and to monitor process capability and shift.

Sale and Service Management (SSM):
A mechanism that supports sales force automation, product configurations, service quoting, product returns, and post-sale service.

Standard Part Library:
An information library or database that contains standard parts. A standard part is a member of a class of parts that has a generic function and is manufactured routinely without reference to its use in any particular product. Examples of standard parts are screws, bolts, rivets, jar tops, buttons, most beams, gears, springs, and washers.

Supply Chain Management (SCM):
A mechanism that aids users to manage the supply of resources, including forecasting, distribution and logistics, transportation management, electronic commerce, and advanced planning.

Task Assignments:
Records of assigning tasks with due dates to workers.

Tool, Equipment Maintenance Order:
An instruction indicating specific tools, machines, or devices that need maintenance before performing any production activities.

Tool and Equipment Status:
The condition of all tools and equipment. Condition includes the usage load, wear and tear, broken status, and the forecasted life span.

Tool Usage Instructions:
Instructions that guide users to properly use tools in production.

Tooling Design:
Specification of the form, function, and material of a tool (for example, cutter, fixture, and probe). There are two major subtypes of tooling design: (1) tool assembly design that specifies the assembly of a tool or fixture from standard components, and (2) special tool design that must be fabricated.

Updated Document:
Document that is modified to include new information.

Workpiece Measurements:
The assessment and comparison of workpiece geometry, dimension, tolerance, and functions for the conformance to the design attributes.

APPENDIX B

MODULE DATA BREAKDOWN

Table B.1: A41 Module Data

Mechanism			
Manufacturing Execution System (MES): A production activity management system that initiates, guides, responds to, and reports on production activities on-line and in real time to production management people. The system aids the Execute Manufacturing Orders activity.			
Input			
***	*Process Status*	see Table 2.1	Inputted from: A6 (feedback)
* / **	*Master Production Schedule*	see Table 2.1	Inputted from: A2
*	*Specifications*	see Table 2.1	Inputted from: A6 (feedback)
*	*Resource Requirements*	see Table 2.1	Inputted from: A2
* / **	*Manufacturing Orders*	see Table 2.1	Inputted from: A2
***	*Updated Document*	Document that is modified to include new information.	Inputted from: A44 (feedback)
**	*Resource Availability*	A report on whether needed resources are available for production during specified time periods.	Inputted from: A44 (feedback)
Output			
**	*Production Unit Routing*	A plan that specifies the traveling route of a production unit in a manufacturing facility. The plan also specifies stops for processing and queuing.	Outputted to: A43-Control
* / **	*Operation Sequence*	A set of step-by-step instructions that specify how to perform tasks to process a workpiece in a local area, such as a machine, a workstation, a workcell.	Outputted to: A412, A414, A414-Control, A42, A43-Control
**	*Detailed Schedule*	A plan that specifies starting time and finished time of each production unit in the queue locally to an area in the manufacturing facility, such as a workcell, a workstation, or a machine.	Outputted to: A42-Control, A43-Control
**	*Production Orders*	Instructions that are sent to a local area of a factory to start processing a production unit with the starting date	Outputted to: A42, A43-Control

		and time and the ending date and time.	
*	*Personnel Assignment*	A list of workers who are assigned to perform specific operations in the production plan. Each worker is assigned to perform or monitor one or more operations, usually, with due dates.	Outputted to: A42, A43
**	*Resource Schedule*	A plan of control resource availability and allocation. It specifies each resource and its assigned resource group, as well as the operation or transferred from one place to another in a specific time period assigned to it. Only resources that are used/shared by multiple workcells or workstations are on the resource schedule.	Outputted to: A42, A43

Table B.2: A411 Module Data

Mechanism			
Production Activity Planning Component: A software component in a MES that aids users to plan production activities.			
Input			
***	*Process Status*	**see** Table 2.1	Inputted from: A6 (feedback)
* / **	*Manufacturing Orders*	**see** Table 2.1	Inputted from: A2
***	*Updated Document*	Document that is modified to include new information.	Inputted from: A44 (feedback)
**	*Resource Availability*	A report on whether needed resources are available for production during specified time periods.	Inputted from: A44 (feedback)
Output			
**	*Production Orders*	Instructions that are sent to a local area of a factory to start processing a production unit with the starting date and time and the ending date and time.	Outputted to: A42, A43-Control

Table B.3: A412 Module Data

Mechanism
Manufacturing Execution System (MES): A production activity management system that initiates, guides, responds to, and reports on production activities on-line and in real time to production management people. The system aids the Execute Manufacturing Orders activity.

Input			
***	*Process Status*	see Table 2.1	Inputted from: A6 (feedback)
Output			
**	*Production Unit Routing*	A plan that specifies the traveling route of a production unit in a manufacturing facility. The plan also specifies stops for processing and queuing.	Outputted to: A413, A43-Control
**	*Production Unit Definition*	Definition of a lot or a batch. It includes an ID, number of workpieces, and the descriptions of the workpieces. Each workpiece may have a serial number. In the product record, workpiece ID and production unit ID are associated.	Outputted to: A413, A414

Table B.4: A413 Module Data

Mechanism			
Production Unit Scheduling Component: A software component in a MES that aids users to schedule production unit to be processed locally in a manufacturing facility.			
Input			
***	*Process Status*	see Table 2.1	Inputted from: A6 (feedback)
**	*Production Unit Routing*	A plan that specifies the traveling route of a production unit in a manufacturing facility. The plan also specifies stops for processing and queuing.	Inputted from: A412
**	*Production Unit Definition*	Definition of a lot or a batch. It includes an ID, number of workpieces, and the descriptions of the workpieces. Each workpiece may have a serial number. In the product record, workpiece ID and production unit ID are associated.	Inputted from: A412
Output			
**	*Detailed Schedule*	A plan that specifies starting time and finished time of each production unit in the queue locally to an area in the manufacturing facility, such as a workcell, a workstation, or a machine.	Outputted to: A42-Control, A43-Control, A414-Control

Table B.5: A414 Module Data

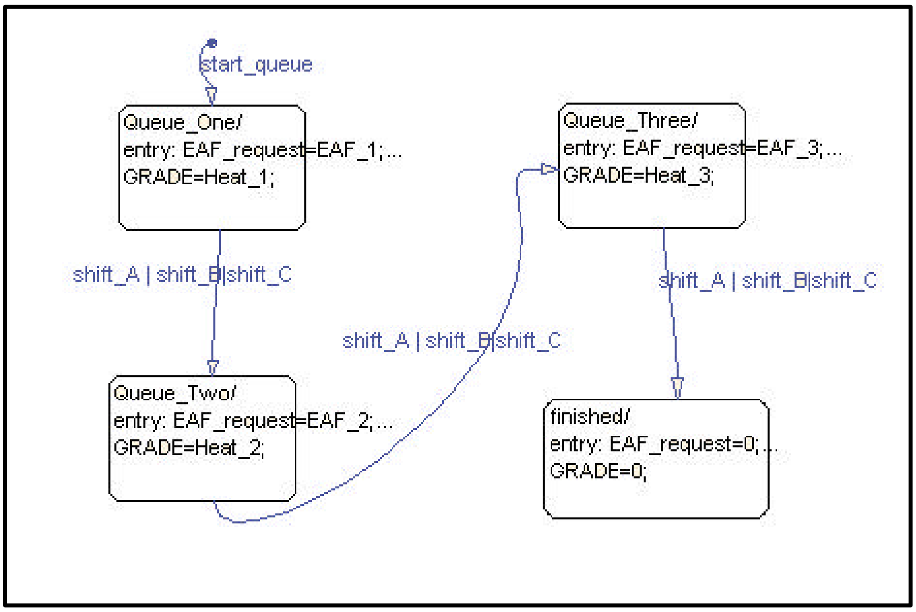
Mechanism			
Resource Scheduling Component: A software component in a MES that aids users to schedule the release of resources to workcells, workstations, and/or machines.			
Input			
***	*Process Status*	**see** Table 2.1	Inputted from: A6 (feedback)
**	*Production Unit Definition*	Definition of a lot or a batch. It includes an ID, number of workpieces, and the descriptions of the workpieces. Each workpiece may have a serial number. In the product record, workpiece ID and production unit ID are associated.	Inputted from: A412
Output			
**	*Resource Schedule*	A plan of control resource availability and allocation. It specifies each resource and its assigned resource group, as well as the operation or transferred from one place to another in a specific time period assigned to it. Only resources that are used/shared by multiple workcells or workstations are on the resource schedule.	Outputted to: A43

Table B.6: A415 Module Data

Mechanism			
Manufacturing Execution System (MES): A production activity management system that initiates, guides, responds to, and reports on production activities on-line and in real time to production management people. The system aids the Execute Manufacturing Orders activity.			
Output			
**	*Personnel Assignment*	A list of workers who are assigned to perform specific operations in the production plan. Each worker is assigned to perform or monitor one or more operations, usually, with due dates.	Outputted to: A43

APPENDIX C

STATEFLOW MODEL AND DETAILED DESCRIPTION OF INPUT QUEUE



The Input Queue STATEFLOW model has four distinct states. All the states 'Queue_One', 'Queue_Two', 'Queue_Three' and 'Finished' are exclusive. The states are represented using a solid board box and only one of the states can be active at a time.

The functionality of the input Queue is as follows:

- An input* event** 'start_queue' triggers the transition of the model to queuing.
- During the 'Queue_One' state, the output*** 'EAF_request' is set to the appropriate EAF-LF machine for Heat 1, the output 'GRADE ' is set to Heat 1. Both the 'GRADE' and 'EAF_request' are sent to all the EAF-LF machines, and is accepted only by the machine that was requested by 'Queue_One'.
- When the correct EAF-LF has accepted Heat 1 and starts to process the heat, an acknowledge back event, either 'shift A' for EAF-LFA, 'shift B' for EAF-LFB or 'shift C' for EAF-LFC. Once the Input Queue block receives any one of these three events, the queue will shift to the next state 'Queue_Two'.
- During the 'Queue_Two' state, similar to 'Queue_One' state, the queue sets the two outputs 'EAF_request' and 'GRADE' with the information of Heat 2.
- When one of the EAF-LFs (A, B or C) has accepted Heat 2 and starts to process the heat, they will broadcast back an event again, either 'shift_A' for EAF-LFA, 'shift B' for EAF-LFB or 'shift C' for EAF-LFC . Once the Input Queue block receives any one of these three events, the queue will shift to the next state 'Queue_Three'.
- During the 'Queue_Three' state, the queue sets the two outputs 'EAF_request' and 'GRADE' with the information of Heat 3.
- When one of the EAF-LFs (A, B or C) has accepted Heat 3 and starts to process the heat, they will broadcast back an event again, either 'shift_A' for EAF-LFA,

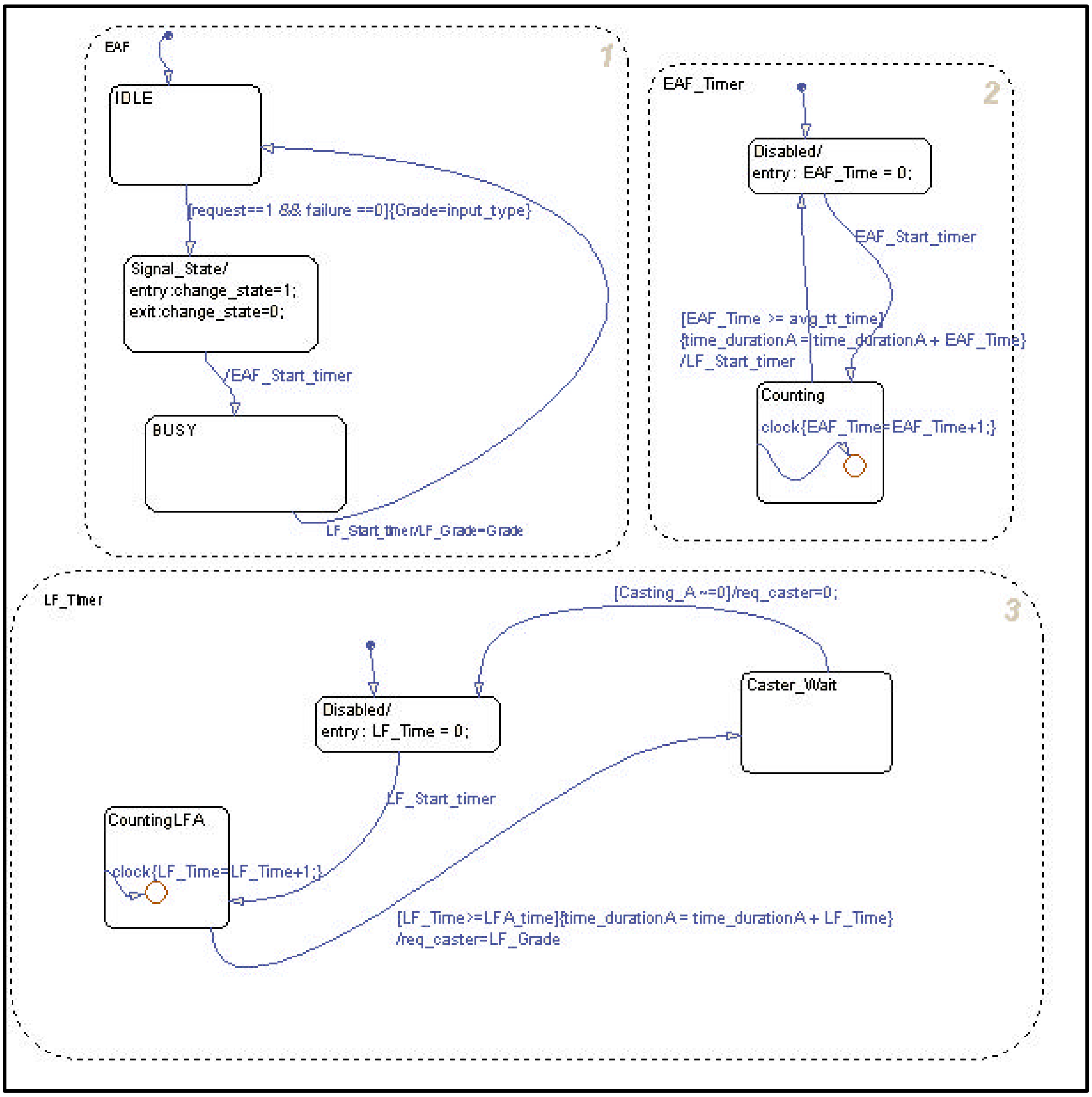
'shift B' for EAF-LFB or 'shift C' for EAF-LFC . Once the Input Queue block receives any one of these three events, the queue will shift to the next state 'finished'.

- The 'finished' state means that all the heats have been processed or are being processed and there is no job left. Therefore, both outputs 'EAF_request' and 'GRADE' are set to zero.

Note* : Scope 'Input' means the value inputted from another SIMULINK or STATEFLOW block.
Note** : Type 'Event' drives the STATEFLOW diagram execution. The broadcast of an event can trigger a transition of states to occur or can trigger an action to be executed.
Note***: Scope 'Output' means the output value to other SIMULINK or STATEFLOW blocks.

APPENDIX D

STATEFLOW MODEL AND DETAILED DESCRIPTION OF EAF-LF



The 'EAF' subsystem describes the status of the EAF. This subsystem has three states.

- The 'IDLE' state indicates that the EAF is ready to process the next heat. When the 'request' input from the Input Queue equals to one, which means the next heat is going to use EAF-LFA, the 'EAF' subsystem goes to the next state 'Signal_State'. At the same time it copies the local variables 'GRADE' with the input value 'input_type' from the Input Queue, which defines the grade of the heat.

- In the 'Signal_State' subsystem a rising-edge event 'shift_A' is created for the Input Queue by setting output 'change_state' to 1. This will trigger the queue transitioning to its next state. Immediately following this the 'change_state' is reset back to zero, and the subsystem 'Signal_State' switches to the "BUSY' state. During the transition, a broadcast local event 'EAF_Start_timer' is executed in order to start the timer subsystem 'EAF_Timer'.
- The 'BUSY' state represents the processing of the incoming heat by the chosen EAF. This state is active up until the 'EAF_Timer' broadcasts the local event 'LF_Start_Timer' to indicate that the EAF processing is done. Then the 'EAF' subsystem sends the heat grade information to the 'LF_Timer' subsystem by copying the grade value in the local variable 'Grade' to the variable 'LF_Grade', at which point the 'EAF' subsystem goes back to the 'IDLE' state for the next possible heat.

Note:

i. The event 'LF_Start_Timer' also triggers the start of the subsystem 'LF_Timer' as described below.

ii. Here the assumption is made that the transfer from EAF to LF takes very little time.

The 'EAF_Timer' subsystem keeps track of the time the EAF needs to process the heat. This subsystem has two states:

- During the 'Disabled' state, the timer 'EAF_Time' is set to zero. When the 'EAF' subsystem accepts the incoming heat, it broadcasts the event 'EAF_Start_timer' (as described above) and starts the 'EAF_Timer' subsystem by going to the 'Counting' state.
- In the 'Counting' state, the 'EAF_Timer' subsystem increments the timer 'EAF_Time' from 0 to the value 'avg_tt_time' which is inputted by an external STATUS block. For EAF-LFA it will be inputted by ELA_STATUS. Similarly for EAF-LF B and C, the input will be given from ELB_STATUS and ELC_STATUS respectively. This time is specific to the heat being processed and is known ahead of time based on the schedule (see Section 4.6 STATEFLOW AND MATLAB GA INTERFACING for how this is done).

The LF is defined by the subsystem 'LF_Timer'. This subsystem serves as a timer for processing the heat in the LF. This subsystem has three states.

- In the 'Disabled' state, the timer 'LF_Time' is set to zero. Event 'LF_Start_timer' starts this timer and the subsystem transitions to the next state. In the 'CountingLFA' state, the 'LF_Timer' subsystem increments the timer 'LF_Time' from 0 to the value 'LFA_Time' which is inputted by an external status block. For EAF-LFA it will be inputted by ELA_STATUS. Similarly for EAF-LF B and C, the input will be given from ELB_STATUS and ELC_STATUS respectively. This time is specific to the heat being processed and is known ahead of time based on the schedule (see Section 4.6 STATEFLOW AND MATLAB GA INTERFACING for how this is done).
- Similar to the 'EAF_Timer', during the 'CountingLFA' state, the timer 'LF_Time' is incremented to a specific value before it goes to the next state

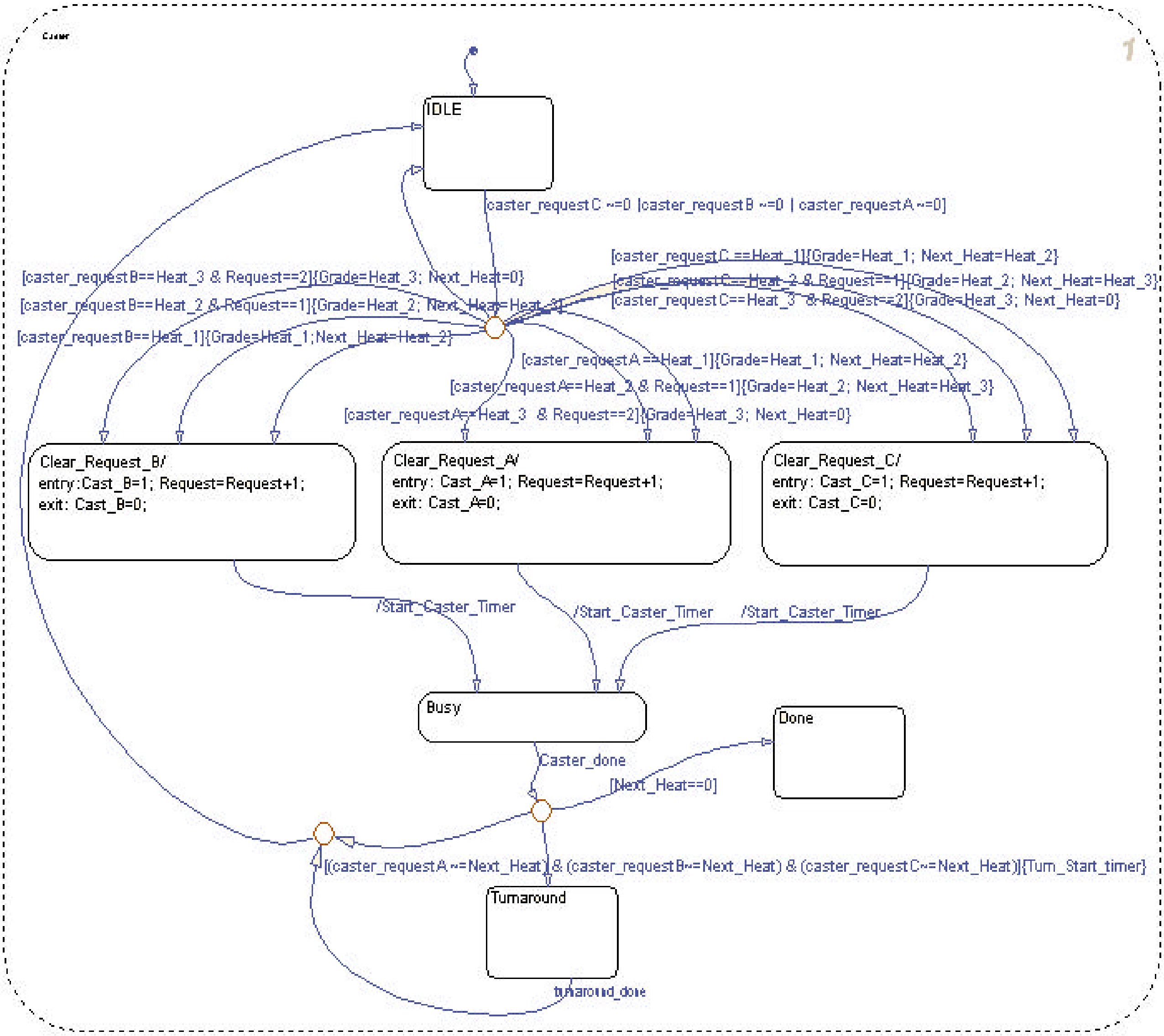
'Caster_Wait'. During this transition the heat grade value is copied from variable 'LF_Grade' to the output 'req_caster' signal. The 'req_caster' informs the Caster the heat in the specific LF is ready for casting.

- During the 'Caster_Wait' state, the LF waits for the Caster to begin processing the heat in the LF. When the Caster accepts the heat in the LF, it sets the variable 'Casting_A' to a non_zero value and the 'LF_Timer' for EAF-LFA goes back to the 'Disabled' state. Now it is ready for the next heat from the EAF.

APPENDIX E

STATEFLOW MODEL AND DETAILED DESCRIPTION OF CASTER

The model for the Caster has three subsystems. They are ‘Caster’ (as shown below), ‘Turnaround_Timer’ and ‘Caster_Timer’. These subsystems are shown in separate diagrams due to their complexity.



The ‘Caster’ in the steel making process shown above has seven states.

- ‘IDLE’ state means the Caster is idle. When any one of LFA, LFB or LFC are ready for casting, they broadcast the event, ‘caster_requestA’, ‘caster_requestB’or ‘caster_requestC’ respectively to the Caster. Because the ‘Caster’ subsystem has to enforce the sequence of heat processing, it does not always accept the heat that is ready first. Instead it always checks if the ready heat is the expected heat. The expected heat value is defined by local variable ‘Request’. When ‘Caster’ expects the first heat the

'Request' value is 0; when 'Caster' expects the second heat the 'Request' value is 1; when 'caster' expects the third heat the 'request' value is 2. The 'Caster' subsystem also checks which heat from LF (A, B or C) is ready for casting.

- If 'Caster' decides to cast the heat from LF-A, it goes to 'Clear_RequestA' state.
- If 'Caster' decides to cast the heat from LF-B, it goes to 'Clear_RequestB' state.
- If 'Caster' decides to cast the heat from LF-C, it goes to 'Clear_RequestC' state.
- If the heat in the LF is ready but that heat is not the one expected by the 'Caster', it goes back to 'IDLE' to wait for the right heat.

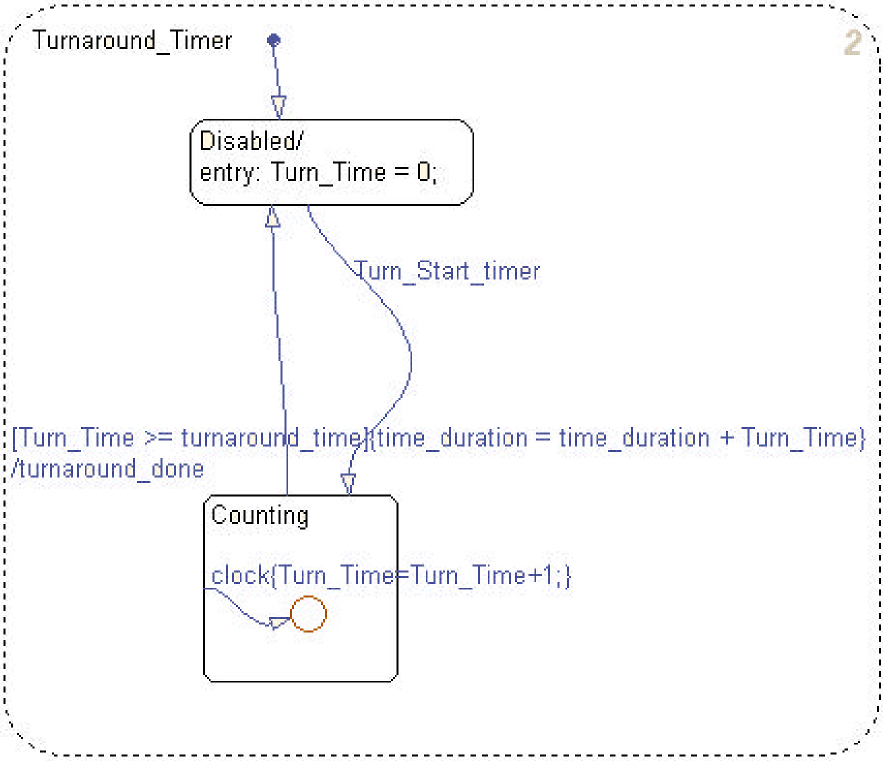
During either 'Clear_RequestA', 'Clear_RequestB' or 'Clear_RequestC' state, the 'Caster' subsystem informs the corresponding EAF-LF block that it accepts their ready heat by setting output variable 'Caster_A', 'Caster_B' or 'Caster_C' to 1. This event also causes the variable 'Request' to be incremented by 1, so the 'Caster' will expect the next heat after the received heat is cast. Then the 'Caster' goes immediately to the 'Busy' state.

- o During the 'Busy' state, the 'Caster' is casting the heat and waits until the subsystem 'Caster_Timer' counts to the specific value as an indication of the casting of this heat is finished. This value is a local variable in the 'CASTER' block and is fixed based on the last column of values defined in Table 3.1 (capacity) multiplied by the bottom row of values defined in Table 3.4 (tons/min), to give the total minutes needed for casting the capacity of tons needed. Then the 'Caster' decides which next state to go to.
    - If the last heat is finished, the whole simulation is done and the 'Caster' goes to 'Done' state.
    - If the finished heat is not the last one and the next expected heat is ready, 'Caster' then goes back to the 'IDLE' state to process the next heat.
    - If the expected heat is not ready, the 'Caster' goes to the 'Turnaround' state to wait for the turnaround time.
- o During the 'Turnaround' state, the subsystem 'Turnaround_Timer' counts to a specific value, the 'Caster' then goes back to 'IDLE' state

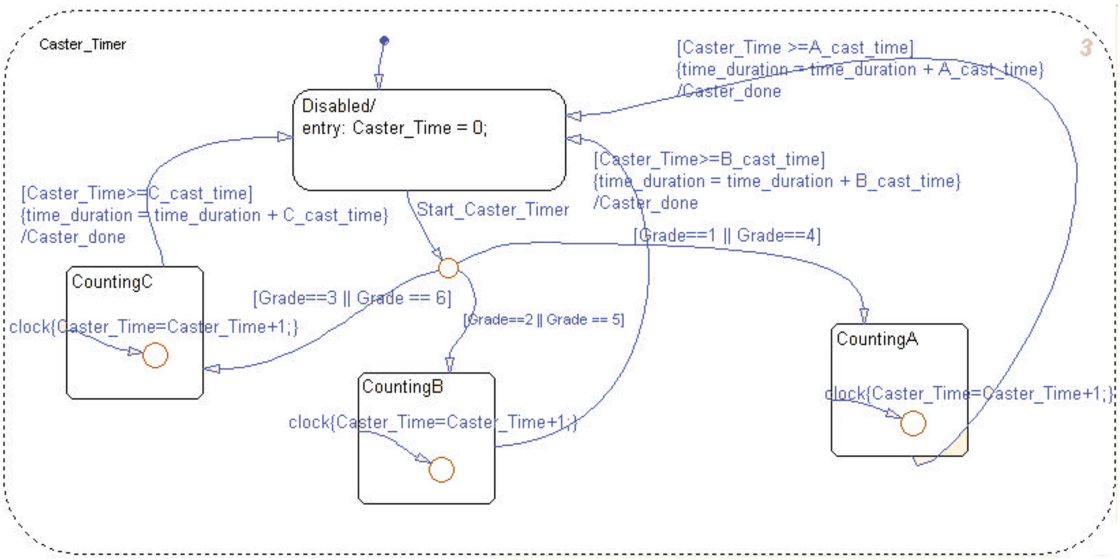
<u>TURNAROUND TIMER SUBSYSTEM OF CASTER MODEL</u>

The subsystem 'Turnaround_Timer' is a timer to keep track of the caster turnaround time. It has two states: 'Disabled' and 'Counting'. The implementation of 'Turnaround_Timer' subsystem is very similar to the 'EAF_Timer' in the EAF-LF block as discussed above.

The detail of the model is shown in the figure below.



CASTER TIMER SUBSYSTEM OF CASTER MODEL



The above figure shows the detail design of the subsystem ‘Caster_Timer’. It is a timer to keep track of the caster time for one heat. It has four states: ‘Disabled’, ‘CountingA’, ‘CountingB’ and ‘CountingC’. The implementation of ‘Caster_Timer’ subsystem is very similar to the ‘LF_Timer’ in the EAF-LF model as discussed above. Since each grade of heat has unique casting times, it is needed to have distinct states to count these different casting times.

APPENDIX F

DETAILED CASE STUDY DATA FOR A4 MODULES

Table F.1: A41 Module Case Study Data

Mechanism			
Manufacturing Execution System (MES): A production activity management system that initiates, guides, responds to, and reports on production activities on-line and in real time to production management people. The system aids the Execute Manufacturing Orders activity.			
	Input Data	Specific Example	Plant-Specific Data
***	*Process Status*	**see** Table 3.5	*Heat Status at Failure*
* / **	*Manufacturing Orders*	**see** Table 3.5	*Heat Orders*
***	*Updated Document*	The information that affects the detailed schedule. Contains production status information, feedback from A44. Some examples are: machine breakdowns and power consumption available. Note each module can stream different updated data, depending on what data is needed.	*EAF_failed, instant_ftime*
**	*Resource Availability*	When an EAF or Ladle goes out of service, need to know for how long, to adjust the schedule.	*EAF_status*
Output			
**	*Production Unit Routing*	The heating time array for the day for the given machines (EAFs and LFs). Based on the scheduled heats, define the time needed for heating on each machine.	The *sf_schedule*(*ga2sf.m*), *ELA_status*(*ELAstatus.m*), *ELB_status* and *ELC_status* give the appropriate definitions to the STATEFLOW model
* / **	*Operation Sequence*	The output of the GA giving the order of heats and the machines to process them on. (Note this is between a 6 hour half day to 12 hour full day schedule)	*H1|M1|H2|M2|H3|M3* *H –Heat,* *M –Machine(EAF,LF)*
**	*Detailed Schedule*	Based on the short term day operation but also need to incorporate the long term. Using Genetic Algorithm (GA) output, extrapolate when heats will be started and finished.	*Times_for_Schedule*
**	*Production Orders*	This is taken from the *Detailed Schedule*, and simply represents the processing order of the different heats.	*H1| H2| H3* *H –Heat*

*	*Personnel Assignment*	Personnel are not modeled in this example. However, there are cases where certain personnel are needed on certain resources depending on the schedule. This could be as simple as a database for resources that allows the operator to sign in and use the resource.	Outputted to: A42, A43
**	*Resource Schedule*	Availability of resources is modeled by the Idle state. The different heats are assigned to the EAFs and Ladles. Given GA output extrapolate the order of EAF/Ladles.	*M1 |M2 |M3* *M –Machine (EAF,LF)*

Table F.2: A411 Module Case Study Data

Mechanism		
Genetic Algorithm Component: An algorithm component for MES that aids users to optimize production activities (see Chapter 4).		
Input Data	Specific Example	Plant-Specific Data
Process Status	**see** Table 3.5	*Heat Status at Failure*
Manufacturing Orders	**see** Table 3.5	*Heat Orders*
Specifications	**see** Table 3.5	*Heat_Time_Table*
Resource Availability	A workstation is scheduled for use in production. Check the maintenance schedule to confirm that downtime is not planned within the current planning horizon.	Inputted to GA *EAF Planned Status* for day. This checks the maintenance schedule to determine which machines are available to be scheduled for the day. Note the EAF and LF are coupled, if EAFA is unavailable so is LFA (**ELA_DOWN = 1**).
Updated Document	The information that affects the detailed schedule. Contains production status information, fed back from A44. Specific machine status capability. Note each module can stream different updated data, depending on what data is needed.	This information is contained in *ELA_heat _not_allowed* for the heats that are not allowed to be processed on EAFA and LFA based on the capacity demand.
Output Data	Specific Example	Plant-Specific Data
Operation Sequence	The output of the GA giving the order of heats and the machines to process them on. (Note this is between a 6 hour half day to 12 hour full day schedule)	*H1|M1|H2|M2|H3|M3* *H –Heat, M –Machine (EAF,LF)*

Table F.3: A413 Module Case Study Data

Mechanism
Time Translator Scheduling Algorithm: A software component in a MES that aids users to schedule start and stop times for production units to be processed locally in a manufacturing facility.

Input Data	Specific Example	Plant-Specific Data
Specifications	**see** Table 3.5	*Heat_Time_Table*
Production Unit Definition	Definition of a heat. It includes an ID, number of tons, the position of it in the *Operation Sequence* and descriptions of the customer requirements (*Heat Orders*).	*Heat Definition* Contains: *Operation Sequence* and *Heat Orders*
Output Data	**Specific Example**	**Plant-Specific Data**
Detailed Schedule	Given schedule extrapolate when heats will be started and finished using *Heat_Time_Table*.	*Times_for_Schedule* (See Time Translator Scheduling Algorithm in Section 4.4.2)

Table F.4: A414 Module Case Study Data

Mechanism		
Resource Rescheduling Algorithm: A software component in a MES that aids users to schedule and reschedule (if a failure occurs) the release of resources to workcells, workstations, and/or machines.		
Input Data	**Specific Example**	**Plant-Specific Data**
Process Status	**see** Table 3.5	*Heat Status at Failure*
Detailed Schedule	Given schedule extrapolate when heats will be started and finished using *Heat_Time_Table*.	*Times_for_Schedule* (See Time Translator Scheduling Algorithm in Section 4.4.2)
Output Data	**Specific Example**	**Algorithm**
Resource Schedule	Knowing the *Detailed Schedule* and *Process Status* the Production Unit Rescheduling Algorithm is run to provide a rework schedule for the instantaneous heat activity. This information can be used to determine if Turnaround can be avoided and find a solution for least cost based on the availability of resources.	*Re_Schedule* Given *Heat Status at Failure* assign new order for EAF/Ladles (see Production Unit Rescheduling Algorithm in Chapter 4).

APPENDIX G

PSEUDO CODE FOR PRODUCTION UNIT RESCHEDULING ALGORITHM

function DECISION TREE (heat_in_machine, time_left_in_machine, heat_in_CC, time_left_in_CC)
returns *order* for active heats

inputs: *heat_in_machine*, the active heats in the EAF or LF, vector size three
time_left_in_machine, the active heats time (min) left in EAF or LF, vector size three
heat_in_CC, the active heat in the CC, vector size one
time_left_in_CC, the active heat time (min) left in CC

if no turnaround for at least one active heat in machine

-for all heats with no turnaround, find least cost of transition with current heat in CC as first in sequence, followed by heat with no turnaround
- choose heat with no Turnaround and with minimum cost
- choose rest of sequence for all active heats in LF for minimum cost

else

-find best order for least cost of transitions for active heats in LF w/o starting with heat in CC, since no transitioning will occur due to turnaround

return best order for active heats in machine

Run DECISION TREE for active heats in LFs *; return **order**

Was **order** empty?
Empty means there were no active LF heats

* here the inputs to DECISION TREE function are the instantaneous status of the specific plant machines at times of failure.

YES
Run DECISION TREE for active heats in EAFs with: same heat_in_CC, and time_left_in_CC as was used when running LFs DECISION TREE

NO
Run DECISION TREE for active heats in EAFs, while taking into account the time shift produced and the input times it effects. Now the heat_in_cc will be the last heat from **order**. The input times affected are due to Turnaround and casting the active LFs. So time_left_in_cc will reflect the time that passes for casting the active heats in the LFs, plus turnaround if it occurred.

Input: time_left_in_machine for EAF will be its time plus time needed on its LF. Note if its LF was active with a heat, the EAF must incorporate time of waiting for the LF to release its active heat for casting on top of the time it will need for the LF heating.

Are heats left affected by failed machine?

yes-do GA (run "sub_genetic_algo.m" **) using machines that are not down, for heats left that were not active at time of failure (first heat and machine for it will be taken from **order** above)

** see Hadjimichael [7] for details

no- since GA already done for optimal sequence, do not redo it if heats left are not affected by machine that failed. This entails that the last active heat maintains the GA sequence with the first of the non-active heat(s).

APPENDIX H

RAW DATA FOR FIRST_POP AND LAST_POP GENERATED BY THE GA

FIRST POP

HEAT	MACHINE	HEAT	MACHINE	HEAT	MACHINE	HEAT	MACHINE	HEAT	MACHINE	HEAT	MACHINE	PENALTY	MAKESPAN	COST	FITNESS
4	2	3	3	6	1	5	2	2	1	1	3	0	485	4500	0.01105
5	1	1	2	3	1	4	2	2	1	6	3	0	485	7500	0.010695
3	2	4	2	5	3	6	3	1	2	2	1	0	485	10000	0.010417
5	1	1	2	2	1	3	2	6	1	4	2	0	485	5000	0.010989
3	1	5	2	4	2	1	2	6	1	2	1	0	475	6500	0.011299
4	2	1	2	2	2	3	2	5	1	6	2	400	505	7000	0.009524
6	3	3	3	1	2	2	2	5	1	4	3	0	485	5000	0.010989
4	3	5	3	2	2	3	3	6	1	1	3	400	505	5000	0.009709
2	2	4	2	3	1	6	2	5	1	1	3	0	495	4500	0.010582
3	2	4	3	5	3	6	1	1	3	2	1	0	485	10000	0.010417
6	1	5	1	1	2	4	2	2	1	3	2	0	475	7000	0.011236
6	1	5	1	2	1	3	1	4	2	1	2	0	475	2000	0.011905
5	1	2	3	1	2	6	3	4	2	3	3	0	485	5000	0.010989
3	2	4	2	1	2	2	1	5	2	6	1	0	485	5000	0.010989
2	1	5	1	1	3	3	1	6	1	4	2	0	485	2500	0.011299
2	3	1	3	3	1	4	2	6	1	5	1	0	495	7000	0.010309
5	3	1	2	3	1	2	1	4	2	6	1	0	495	7000	0.010309
5	3	4	2	6	1	1	2	2	1	3	1	0	495	7500	0.010256
3	1	1	2	6	2	2	1	5	2	4	2	0	475	4500	0.011561
6	3	2	3	3	2	4	2	1	2	5	2	400	505	7000	0.009524
4	2	2	3	3	2	1	2	5	1	6	3	0	485	10000	0.010417
4	3	3	2	1	2	5	1	6	1	2	1	0	485	9500	0.010471
1	2	5	1	3	1	6	1	2	2	4	2	0	485	7000	0.010753
4	2	6	2	3	3	5	2	2	1	1	3	0	485	4500	0.01105
AVG												50	487.5	6270.833	0.010697

LAST_POP

HEAT	MACHINE	HEAT	MACHINE	HEAT	MACHINE	HEAT	MACHINE	HEAT	MACHINE	HEAT	MACHINE	PENALTY	MAKESPAN	COST	FITNESS
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
6	1	5	1	2	1	3	2	4	2	1	2	0	475	2000	0.011905
AVG												0	475	2000	0.011905