Planning and Scheduling Optimization in Integrated Steel Production

PLANNING AND SCHEDULING OPTIMIZATION

in

Integrated Steel Production

by

Patrick Alexander Philippe Carter, B.Eng.Mngt

A Thesis
Submitted to the School of Graduate Studies
in Partial Fulfillment of the Requirements
for the Degree
Master of Applied Science

McMaster University

MASTER OF APPLIED SCIENCE (2015)	McMaster University
(Chemical Engineering)	Hamilton, Ontario, Canada

TITLE:	Planning and Scheduling Optimization in Integrated Steel Production
AUTHOR:	Patrick A.P. Carter, B.Eng.Mngt (McMaster University, Canada)
SUPERVISOR:	Dr. Christopher L.E. Swartz
NUMBER OF PAGES:	x, 110

ABSTRACT

Production planning is a critical component in supply chain management. The goal of production planning is to meet market demand while minimizing operational costs. There is inherent uncertainty in manufacturing systems due to unscheduled shutdowns and variable production rates. Additionally, actual demand levels cannot be predicted accurately. As a result, there is value in creating a production plan that considers these uncertainties.

Scheduling is also a critical component in supply chain management, but at a smaller level of time granularity. Industrially sized scheduling problems are often on such a large scale that the problem is computationally difficult to solve. Consequently, there is value in creating a mathematical model and selecting a solution algorithm that minimizes this burden.

This work aims to determine the benefit of a stochastic production planning model over its deterministic counterpart. The problem utilizes a multi-period, multi-product aggregated planning model with a finite horizon in a steel manufacturing environment. The production and demand uncertainty is modelled as a two-stage stochastic mixed integer linear program. The problem utilizes a Monte Carlo simulation technique to create the scenarios used in the optimization. The objective of the optimization is to determine the production volume and inventory levels for each discrete time interval while minimizing the weighted cost of production and surplus. The production decisions must be non-anticipative, immediately implementable, and are subjected to production capacity and inventory holding constraints. This work also investigates the advantages a cost-based model has over its goal-programming counterpart. Finally, this thesis develops several mathematical batch scheduling models that use different modelling paradigms in an effort to compare their computational complexity. With the selection of an appropriate model, model extensions are added to replicate an industrially relevant steel mill scheduling problem for a finishing line using data from a facility located in Ontario, Canada.

ACKNOWLEDGEMENTS

I wish to express my gratitude to Dr. Christopher Swartz for his continued support these past two years. I would not be where I am today without his inspiration and guidance. This thesis would not have been possible without the involvement of our industry sponsor, ArcelorMittal Dofasco. In particular, Adam Smyth, Colin Chisholm, and Steven Somerville have graciously provided their time and expertise in this endeavor. I would also like to thank my colleagues in the McMaster Advanced Control Consortium for the great experiences they have provided me both academically, athletically and socially. My thesis would not be possible without the financial assistance from the McMaster Department of Chemical Engineering and the Ontario Graduate Fellowship program.

This thesis is dedicated to my parents, Dawn & Richard Carter, for instilling in me the value of education, and for providing me unconditional support to pursue any goal that I may have.

Patrick Carter

April 2015

Table of Contents

1 Introduction **1**

1.1 Motivation and Research Objectives 1

1.2 Main Contributions 3

1.3 Thesis Overview 4

2 Literature Review **6**

2.1 Integrated Steel Production 6

2.1.1 Primary Operations 7

2.1.2 Finishing Operations 8

2.1.3 Solution Methodologies 11

2.2 Aggregate Production Planning 17

2.3 Short Term Batch Scheduling 20

2.3.1 Campaign Scheduling 25

2.3.2 Real-world scheduling examples 26

2.4 Optimization Under Uncertainty 27

3 Aggregate Production Planning Optimization **30**

3.1 Application Context: Medium Term Planning Process 31
3.2 Deterministic Aggregate Production Planning Model 34
3.2.1 Constraints 36
3.2.2 Objective Function 39
3.3 Stochastic Aggregate Production Planning Model 42
3.3.1 Objective Function 43
3.3.2 Constraints 43
3.3.3 Scenario Generation 45
3.4 Cost-Based Stochastic Aggregate Production Planning Model 47
3.4.1 Objective Function 51
3.4.2 Constraints 52
3.5 Production Planning Model Comparison Case Studies 56
3.5.1 Case Study 1 - Benefits of Stochastic Modelling 57
3.5.2 Case Study 2 - Cost Reductions and Operability 59

4 Scheduling Using Mathematical and Constraint Programming 65

4.1 Application Context: Campaign Scheduling 66
4.2 Comparison and Selection of Modelling Type 70
4.2.1 Continuous Time Models 70
4.2.2 Constraint Programming Model 83
4.2.3 Case Study 3 - Model Solution Time 86
4.3 Complete ISP Mathematical and Constraint Programming Models . . 89
4.3.1 ISP Immediate Precedence Model 91

4.3.2 ISP Constraint Programming Model 92
4.3.3 Case Study 4 - Industrial Solutions 98

5 Conclusions and Recommendations 101
5.1 Conclusions 101
5.2 Recommendations for Further Work 103

References 105

List of Figures

1.1 Supply chain planning matrix (Meyr et al., 2002) 2

2.1 Steel mill allocation problem (Okano et al., 2004) 9

2.2 Steel mill sequencing problem (Okano et al., 2004) 10

2.3 Branch and bound algorithm 14

2.4 Constraint programming propagation algorithm 16

2.5 Characterization of scheduling problem features (Mendez et al., 2006) 21

2.6 Classification of scheduling problem models (Mendez et al., 2006) . . 22

3.1 Simplified process flow diagram of a typical integrated steel mill . . . 32

3.2 Planning hierarchy used at a typical steel mill 32

3.3 Information flow required to make production decisions 35

3.4 Demand uncertainty distribution for product G 46

3.5 Histogram of sampled demand profile for n=10 scenarios 46

3.6 Histogram of sampled demand profile for n=100 scenarios 47

3.7 Noise distributions in production for a sample data set 51

3.8 Stochastic solution time 59

3.9 Cost break down and comparison for the inventory target SAPP (a) and CBSAPP (b) 60

3.10 Baseline(a) and oscillatory behaviour introduced at the Hot Mill due to the CBSAPP(b) 61

3.11 Oscillation reduction at the Hot Mill shown in (b) when compared to the CBSAPP(a) 62

3.12 Comparison of the selected inventory targets under the SAPP(a), CB-SAPP(b) and COIT(c) 63

4.1 Classification of customer orders into campaigns and campaign types 67

4.2 Solution time as a function of problem size for different model types . 88

4.3 Cycle of campaigns and specific steel mill constraints 90

4.4 Selection of sequenced campaigns from the list of alternates 94

4.5 Reduction in objective function value as a function of solution time for the number of late campaigns 99

4.6 Gantt chart of OPL solution for 33 campaigns 100

List of Tables

2.1 Characteristics of continuous time optimization models (Mendez et al., 2006) . . . 25

3.1 Estimated category contribution towards holding cost (Richardson, 1995) 49

3.2 Linear cost coefficient for different plant capacities of the form $y = mx + b$ . . . 50

3.3 Summary of stochastic production planning models . . . 63

3.4 Comparison of stochastic production planning models problem size . 64

3.5 Comparison of stochastic production planning models solution time . 64

4.1 Problem sizes and specifications for different models . . . 87

4.2 Time taken to reach the best feasible solution for different models . . 88

Chapter 1

Introduction

1.1 Motivation and Research Objectives

A supply chain is an interconnected network of business functions involved in procuring, producing, distributing and selling a product or service. A supply chain also consists of multiple stakeholders including suppliers, manufacturers, warehouses, retailers and customers. Each individual business function requires extensive planning in order to meet the stakeholders demands while remaining competitive in today's environment. Figure 1.1 shows the network between the business functions and stakeholders, and the level of planning required to successfully navigate this problem. This becomes particularly complex due to horizontal and vertically integrated processes that require cross-functional co-ordination. Decisions made by one stakeholder often affect the upstream and downstream functions. Additionally, decisions made at longer term time horizons often affect shorter time horizon decisions. Steel manufacturers are a classic example of companies with a complex supply chain. In particular, In-

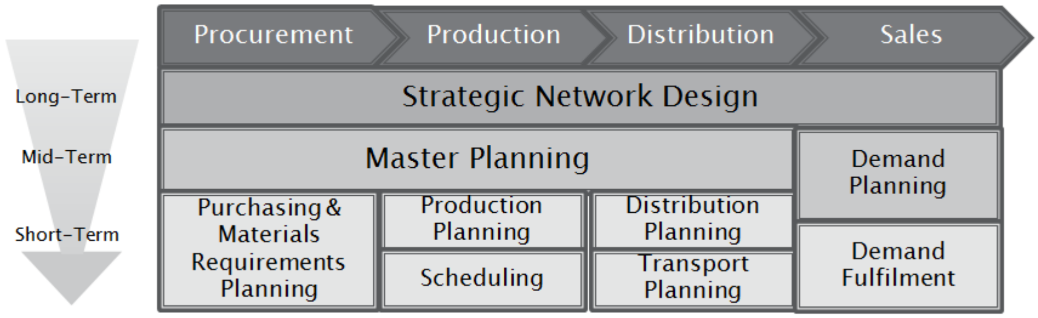

Figure 1.1: Supply chain planning matrix (Meyr et al., 2002)

tegrated Steel Production (ISP) includes all aspects of steel making required to turn raw ingredients into value-added steel. An ISP facility directly connects the steel making furnace, continuous caster, hot rolling mill, cold rolling mill and finishing mill into a synchronized production flow. These processing units will be discussed in detail in the following chapter. The important concept is that this type of integrated production scheme has many advantages when compared to a traditional segregated process including improved productivity, reduced energy consumption, and enhanced product quality (Tang et al., 2001). As a consequence, this large-scale facility relies heavily on a well developed production plan and schedule in order to realize these benefits of integration.

Mathematical optimization is one method for solving the complex supply chain planning problem found in industrial applications like steel making. An area of focus in supply chain optimization includes the development of models for strategic planning of large-scale multi-period optimization problems and their integration with scheduling problems (Grossmann, 2004). Grossmann (2004) has also stated that an efficient solution of these models and their extensions is still lacking when considering problems of industrial proportions. Finally, Grossmann (2004) has identified that the area of incorporating uncertainty through the use of stochastic optimization models is an area that is ripe for advances.

The problems and opportunities outlined above have motivated a number of research objectives investigated in this thesis. The first objective is to develop a deterministic and stochastic production planning optimization model suitable for use in an ISP facility. This will allow for a quantification of the benefits of including uncertainty in a medium term planning model. The second objective is to investigate several different deterministic ISP scheduling models in an effort to produce a solution to an industrially sized problem in a computationally tractable way.

1.2 Main Contributions

The objectives discussed above have led to a number of contributions to the research literature which are outlined in the following:

1. **Stochastic aggregate production planning formulation.** A two-stage stochastic aggregate production planning model is developed for the inclusion of demand uncertainty. A case study is presented to outline the benefits of including uncertainty in the optimization model in a non-anticipative formulation rather than a reactionary deterministic formulation.

2. **Cost savings by objective function selection in stochastic production planning.** The effect on the profitability of an ISP in regards to the selection of different objective functions is investigated. In particular, the tradeoffs between a goal-programming objective function, a strictly cost-based objective function, and a compromise between the two is explored in a case study.

3. **Analysis of mathematical programming formulations for use in scheduling problems.** Modelling decisions and their impact in regards to computational efficiency is investigated. Several modelling paradigms of varying time representation are developed and compared.

4. **Constraint programming formulation for steel finishing line scheduling.** Constraint programming is an alternative to mathematical programming for solving combinatorial optimization problems. This paradigm is contrasted against the traditional mathematical programming models and introduced to the process systems community. This formulation is extended to be implementable at an ISP facility. Direction for future research areas is provided.

1.3 Thesis Overview

Chapter 2 – Literature Review

An overview of relevant research areas is provided. This includes a survey of current literature and outlines key terms and motivations for the research area. The areas include integrated steel production, aggregate production planning, short term batch scheduling including constraint programming, and optimization under uncertainty.

Chapter 3 – Aggregate Production Planning Optimization

This section first provides context to the industrial application problem. A deterministic aggregate production planning model is developed as the control. Incremental improvements on this model are made in the form of including uncertainty, and moving the entire model to cost basis. The benefits of these incremental improvements are quantified in two case studies using industrially relevant data.

Chapter 4 – Scheduling Using Mathematical and Constraint Programming

A series of mathematical optimization models are introduced in an effort to solve an industrially sized batch scheduling problem. These models are compared with a constraint programming model in terms of computational complexity in a case study. The constraint programming model is extended to solve a scheduling problem using industrial constraints and data.

Chapter 5 – Conclusions and Recommendations

Concluding remarks on the benefits of stochastic production planning and the proper selection of scheduling models are given. Key results of the industrial case studies are presented and the main conclusions are highlighted. Recommendations for future areas of research are proposed.

Chapter 2

Literature Review

The intent of this chapter is to provide a review of relevant concepts in the research literature. The main topics under consideration include current work in integrated steel production, aggregate production planning, short term batch scheduling, and optimization under uncertainty. Each section will provide direction, context and motivation to key reviews and important papers.

2.1 Integrated Steel Production

The reader is directed towards a review of the planning and scheduling systems for integrated steel production by Tang et al. (2001). In this paper, Tang discusses the emergence of a single integrated facility that incorporates primary and finishing operations. Primary operations include processing elements such as steel making, continuous casting, and hot rolling. Finishing operations include processing elements such as cold rolling, pickling, annealing, galvanizing and tinning. The steel mill used in the case studies of this thesis is highly integrated and as such, this research paper is particularly relevant. The reason for the emergence of this integrated facility is due to the benefits of improved productivity, reduced energy consumption, enhanced product quality and shortened wait times between stages (Tang et al., 2001). As a consequence, the procedure of planning and scheduling a facility like this becomes increasingly difficult due its large scale and interconnectedness. This problem becomes additionally complex since the objectives of the different production stages are often conflicting. In order to handle this large scale problem, most researchers break down the problem into its individual processing elements. Following suit, this literature review will start with the processing element titled Primary Operations.

2.1.1 Primary Operations

Steel making involves selecting and processing material in large blast or electric arc furnaces where iron ore is reduced into molten metal and combined with specific ingredients to produce a certain grade of steel. Steel making is an energy-intensive and time-critical process with highly complex and constrained environments (Harjunkoski et al., 2003). Additionally, the decisions made are still often performed manually. As a result, extensive research has been conducted that attempts to create a mathematical model of the process environment from which to optimize. Continuous casting is one particular processing element that has received most of the research attention, specifically in the operations research literature community. The reason for its attraction is that it is typically the bottleneck in steel making (Bellabdaoui and Teghem, 2006). Continuous casting is the process of drawing molten steel from a tundish into a solid steel band. The typical casting problem aims to determine the sequencing, timing, and allocation of steel to a specific processing unit. There are strict constraints on material balances and timing (Atighehchian et al., 2009). The hot mill consists of a series of rollers which exert extreme pressures on the steel as it passes along a conveyor. The sum of these forces of individual rollers reduces the steel slab into a coil with specific dimensions of length, width and thickness. The metallurgical properties are controlled by the temperature and rate of temperature change during the process. The reader is directed towards the work of Bellabdaoui and Teghem (2006), Harjunkoski et al. (2003), Tang et al. (2000) for more information.

2.1.2 Finishing Operations

Finishing operations planning and scheduling in a steel mill is a relatively unexplored area of research compared to primary operations. However, it behaves largely like a flow-shop scheduling problem, which has received much attention. The subunits included in finishing operations include all units after the hot mill such as picking, annealing/tempering, and various coating operations. Pickling removes the surface oxidation created by the extreme heat used in primary operations. One of the exact pickling processing units used in the case studies of this thesis is explored in Sekiguchi et al. (1996). However, this particular paper was focused on advanced control of the unit and not planning or scheduling optimization. Annealing and tempering is the application of heat to change the ductility of the steel. Coating includes galvanization and tinning that are used to prevent future rusting.

The leading introduction to finishing operations in the steel industry is provided by Okano et al. (2004). This author also concluded that there are no papers in the literature that currently address the finishing line scheduling problem. In this paper, the

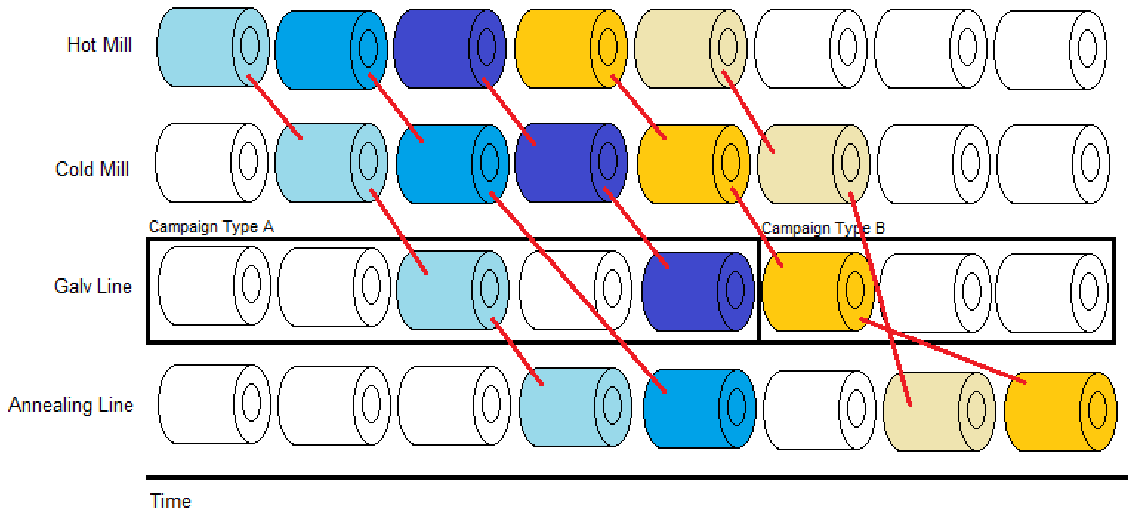

Figure 2.1: Steel mill allocation problem (Okano et al., 2004)

authors replace a scheduling process currently solved manually with an optimization algorithm that creates campaigns (specific production runs) for steel coils on four continuous processing lines for one month of operation. The authors explored two subproblems: campaign allocation and campaign sequencing. The former involves creating campaigns for each unit and partitioning coils into these campaigns. A diagram of this process can be found in Figure 2.1. The latter problem involves the ordering of coils inside a campaign which is a problem with a complex list of constraints. Each coil has specific properties including width, thickness, length, type, due date, priority, and grade. Certain coils may not be allowed to be linked together if their temperature, width, or thickness are sufficiently different. Additionally, the edges of the steel coil cause marring of the rollers which in turn transfers blemishes onto the subsequent steel products. As a result, the sequencing of the slabs effects the product quality. Finally, any significant change in the type of subsequent steel coils may require downtime in order to repair or replace the rollers. The number of coils assigned to each campaign is between 50 and 500, making this a particularly

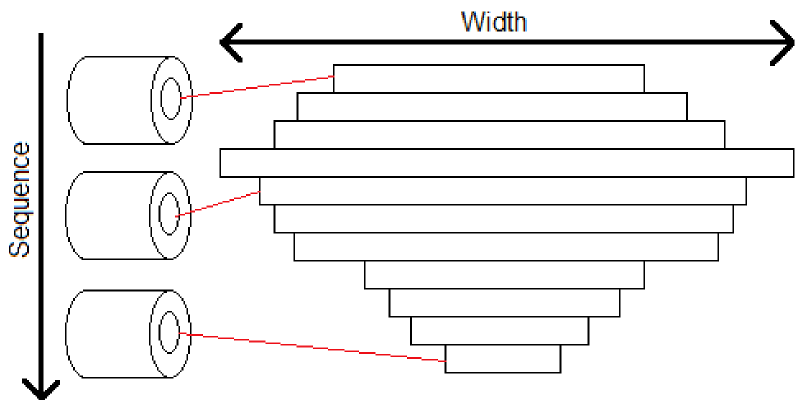

Figure 2.2: Steel mill sequencing problem (Okano et al., 2004)

difficult problem to solve. A diagram of this problem can be found in Figure 2.2. The authors solved this problem using a four tiered methodology. First, to reduce the complexity of the problem, clustering was used, which groups together coils with similar parameters. Second, each cluster is constrained to a time window, which is localized around the release dates and due dates. Third, the allocation problem and scheduling problems are removed from each other. Allocation of coils to campaigns is determined first, followed by a sequencing of the coils. The solution method used is similar to a traveling salesperson search heuristic. The authors were able to solve 20 to 25 thousand coils within a one-hour time limit on solution time.

The steel making, continuous casting, hot/cold rolling, and finishing mill planning and scheduling problems discussed above are all combinatorial in nature. The solution methodologies currently employed to solve these problem are outlined in the following section.

2.1.3 Solution Methodologies

A number of different solution methodologies have been proposed to solve planning and scheduling problems in the research literature. These methodologies can be broken down into several categories including:

1. Mathematical Optimization: This method establishes a mathematical model of the planning or scheduling problem and passes the model to a solution algorithm that solves the problem to a specified solution tolerance.

2. Intelligent Search: This method uses random search or heuristic methods such as genetic algorithms or simulated annealing in order to find a feasible solution to the problem in a comparatively shorter amount of time when considering Mathematical Optimization.

3. Constraint Programming: This method also uses a mathematical model of the problem but the solution algorithm searches only the feasible region. Additionally, the model allows for logical constraints and uses the constraints to direct the solution algorithm.

4. Manual Heuristics: This method uses a scheduler's expertise and experience to follow a series of steps that form a near-optimal solution.

5. Human-Machine Coordination: This method combines the previous ideas and allows a human scheduler to interact with a computerized scheduling system.

There are positives and negatives to all of these methods. However, the solution methodologies used in this thesis focus on mathematical optimization and constraint programming. As such, these topics will be discussed in detail in the following sections.

Mathematical Optimization

Mathematical Optimization is the science of determining the best solution to mathematically defined problems. The problems are defined using models of reality in some form or another that include an objective or goal, and optionally include a series of equalities and inequalities that restrict the feasible region. Formally, this is defined in Equations 2.1 through 2.3

$$\text{Minimize } f(x), \qquad x = [x_1, x_2, ..., x_n]^T \in \mathbb{R}^n \tag{2.1}$$

$$g_j(x) \leq 0, \qquad j = 1, 2, ..., m \tag{2.2}$$

$$h_j(x) = 0, \qquad j = 1, 2, ..., r \tag{2.3}$$

The components of x_i are considered the variables and the optimum of vector x that solves the set of equations is denoted by x^* with an objective function value $f(x^*)$ (Snyman, 2005). Practically, the number of variables in engineering problems is often very high and analytical solutions are not possible. As a result, numerous solution algorithms that are tailored to a specific type of problem have emerged. Optimization involves the development of a mathematically accurate model and its use within an appropriate solution algorithm in an effort to find the optimum x^*.

The problem structure often found in planning and scheduling problems is a Mixed Integer Linear Program (MILP). A MILP conforms to the structure introduced in Equation 2.1 thorough 2.3 with the addition of the constraint 2.4 that states that some variables are restricted to be integer-valued.

$$x_i \in \mathbb{Z} \qquad j = 1, 2, ..., n \tag{2.4}$$

This is often due to the binary variables introduced for assigning production to a unit or worker, or deciding the temporal order of jobs, referred to as sequencing. The Branch and Bound algorithm is often used in the solution of MILPs. To explain this algorithm, an example problem is introduced in Equations 2.5 through 2.8.

$$\text{Min } f = x + 0.8y + 1.3z \tag{2.5}$$

$$\text{s.t. } x < y - 1.1 \tag{2.6}$$

$$z < x - 2.7 \tag{2.7}$$

$$x, y, z \in \{0, 1, 2, 3, 4, 5\} \tag{2.8}$$

A commercial solver, such as CPLEX, would perform a branch and bound based search that is outlined in Figure 2.3. The first step is to relax the integer constraints and obtain a continuous solution. This solution may be non-integer and forms the root node of the solution tree. The algorithm proceeds by creating two nodes that bound one of the variables with a non-integer solution to its closest integer value, with the x-variable chosen here. Using this method, one infeasible solution and one non-integer solution is found. This method is repeated for the z variable that remains non-integer. The final solution has an objective function value of 7 and all variables remain integer. The solution of a node problem provides a lower bound for all the branch subproblems that emanate from it. If this lower bound exceeds a valid upper bound obtained by an integer solution to a node problem, then the node can be

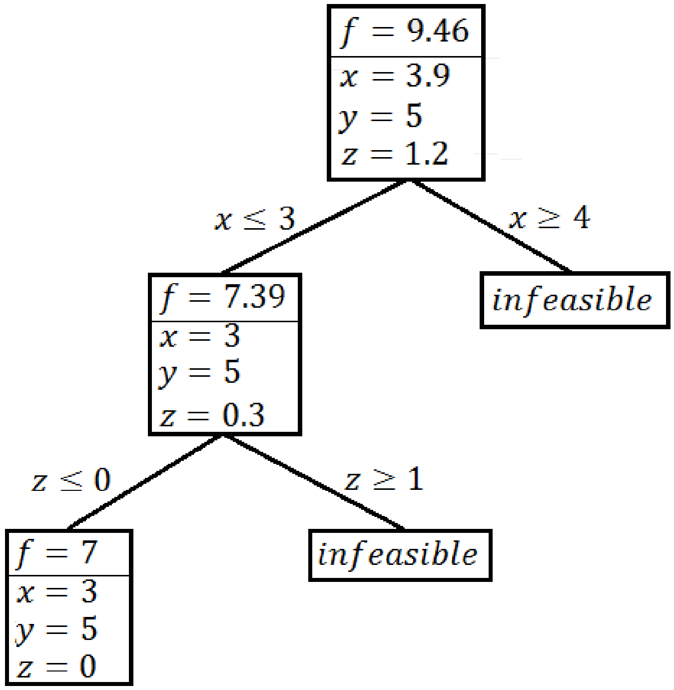

Figure 2.3: Branch and bound algorithm

eliminated. This pruning process or fathoming procedure can substantially reduce the branch and bound search space. In the next section, a solution algorithm that is fundamentally different is explored and compared to the standard MILP branch and bound procedure.

An early attempt at mathematical optimization in a steel mill was performed by Redwine and Wismer (1974) who created a deterministic MILP to minimize lateness when considering sequencing and resource availability constraints. This problem was solved for 102 orders using a Benders partitioning approach. Petersen et al. (1992) developed a strategy to synchronize the reheat furnace and the rolling mill while solving the sequencing problem using a heuristic based on a modified greedy algorithm. Jacobs et al. (1988) created a multi-objective model using a goal programming approach for the hot mill section of a steel plant that optimized the sequence of coils based on their inventory cost, profit, and marring of rollers. This model was solved using a heuristic. Wright et al. (1984) developed a mathematical programming model for the hot strip mill that was found to be too cumbersome to solve with conventional resources.

Fabian (1958) proposed a steel planning model that determined the economical usage of materials in an integrated steel mill and connected the various stages of production to form a master model. Fabian (1967) then went on to develop a production planning model for the blast furnace. Sasidhar (1991) formulated the production planning of a steel mill as a maximal flow problem in a Multiple Arc Network as well as the algorithm used to solve it while considering the prioritization of certain customer orders. Chen and Wang (1997) created a linear programming model for a Canadian steel mill that integrated steel production and distribution for one plant and several finishing factories taking into consideration raw materials, capacity allocation, customer demands, material supply, and distribution. Harjunkoski and Grossmann (2001) presented a decomposition strategy for the scheduling of a steel plant. The authors proposed to break the large original problem into subproblems using special features of steel making to avoid the need for complex constraints. The special features include breaking customer orders into groups of similar orders called heats and solving these heats separately. This method was found to work for problems up to 100 orders in size to within 3% of the global optimum.

Constraint Programming

Constraint Programming (CP) is a relatively new solution methodology to optimization problems that has emerged from constraint satisfaction research that has histor-

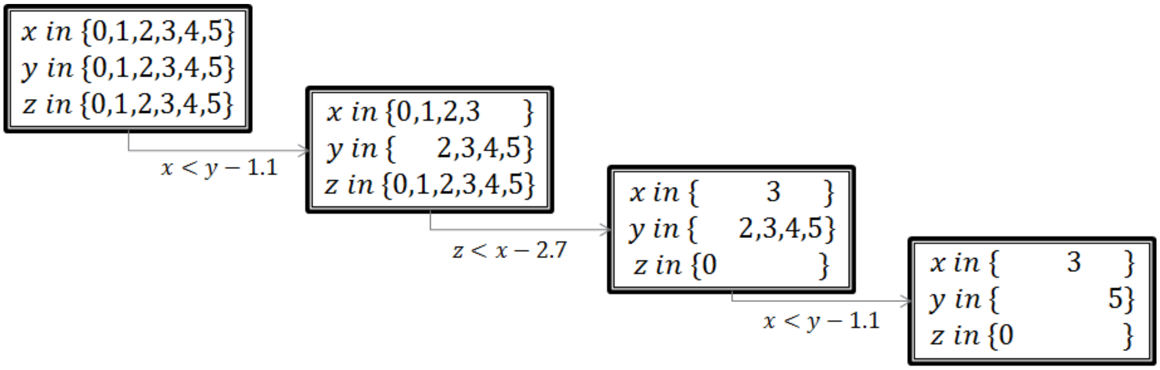

Figure 2.4: Constraint programming propagation algorithm

ically been used to solve feasibility problems. CP is described by a set of variables with a set of possible values, and a set of constraints that restricts the set of possible values of the variable called a domain (Baptiste et al., 2001). The key behind CP is that constraints can be used for more than simply testing the validity of a solution. Rather, they can be actively used to direct the search and reduce the computation effort used, called *constraint propagation*. An example of constraint propagation is provided for the same example introduced in the previous section. This can be seen in Figure 2.4. This methodology works by systematically reducing the domains of the variables according to the constraints. Equation 2.6 is applied to the initial domain to reduce the feasible search space. With this new domain, Equation 2.7 can be applied to further reduce the variables domains. Finally, Equation 2.6 can be reintroduced until the only remaining feasible solution is found. The objective function can be evaluated and the solution can be found to be identical to that generated by the Branch and Bound method. If the remaining search space was not one single solution, branch and bound or complete enumeration can be applied. The purpose of the constraint propagation step is to reduce the search space to a point where traditional complete enumeration or branch and bound is not computationally inefficient.

To solve optimization problems rather than feasibility problems, the previously found best objective function becomes a constraint in the next iteration, until all feasible combinations have been exploited. One of the many benefits of constraint programming include its expressive language, that allows boolean variables and variables that can be indexed by variables. Another key benefit of CP is the development of *global constraints*. This allows modelling to be relatively easy, and directs the solution algorithm on the most efficient route for the particular set of constraints being used. There are a number of tools that allow for CP modeling and packaged solution algorithms. Current commercial tools include ILOG SOLVER, CHIP, PROLOG IV, ECLIPSE, CLAIRE and CHOCO (Baptiste et al., 2001).

Constraint programming has successfully been used in the literature. Bisdorff and Laurent (1995) used constraint logic to model the selection of coils as a mixed goal-program. The model included capacity constraints of the mill, due dates and sequencing. The CP solution performed comparably to traditional mathematical programming solvers. Suh et al. (1998) used a constraint satisfaction heuristic in the reactive scheduling of a hot rolling mill. However, the quantity of constraint programming models in the process systems research literature is notably small.

2.2 Aggregate Production Planning

A survey of the models and methodologies used in aggregate production planning is provided in Nam and Logendran (1992). Production planning involves deciding the quantity and mix of products, inventory levels, and staffing levels over a fixed time horizon in an effort to meet anticipated consumer demand using a finite set of resources. The objective is typically a minimization of relevant costs, or a maximization of profit. Aggregation is systematically grouping production decisions into quantifiable and manageable units. Aggregate production planning as a whole falls into the mid-term planning level when viewing the hierarchy presented in Figure 1.1.

Production planning begins with an aggregation of products into broader categories. Hax and Meal (1975) have broken down aggregation into three categories: Items, Families and Types. Items are the final products and are commonly referred to as SKU's or stock keeping units. Families are items that contain a common manufacturing setup. Types are composed of similar families. The level of aggregation is not always obvious and is problem dependent. With the appropriate aggregation in place, the plan moves to a forecast of demand. The forecasts may be based on currently booked customer orders, or may be an estimate based on historical trends. The forecasts are inherently uncertain and a key assumption must be made. An average value may be assumed, or a more rigorous process such as stochastic optimization may be used. This will be discussed in the following section.

Next, an assumption must be made with regard to the finite planning horizon under consideration. An appropriate length of this horizon must be chosen to ensure that the production decisions incorporate future demand fluctuations. The end-of-horizon must also be accounted for to ensure that inventory levels are not reduced to zero due to the minimization of holding costs. Additionally, the aggregate plan typically uses a rolling horizon implementation. This means that as new information becomes available, the plan is revised and production levels may vary from the previous solution.

Finally, a number of assumptions with regard to planning costs must be made. There are well defined costs in the production planning problem. These costs include (1) smoothing, (2) bottlenecking, (3) holding, (4) shortage, (5) production, (6) idle time and (7) subcontracting costs. Smoothing refers to the costs incurred from changing production levels. Bottlenecking refers to the inability of the plant to accommodate rapid fluctuations in demand. Holding costs are a result of the opportunity cost to hold capital in the form of inventory. Shortage costs result when insufficient inventory is available to meet demand. Production costs are the cost to produce one unit of material during regular operation. Idle time arises when production lines or personnel are being under-utilized. Finally, subcontracting costs are a result of using an outside manufacturer to produce product. There are also intangible costs that are linked to management preferences or tactics. These arise when management prefers to keep hiring and firing to a minimum or to prevent disgruntling customers by meeting shipping dates even when not cost-optimal.

There are a number of articles that have investigated aggregate production planning models. Chen and Wang (1997) developed a linear programming model for a Canadian steel company while considering production costs, throughput rates, raw materials, purchasing costs, transportation costs, and distribution for multiple plants in different locations. The resulting model was used on location and found to improve efficiency and reduce information redundancy. Balakrishnan and Geunes (2003) developed a profit maximization MIP for a steel mill whose customers have flexibility in their product specifications. Their model provided solutions within 0.59% optimality and contributed to 7% additional profits for the case study company. Mohanty and Singh (1992) proposed a two level system where the higher level solves a multi-objective production planning model at in integrated steel plant. The model considered raw materials, intermediate and finished production. Finally, Zanoni and Zavanella (2005) investigated an optimal planning model for production of steel billets in a make-to-order manufacturing facility. The linear program minimizes holding costs, production costs and delayed orders while using CPLEX as the solution algorithm. The focus of the research was on the consequence of a billet cooling warehouse in the production line.

The goal of production planning is to create a plant that is able to respond faster to demand changes and exploit the flexibility of the production plant without losing overall productivity (Pochet and Wolsey). The benefits of an aggregate model are three-fold. First, it provides a means of absorbing demand fluctuations via smoothing. Second, it does not require a costly estimation of input parameters. Third, the standard deviation of the aggregate forecast error is less than the sum of its corresponding individual error resulting in a more accurate demand forecast (Nahmias and Olsen).

2.3 Short Term Batch Scheduling

This section is based on the excellent review found in Mendez et al. (2006). The author describes 13 classifications of batch scheduling problems. Any one particular problem can have any combination of these thirteen classifications. As a result, it is difficult to provide a unified solution methodology that applies to all problems. The classifications include the process topology, equipment assignment, equipment connectivity, inventory storage policy, material transfer, batch size, batch processing time, demand patterns, changeovers, resource constraints, time constraints, costs, and degree of uncertainty. The classifications are reproduced in Figure 2.5. The author also classifies the types of scheduling models that are available. This road map is reproduced in 2.6.

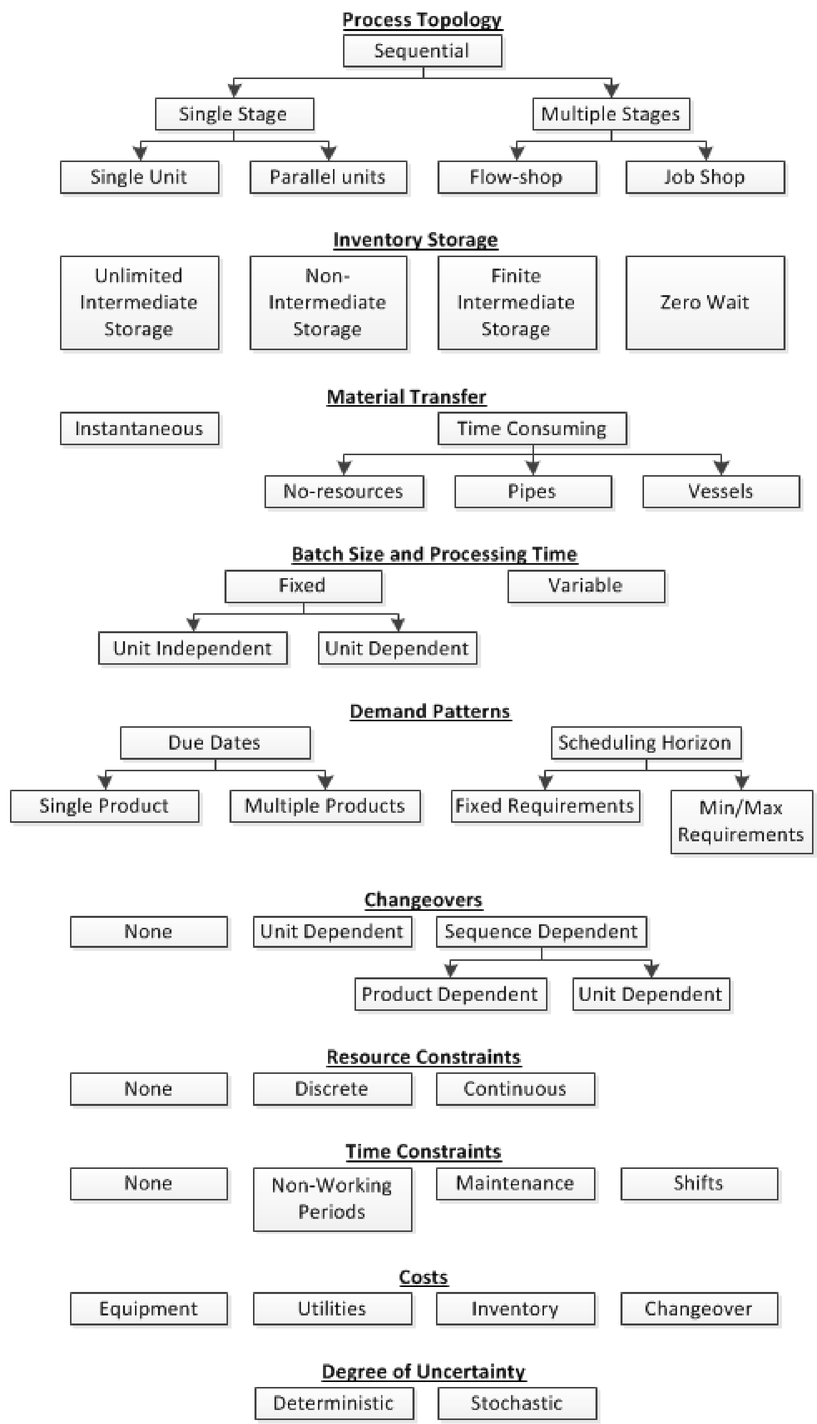

Figure 2.5: Characterization of scheduling problem features (Mendez et al., 2006)

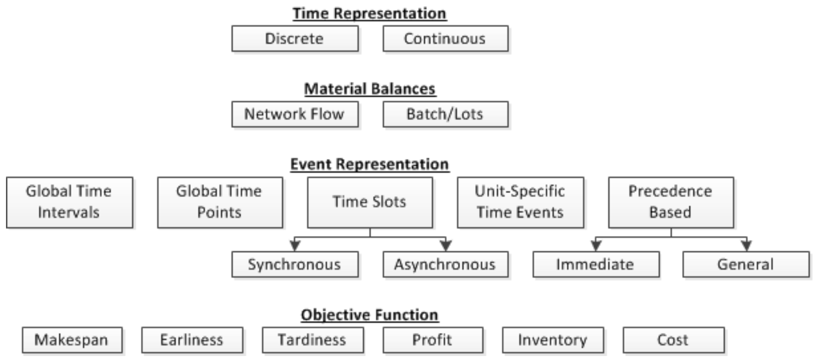

Figure 2.6: Classification of scheduling problem models (Mendez et al., 2006)

As seen in this figure, a fundamental decision in scheduling concerns the representation of time. The simplest method involves time slots with an equal and fixed duration known as the discrete time representation. Conversely, the time slots can have variable length where the slots are defined for each unit, or for the entire process as a whole. Thus is known as a continuous time representation. The former is known as a unit-based or sequential approach and the latter is known as global event based approach. Floudas and Lin (2004) provide a review on the limitations and benefits of two main classifications of scheduling problems: discrete-time and continuous-time. A summary is provided in the subsequent paragraphs.

The discrete method involves a division of the time horizon into a fixed number of uniform intervals where the events are associated with the boundaries of these time intervals. All scheduling models contain the allocation of tasks to units through the use of binary variables that determine whether a task starts in the unit at the beginning of the time interval. Mass balances are typically accounted for through the use of a state-task network which uses continuous variables to represent the amount of material undergoing a particular task at a particular time while also tracking the amount of material in each state with an additional variable. The time intervals used must be sufficiently small to achieve a suitable approximation which typically results in very large problems as the number of binary variables required scales with the number of discrete intervals.

As a result of this limitation, the continuous-time approach was developed, which allows events to occur at any time point in the time horizon. These events can either be defined globally or for each unit. As a trade-off, continuous time models are often more difficult to model due to the variable nature of the timing events. There are three subcategories of continuous time models:

1) Sequential Process: The sequential process is based on the concept of time slots. This model contains a binary variable that determines whether or not a stage of an order is assigned to a time slot of a unit. The three time variables are continuous and represent the starting and completion times of a stage of an order as well as the starting time of the slot for the unit. This model typically has a large number of binary variables due to the dimensionality needed. The modeler must select a predefined number of time slots which may be suboptimal. An iterative procedure to reduce the number of slots is required to achieve optimality. Another modelling option is to abandon the use of slots, and instead include binary variables that dictate the precedence of individual jobs or orders. Both slot based and precedence models will be investigated in this thesis.

2) Global Event Based Models: These models use continuous variables to determine event timing and variable time slots in addition to binary variables which assign the start and end of a task. Most of these models are also based on the State-Task Network concept. This is a unique method of modelling the production process as a series of state nodes that correspond to product forms, and task nodes that correspond to processing steps. These models typically lead to large MINLP formulations that

can be linearized under the appropriate assumptions. An important issue with these models is the appropriate estimation of the number of events; an underestimation leads to suboptimal or infeasible solutions while an overestimation results in large problems. This makes an iterative procedure helpful.

3) Unit-Specific Event Based Models: These models use event points that are a sequence of time instances representing the beginning of a task for each unit. Because the ending points of the tasks are not tracked, the number of binary variables required is reduced. This model also has the issue of estimating the number of events. Compared to discrete time models, this formulation typically has fewer binary variables.

In addition to time representation, Mendez et al. (2006) discuss the treatment of mass balances. Scheduling problems can be represented as either a sequential process or a network-represented process. In a sequential process, batches are used to represent production and different products follow the same sequence divided into stages (which can be single, multiple or parallel units). This model is relatively simple as mass balances do not need to be explicitly accounted for. Conversely, network-represented processes are typically more complex because batches can mix and split. As such, the network is divided into state nodes which represents a material and task nodes which represent an operation. The mass balance is conserved by depicting the percentage of state consumed per task.

The event representations that are investigated in this thesis include time slots, unit-specific immediate precedence and general precedence models. The characteristics of these models are provided in Table 2.1

	Time Slot	Unit-Specific Immediate Precedence	General Precedence
Key discrete variable	X_{ijk} defines if unit j starts task i at slot k	X_{ijm} defines if task i starts right before task i on unit m	X_{ij} defines if task i starts before task i
Critical modeling issue	Number of estimated time slots	lot-sizing and units	lot sizing

Table 2.1: Characteristics of continuous time optimization models (Mendez et al., 2006)

2.3.1 Campaign Scheduling

A campaign can be defined as, "a production run with specific start and end times in which coils of a particular type are processed continuously on a process line" Okano et al. (2004). Essentially, if a manufacturing facility has a flexible production line that is capable of producing more than one family of products, the manufacture of each family is classified as a campaign. Campaign scheduling is then the decision of the length of time allocated to producing each product family while taking into consideration the costs associated with switching between families, and the due dates of the orders allocated into each campaign. Other notable contributions in the research literature can be found in Wellons and Reklaitis (1998). The author developed a mixed integer nonlinear program (MINLP) for a multipurpose batch chemical plant. At the end of the campaign, the production line is cleaned and set up for a new product. Another example can be found in Papageorgiou (1996). In this paper, the authors developed a mathematical model that encompassed both planning and scheduling considerations including the determination and allocation of campaigns, task timing, and material flow. The authors also provide benefits of campaign operation for multipurpose plants including lower inventory costs, fewer changeovers and improved operability. Campaign operation is appropriate for plants with stable demand patterns over long term planning horizons.

2.3.2 Real-world scheduling examples

Scheduling problems found in industry are typically large and difficult to solve due to the computation complexity of combinatorial optimization problems (Mendez et al., 2006). As a result, most industrially sized problems are currently solved with alternative methods to classical MILP solution algorithms. These methods include meta-heuristics, artificial intelligence, constraint programming and forms of hybrid methodologies which combine some of these methods. Meta-heuristics employ an iterative procedure that starts with a known feasible solution and attempts to improve this solution. Many well known solution algorithms include simulated annealing, tabu search and genetic algorithms. Artificial intelligence aims to reproduce human thought. For example, one method would involve finding a previous solution to a similar problem, and modifying the solution to fit the new problem. Finally, hybrid solutions attempt to exploit the pros of one method to mitigate the cons of another. Some researchers have combined exact MILP methods with constraint programming. The MILP portion is used to solve a relaxed problem and passes this solution to CP to which quickly exploits the remaining search space.

A polymerization problem was investigated by Schulz et al. (1998). This problem included 36 batches and 360 tasks. Polymerization is non-linear which resulted in a MINLP. The authors had to use a specialized scheduling algorithm that generated a good, but suboptimal solution in a reasonable amount of time. Continuous-casting in a steel mill has also be investigated by Harjunkoski et al. (2003). The full problem size included 74,000 equations and 33,000 discrete variables. This problem was tacked with a three stage decomposition strategy and solved to within 3% optimality.

2.4 Optimization Under Uncertainty

Risk management has become an increasing focus in supply chain optimization. Process industries have turned their attention towards incorporating uncertainty into mathematical models in an effort to reduce risk (Shah, 2005). Production planning within supply chain optimization contains many areas of uncertainty. The most common forms of uncertainty occur in demand, supply and production. Demand uncertainty is a result of the inability to properly forecast customer needs. Supply uncertainty arises from disruptions in the vendors' supply of raw materials. Production uncertainty is a result of unscheduled equipment failures. All forms of uncertainty are important to consider as they affect both feasibility and optimality.

Three methods are commonly employed in industry to handle uncertainty. These methods include (1) Expected Values, (2) Wait-and-See, and (3) Stochastic programming. The Expected Value approach is minimalist in nature. It provides the optimization model with the expected average value of the uncertain parameter. This results in a computationally inexpensive model that provides acceptable results, but may also become infeasible or suboptimal when the true value of the uncertain parameters is realized. The Wait-and-See method solves an optimization program for all possible outcomes of the random parameter. This results in a set of decision vectors for each outcome. The drawback of this method is that it assumes the decision can be made after the uncertainty is realized. This is often not the case. For example, manufacturing companies with long production lead times must select production quantities well in advance of demand realization. Stochastic programming uses scenarios to represent each occurrence of uncertainty. The objective is to obtain a solution that is optimal and feasible no matter what scenario actually occurs.

Stochastic programming is broken down into two main approaches called recourse and chance constrained problems. Chance constraint programming uses a distribution of the uncertain parameter to ensure that the solution complies with the constraints within a specified confidence level. This effectively allows for the quantification of the relationship between profitability and reliability (Li et al., 2008). This work will focus on recourse problems; particularly the type known as two-staged stochastic programming with recourse. A stochastic program with recourse separates the problem into two distinct stages with separate variables according to their decision either before or after the outcome of an uncertain parameter. These are known as *first-stage* (x_1) and *second-stage* (x_2) decisions. First stage decisions are typically proactive and attempt to hedge against potential infeasibilities in the second stage. Conversely, second stage decisions react to the realized value of the uncertain parameter under each scenario. This is included in the model by the set S indexed on the x_{2s} variable. It is important to note that x_1 does not include this scenario index. This brings rise to a term used in stochastic programming called *non-anticipativity*. This term describes the concept that all first stage decisions are made independently of the true outcome of the random variable. The general model for a stochastic linear program with recourse is provided in 2.9.

$$\begin{aligned} \min\ & c_1 x_1 + \sum_{s \in S} p_s c_{2s} x_{2s} \\ \text{s.t.}\ & h(x_1) \leq 0 \\ & g(x_1) = 0 \end{aligned} \tag{2.9}$$

$$h_s(x_1, x_{2s}) \leq 0 \qquad \forall s \in S$$

$$g_s(x_1, x_{2s}) = 0 \qquad \forall s \in S$$

In this model, the first stage decision and second stage scenario dependent decisions are given a weighting factor c_1 and c_{2s} respectively to account for the probability of each scenario occurring. The probability of each scenario occurring is commonly created by a probability distribution. This distribution can be uniform, Poisson, Gaussian, etc. but is often chosen to be normal in the case of demand. The distribution is defined by the mean, μ and standard deviation, σ. This distribution is then sampled to create the scenarios. Sampling methods include Monte Carlo, Descriptive, Latin Hypercube and Hammersley Sequence Sampling (Kim and Diwekar, 2002).

To quantify the stochastic models improvement over the deterministic model, the Value of Stochastic Solution (VSS) can be used. This is defined as the difference between the objective function value of the stochastic model and the objective function value when using an expected value model. Another measure is the Expected Value of Perfect Information (EVPI). This is the value of the objective function when the demand forecasts are known precisely. It provides an upper bound on the stochastic models' usefulness.

Chapter 3

Aggregate Production Planning Optimization

The main objective of this chapter is to present aggregate production planning models of increasing complexity for a typical steel mill. Each additional layer of complexity intends to add value to the supply chain. The first part of this chapter explains the current planning method employed at a typical steel mill. The first method explained is considered a manual heuristic. The following section develops a quadratic goal programming model that attempts to replicate the methodology performed by the heuristic. This model aims to meet targets set by the upper level in the planning hierarchy. The next section incorporates uncertainty in the form of a two-stage stochastic model. This attempts to improve the result by hedging against fluctuations in demand. With the framework built to include uncertainty, it is possible to replace the objective function with one that aims to minimize overall cost. This is accomplished by driving inventory levels as close to null as possible, while hedging against the possibility of stock-outs. Finally, the last section uses multi-objective optimization as a compromise between the inventory target and cost minimization models. A comparison between the deterministic and stochastic models is presented in the first case study. An investigation into the operability and cost saving potential of the steel mill is presented in the second case study. Each section begins with an introduction to the methodology used, develops the mathematical model, and concludes with a table of nomenclature.

3.1 Application Context: Medium Term Planning Process

This thesis uses case studies and data provided by a ISP located in Hamilton, Ontario. The Medium Term Planning Process is the title given in this work to the methodology for developing the weekly production plan. This integrated steel mill consists of a large number of unit operations in series and parallel. The Medium Term Planning Process in concerned with only a subset of the full production flow. Additionally, this steel mill has thousands of stock-keeping units (SKU). A SKU is an identifier for a unique product. At the Medium Term Planning level, each SKU is aggregated into one of a dozen product families. A visual representation of the production flow with the granularity consistent at the Medium Term Planning level is provided in Figure 3.1. The unit operations under consideration in this planning process include the hot mill, two cold mills, and a number finishing lines. Before the models are introduced, it is important to understand the planning structure at the steel mill. The planning structure dictates what parameters and variables are available to the optimization

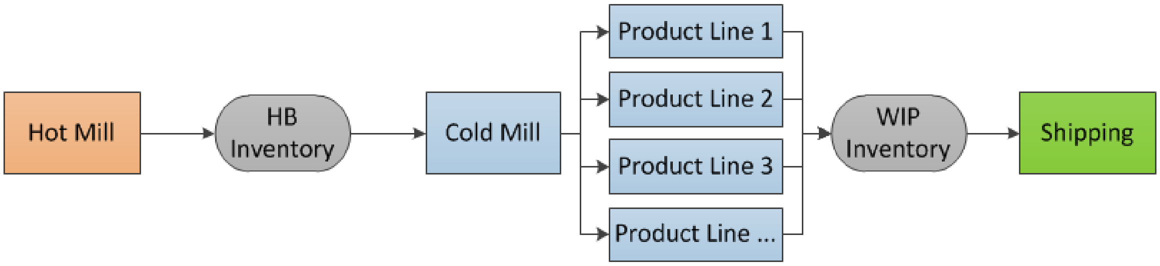

Figure 3.1: Simplified process flow diagram of a typical integrated steel mill

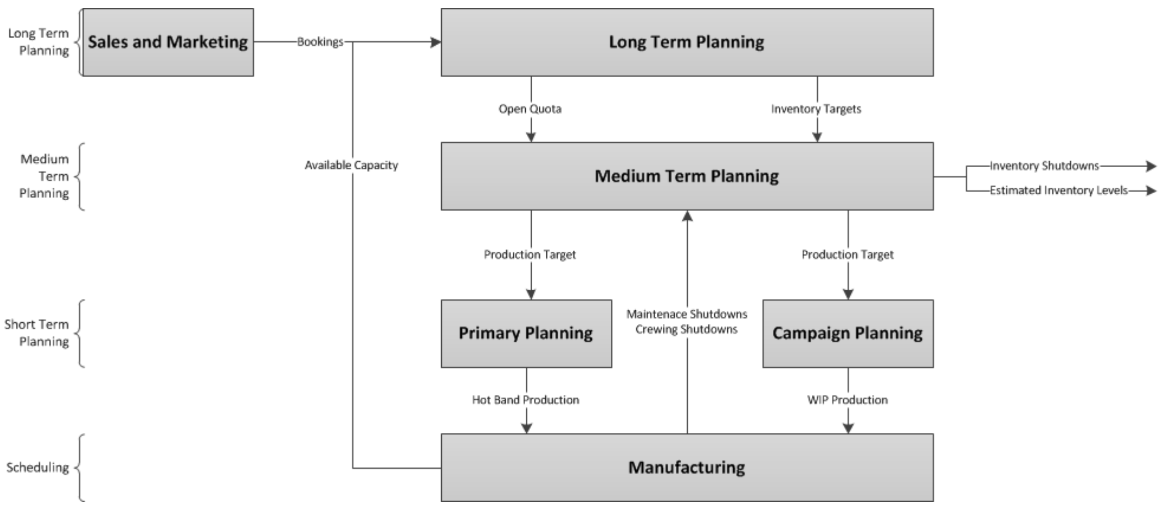

Figure 3.2: Planning hierarchy used at a typical steel mill

models. Figure 3.2 provides a visual representation of this structure. In the top-most layer lies the Long Term Planning group. This group performs strategic network design decisions. This includes a 5 year horizon of demand projects and steel industry trends. This group provides inventory level targets for the steel mill. It also keeps records of the available production capacity of the mills and the current number of customer orders procured by the Sales and Marketing group. The difference between these two numbers is called *open quota*. This information is used as an input to the Medium Term Planning Process. This process attempts to set high level production targets for the primary planning and campaign planning groups. This is accomplished by simultaneously considering the amount of open quota, current production levels, expected downtime, and current inventory levels. Primary planning is responsible for production at the casters and hot mill. Production is defined as the tonnage of steel stabs to be converted into steel coils. Campaign planning is responsible for production at the cold mills and finishing lines. This production is defined as the tonnage of steel coils to be further worked into the final product.

This chapter is focused on the Medium Term Planning level. The disjunctive nature between primary and campaign planning calls for the creation of the Medium Term Planning Process. This planning process resides one level higher than primary and campaign planning. At this level, a coarser time scale is used in an attempt to view primary and finishing operations simultaneously. The subsequent chapter will focus on the shorter time scaled Campaign Planning process.

Model Enhancements using Linear Programming Approaches

A number of enhancements to the Medium Term Planning Model have been identified and are listed below:

1. Deterministic Model Identification: The Medium Term Planning Process is solved using manual heuristics. This is a common practice in the steel industry, as the problem sizes are often prohibitively large, and there exist complicated chemistry, logistical, and scheduling constraints (Harjunkoski and Grossmann, 2001). The first enhancement is to develop a deterministic Mixed Integer Linear Program that performs similarly to the manual heuristics.

2. Stochastic Model Improvement: The time scale of the model includes a five week demand period that is predominately deterministic, followed by a 5 week

demand period that is predominately uncertain. Due to the amount of operator time required to perform one iteration of the manual heuristic, it is impossible to consider more than an average, high, and low scenario of demand. A two-stage stochastic model is built to consider hundreds of scenarios simultaneously and make the best possible choice to hedge against future uncertainty.

3. Cost-Based Integration: The Long Term Planning group provides inventory targets to the Medium Term Planning team which they believe best suit the current market conditions. The planning teams goal is to minimize the distance between these targets and the projected inventory levels at the end of the week. An enhancement made here is to move to a cost-based objective function that minimizes the total cost of production including a variable for the optimal level of inventory.

Each of these enhancements will be addressed in the following sections with the development of mathematical optimization models.

3.2 Deterministic Aggregate Production Planning Model

The Deterministic Aggregate Production Planning (DAPP) model is a conversion of the manual process employed by experienced operators into a mathematical model that is solvable using optimization algorithms. It is considered deterministic due to the assumptions made. The first assumption is that the demand is known and that exactly one half of the open quota will be filled. For example, if 1,000 tons of

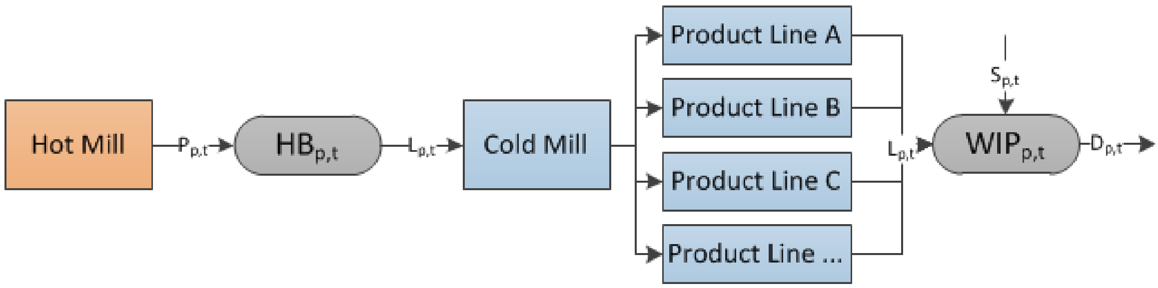

Figure 3.3: Information flow required to make production decisions

product A was currently booked in week 9, and the plant has a capacity of 2,000 tons of product A in week 9, than the assumed demand for that week is equal to 1,500 tons. The other main assumption in this model is that the hot/cold mills and finishing lines produce exactly as much material as outlined in the plan. The plan is considered an aggregation since the inventory levels and production amounts are divided into weekly time periods when in actuality, these levels fluctuate minute-to-minute. A visual representation of the key variables is shown in Figure 3.3. The Hot Mill produces Hot Band of amount $P_{p,t}$ which is stored in inventory as variable $HB_{p,t}$. The Hot Band product is then launched through the cold mills and product lines designated by the variable $L_{p,t}$. The product line is a general term to represent some sort of finishing operation used to create value-added steel. This can include annealing furnaces, galvanization coatings, slitting, tubing, etc. No matter the route, all products end up in the $WIP_{p,t}$ inventory. Since this is an aggregate production plan with a medium term horizon, this sort of granularity is acceptable. Substrate $S_{p,t}$ is material that is purchased from another facility and is stored in the $WIP_{p,t}$ inventory. The actual mill has other production units following this inventory storage that prepares the material for shipping, however they are not included in the model. As such, demand $D_{p,t}$ is drawn directly from the $WIP_{p,t}$ inventory. A table of the additional nomenclature is available at the end of this section.

3.2.1 Constraints

The first equation used as the basis for all production planning models is a mass balance for the inventory. Since there are two inventory storage locations, this model requires balances on the WIP material and the Hot Band Material. Equation 3.1 represents the balance on the former for a majority of the products.

$$WIP_{p,t} = WIP_{p,t-1} + L_{p,t} + S_{p,t} - D_{p,t} \qquad \forall p \in P \backslash Type1, t \in T \tag{3.1}$$

This equation states that the current WIP inventory level ($WIP_{p,t}$) is equal to the previous time periods inventory level ($WIP_{p,t-1}$), plus all material produced ($L_{p,t}$) or bought ($S_{p,t}$), minus all material consumed as demand ($D_{p,t}$). The domain of in inventory WIP inventory levels is restricted to be strictly positive in Equation 3.2.

$$WIP_{p,t} \geq 0 \qquad \forall p \in P, t \in T \tag{3.2}$$

However, this creates infeasibilities in the model when the consumer demand is greater than what is available to be produced and stored in inventory. For this reason, the $WIP_{p,t}$ variable can be broken down into its positive and negative components as shown in Equation 3.3.

$$WIP_{p,t} = WIP^{+}_{p,t} - WIP^{-}_{p,t} \qquad \forall p \in P, t \in T \tag{3.3}$$

$$WIP^{+}_{p,t}, WIP^{-}_{p,t} \geq 0 \qquad \forall p \in P, t \in T \tag{3.4}$$

The $WIP^{-}_{p,t}$ term is commonly referred to in the operations research community as back order. For simplicity, the $WIP^{-}_{p,t}$ term can be called $B_{p,t}$ and $WIP^{+}_{p,t}$ can simply become $WIP_{p,t}$. This concept can be included in Equation 3.1 to become the first inventory constraint in the model as shown in Equation 3.5.

$$(WIP_{p,t} - B_{p,t}) = (WIP_{p,t-1} - B_{p,t-1}) + L_{p,t} + S_{p,t} - D_{p,t} \qquad \forall p \in P \backslash Type1, t \in T \tag{3.5}$$

It is important to note that the set P is in reference to both the product and production line. For example, production line B creates product B. For the previous constraint, there is a single production line for every product. For the following constraint in Equation 3.6, three production lines produce one product called "Type1".

$$\sum_{p:p \in PG} WIP_{p,t} - \sum_{p} B_{p,t} = \sum_{p:p \in PG} WIP_{p,t-1} - \sum_{p} B_{p,t-1} + \sum_{p:p \in PG} L_{p,t} - D_{Type1,t} \qquad \forall t \in T \tag{3.6}$$

For this product, the only consideration is the inventory balance on the sum of these three lines.

Equation 3.7 is the inventory balance for the Hot Band inventory. There is no back order or product purchased from outside producers.

$$HB_{p,t} = HB_{p,t-1} + P_{p,t} - L_{p,t} \qquad \forall p \in P, t \in T \tag{3.7}$$

Equation 3.8 and 3.9 are capacity restrictions on the amount of WIP and Hot Band material produced respectively.

$$\sum_{p} L_{p,t} \leq CRC_t \qquad \forall t \in T \tag{3.8}$$

$$\sum_{p} P_{p,t} \leq HBC_t \qquad \forall t \in T \tag{3.9}$$

Equation 3.10 and 3.11 are capacity restrictions on the amount of WIP and Hot Band material stored in inventory.

$$\sum_p HB_{p,t} \leq HBIC \qquad \forall t \in T \tag{3.10}$$

$$\sum_p WIP_{p,t} \leq WIPIC \qquad \forall t \in T \tag{3.11}$$

Equation 3.12 is a unique constraint consistent with the steel mill operation which restricts the amount of material launched through the finishing lines based on the amount of downtime scheduled to occur during the week. The parameter DR_p^f is the maximum daily rate that each production line p is capable of with units of tons. This number is divided by 24 to turn the daily rate into an hourly rate. The term inside the parenthesis represents how many of the 168 hours in a week are available to be allocated to production. If there are any maintenance hours scheduled for the week, they reduce this number by $MD_{p,t}$. Similarly, if the crew is not fully staffed, the number of hours is reduced in the form of crewing downtime ($CD_{p,t}$). If the number of remaining hours at the full production rate will put the inventory levels above target, additional downtime can be taken in the form of inventory downtime ($ID_{p,t}$). This is a continuous variable with units in hours.

$$L_{p,t} = (\frac{DR_p^f}{24})(168 - MD_{p,t} - CD_{p,t} - ID_{p,t}) \qquad \forall p \in P, t \in T \tag{3.12}$$

The set of products $p \in P$ must travel through the cold mill before being sent to their respective production line. Equation 3.13 serves the same purpose as above but instead for the two cold mills in set m.

$$\sum_{p:p\in CM} L_{p,t} = \sum_{m} (\frac{DR^c_m}{24})(168 - MD^c_t - CD^c_t - ID^c_t) \qquad \forall t \in T \tag{3.13}$$

Equation 3.14 serves the same purpose for the production of steel through the hot mill.

$$\sum_{p} P_{p,t} = (\frac{DR^h}{24})(168 - MD^h_t - CD^h_t - ID^h_t) \qquad \forall t \in T \tag{3.14}$$

Equation 3.15 initializes the back order of the products to zero.

$$B_{p,0} = 0 \qquad \forall p \in P \tag{3.15}$$

The remaining equation restricts the domain of the continuous variables.

$$B_{p,t}, WIP_{p,t}, HB_{p,t}, L_{p,t}, P_{p,t}, ID_{p,t} \geq 0 \qquad \forall p \in P, \forall t \in T \tag{3.16}$$

3.2.2 Objective Function

The main objective used in this model is minimizing the distance from the inventory set point, or target provided by the long term planning group. This group provides a target for the Work-In-Progress material ($WIP^{tar}_{p,t}$) for all products and all time periods as well as a similar parameter for the Hot Band material ($HB^{tar}_{p,t}$). The inventory level variables for the Work-In-Progress material and Hot Band material are $WIP_{p,t}$ and $HB_{p,t}$, respectively. This follows the definition of a goal program and is shown in Equation 3.17.

$$\text{Min} \sum_p \sum_t [(WIP_{p,t} - WIP_{p,t}^{tar})^2 + (HB_{p,t} - HB_{p,t}^{tar})^2] \tag{3.17}$$

$$+ X(L,P) + Y(ID) + \sum_p \sum_t 10^{10} B_{p,t} \tag{3.18}$$

In order to penalize deviations that are both above and below target, a squared objective function is used. This also has the intended consequence of penalizing large deviations from target more heavily than small ones. This objective function additionally includes a term for the prioritization of multiple production lines that are capable of producing identical product ($X(L,P)$) and a prioritization of the inventory downtime taken at the cold mills ($Y(ID)$). Finally the objective function minimizes the back order variable $B_{p,t}$ with a sufficiently large coefficient to constraint it to zero when feasible.

Equation 3.19 is a prioritization of product lines that are capable of producing the same product. Line C is the most cost efficient, so its use is prioritized by a smaller penalty coefficient when compared to Lines A and B.

$$X(L,P) = \sum_t 100L_{LineA,t} + 10L_{LineB,t} + 1L_{LineC,t} + \sum_t 100P_{LineA,t} + 10P_{LineB,t} + 1P_{LineC,t} \tag{3.19}$$

Equation 3.20 is a prioritization of the cold mill usage. It is preferred that the first cold mill receives as much load as possible before the second cold mill is active. This is accomplished by minimizing the amount of downtime that is taken at the first cold mill.

$$Y(ID) = \sum_t [1000ID_{CM1,t} + 1ID_{CM2,t}] \tag{3.20}$$

Section Nomenclature

Indicies

$p \in P$	all products and production lines
$p \in Type1$	A certain type of products (sub-set of P)
$p \in CM$	cold mill products (sub-set of P)
$m \in M$	cold mills
$t \in T$	time period

Parameters

CRC_T	cold rolling mill throughput limit
HBC_T	hot mill throughput limit
$HBIC$	hot band strict inventory holding limit
DR_P^f	daily production rate at finishing mill
DR_C^c	daily production rate at cold mill
DR_P^h	daily production rate at hot mill
$MD_{P,T}$	maintenance down hours
$CD_{P,T}$	crewing down hours
$WIP_{P,T}^{tar}$	work in progress inventory level target
$HB_{P,T}^{tar}$	hot band inventory level target
$S_{P,T}$	substrate relief

Variables

$WIP_{P,T}$	work in progress inventory
$HB_{P,T}$	hot band inventory
$P_{P,T}$	hot band production
$L_{P,T}$	WIP production (launch)
$ID_{P,T}$	inventory down hours per product
$ID^c/h_{P,T}$	inventory down hours per mill
$B_{P,T}$	backorder amount
$D_{P,T}$	demand

3.3 Stochastic Aggregate Production Planning Model

The Stochastic Aggregate Production Planning (SAPP) model aims to enhance the deterministic Medium Term Production Planning Model. A procedure that recalculates the Medium Term Planning model manually for different independent scenarios is is susceptible to suboptimality and infeasibility when the actual realized demand is far from average. It is the goal of this section to implement a mathematical model that will use historical demand data to create a distribution that is either uniform or normally distributed. This demand distribution will then be sampled via Monte Carlo methods to create a number of scenarios. These scenarios will then be incorporated into the model shown below which creates a solution that hedges against demand uncertainty.

3.3.1 Objective Function

The objective function remains the same with the exception of the additional subscript s to indicate that the variables are different under each scenario. Additionally, This is shown in Equation 3.21.

$$\text{Min}(1/N)\sum_{p}\sum_{t}\sum_{s}(WIP_{p,t,s} - WIP_{p,t}^{tar})^2 + (HB_{p,t,s} - HB_{p,t}^{tar})^2 \tag{3.21}$$

$$+ (1/N)\sum_{t}\sum_{s} 100L_{LineA,t,s} + 10L_{LineB,t,s} + 1L_{LineC,t,s} \tag{3.22}$$

$$+ (1/N)\sum_{t}\sum_{s} 100P_{LineA,t,s} + 10P_{LineB,t,s} + 1P_{LineC,t,s} \tag{3.23}$$

$$+ (1/N)\sum_{t}\sum_{s} 1000ID_{CM1,t,s} + 1ID_{CM2,t,s} \tag{3.24}$$

$$+ (1/N)\sum_{p}\sum_{t}\sum_{s} 10^{10}B_{p,t,s} \tag{3.25}$$

Each equation is multiplied by $(1/N)$ to indicate that each scenario is equally weighted.

3.3.2 Constraints

The constraints remain identical to the previous model with the exception of the set s for the appropriate variables.

$$WIP_{p,t,s} - B_{p,t,s} = WIP_{p,t-1} - B_{p,t-1,s} + L_{p,t,s} - D_{p,t,s} + S_{p,t} \qquad \forall p \in P \backslash Type1, t \in T, s \in S \tag{3.26}$$

$$\sum_{p:p\in PG} WIP_{p,t,s} - \sum_{p} B_{p,t,s} = \sum_{p:p\in PG} WIP_{p,t-1,s} - \sum_{p} B_{p,t-1,s} + \sum_{p:p\in PG} L_{p,t,s} - D_{Type1,t,s} \qquad \forall t \in T, s \in S \tag{3.27}$$

$$HB_{p,t,s} = HB_{p,t-1,s} + P_{p,t,s} - L_{p,t,s} \qquad \forall p \in P, t \in T, s \in S \tag{3.28}$$

$$\sum_{p} L_{p,t,s} \leq CRC_t \qquad \forall t \in T, s \in S \tag{3.29}$$

$$\sum_{p} P_{p,t,s} \leq HBC_t \qquad \forall t \in T, s \in S \tag{3.30}$$

$$\sum_{p} HB_{p,t,s} \leq HBIC \qquad \forall t \in T, s \in S \tag{3.31}$$

$$\sum_{p} WIP_{p,t,s} \leq WIPIC \qquad \forall t \in T, s \in S \tag{3.32}$$

$$L_{p,t,s} = (\frac{DR_p^f}{24})(168 - MD_{p,t} - CD_{p,t} - ID_{p,t,s}) \qquad \forall p \in P, t \in T, s \in S \tag{3.33}$$

$$\sum_{p:p \in CM} L_{p,t,s} = \sum_{m} (\frac{DR^c}{24})(168 - MD_t^c - CD_t^c - ID_{t,s}^c) \qquad \forall t \in T, s \in S \tag{3.34}$$

$$\sum_{p} P_{p,t,s} = (\frac{DR^h}{24})(168 - MD_t^h - CD_t^h - ID_{t,s}^h) \qquad \forall t \in T, s \in S \tag{3.35}$$

$$B_{p,0,s} = 0 \qquad \forall p \in P, s \in S \tag{3.36}$$

$$B_{p,t,s}, WIP_{p,t,s}, HB_{p,t,s}, L_{p,t,s}, P_{p,t,s}, ID_{p,t,s} \geq 0 \qquad \forall p \in P, \forall t \in T, \forall s \in S \tag{3.37}$$

Depending on the product being made, it can take up to three weeks for the material to move from the planning stage to the point when it is ready to be shipped. This is defined as the lead time for each product. As a result of this lead time, cold mill production decisions for this product must be made one week ahead of time, and hot band production decisions must be made three weeks ahead of time. These decisions are locked in and cannot change in light of new demand information being provided by the sales and marketing team. As such, these variable values become the first-stage decisions of the two-stage model. The remaining uncertain weeks production decisions are the second-stage. Mathematically, this is incorperated through additional constraints known as non-anticipativity. This is the key feature that states that all scenarios must use the same cold mill production numbers during the first week and is shown in Equation 3.38. Similarly, 3.39 is for the first three weeks at the Hot Mill.

$$L_{p,1,s} = L_{p,1,1} \qquad \forall p \in P, s \in S \tag{3.38}$$

$$P_{p,1,s} = P_{p,n,s} \qquad \forall p \in P, s \in S, n \in 1,2,3 \tag{3.39}$$

3.3.3 Scenario Generation

The possible scenarios of demand are based on the difference between the current level of booked orders and the maximum capacity that can be allocated. For the SAPP model, a normal distribution is fit with mean equal to the center of the open quota. The full width of the open quota is equal to three standard deviations from the norm which represents 99.7% of all values. Figure 3.4 is a visual representation of this process. As can be seen, the size of the distribution is small in the initial weeks, and grows in general during later weeks. If a sample is taken that falls outside of this range, it will be replaced with the closest bounded value. For open quotas that are zero or smaller (occasionally sales overbooks the plant capacity by a small margin), the sampled value will be exactly equal to the mean. For the products under consideration, and ten week time frame, there are 110 unique distributions that need to be sampled. Figure 3.5 shows a histogram of the resulting scenarios of the distributions sampled when only 10 samples are used. From this figure it can be seen that 10 scenarios is not large enough to replicate the true normal distribution. However, when 100 scenarios are used, the distribution is replicated with reasonable accuracy as shown in Figure 3.6. The computation time required to solve the SAPP

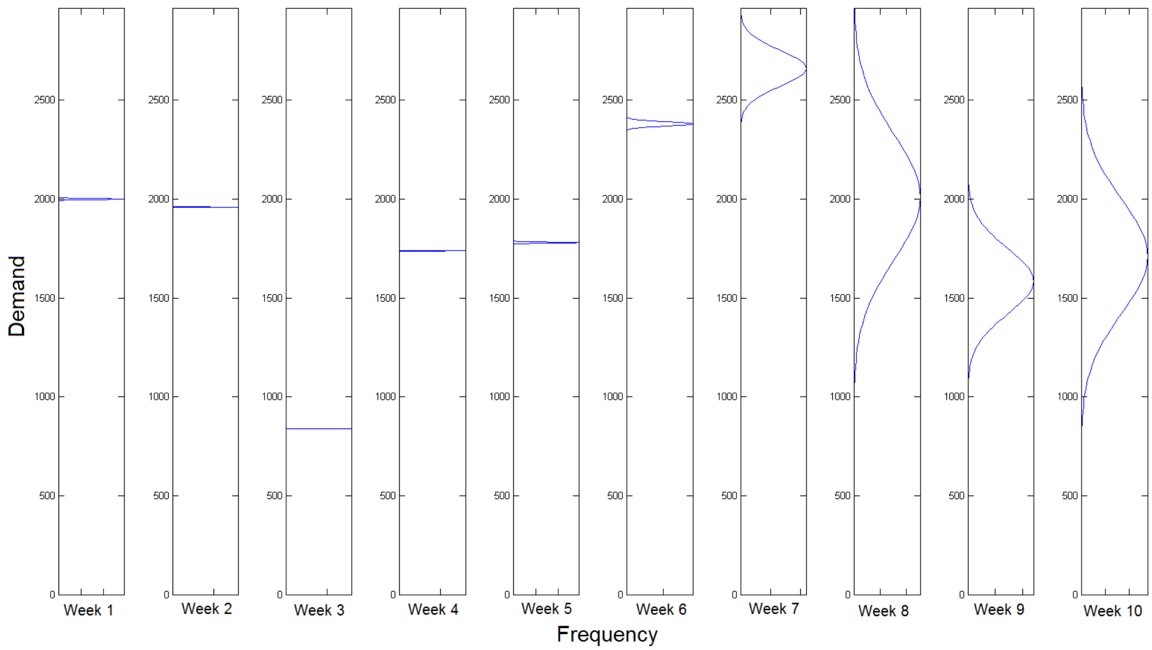

Figure 3.4: Demand uncertainty distribution for product G

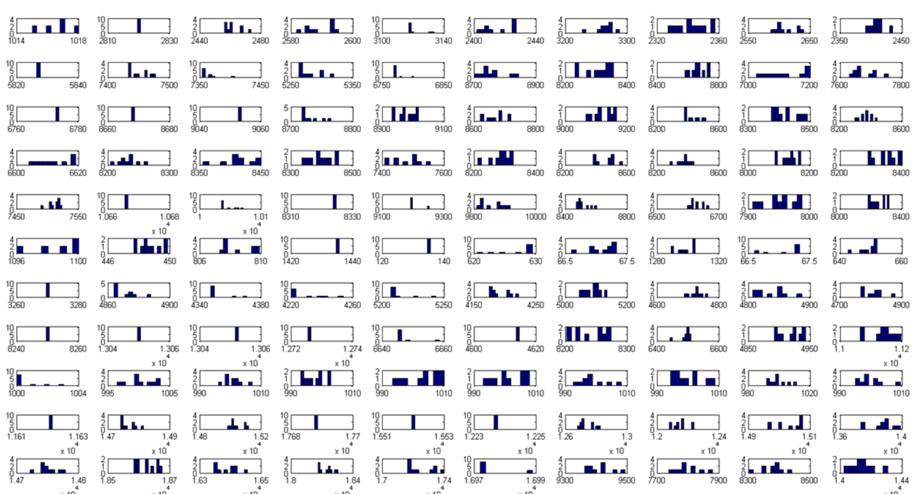

Figure 3.5: Histogram of sampled demand profile for n=10 scenarios

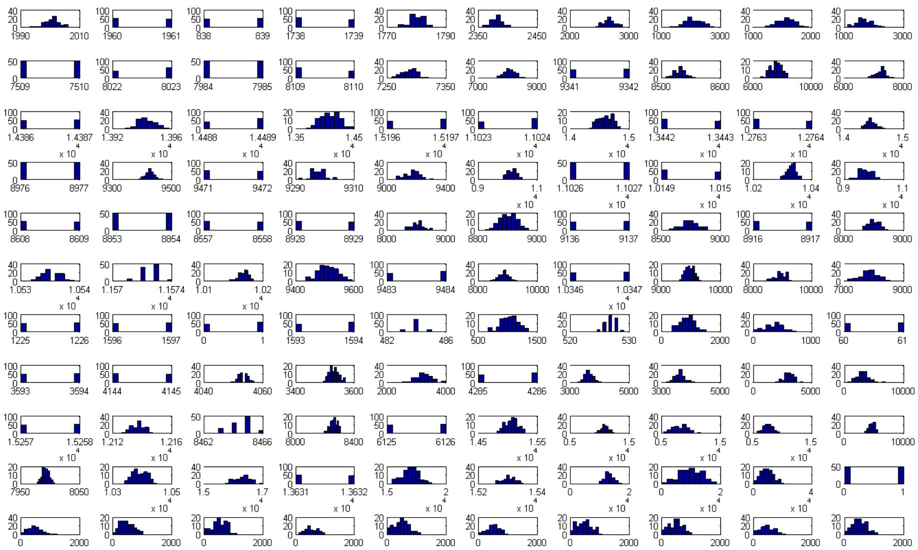

Figure 3.6: Histogram of sampled demand profile for n=100 scenarios

model is a function of the number of scenarios used. Using more than 100 scenarios does not provide any large margin of improvement in the distribution and significantly increases the solution time required. As such, 100 scenarios is the basis used for the first case study. A graph of the solution times is provided in the case study.

3.4 Cost-Based Stochastic Aggregate Production Planning Model

The purpose of this section is to develop a mathematical model that attempts to minimize the cost of producing steel as opposed to meeting inventory targets reffered to here as the Cost-Based Stochastic Aggregate Production Planning (CBSAPP) model. This modelling ideology stems from the fact that the steel mill only needs to keep as much inventory available to handle the maximum value of uncertainty. It is hypothesized that more inventory is being held than necessary to absorb this uncertainty. With a cost-based stochastic model, inventory levels can be reduced while still meeting customer delivery dates.

In order to create this model, a number of parameters must be estimated, and assumptions made. The results of the optimization are largely dependent on the quality of these estimations and assumptions. The parameters that require estimation include holding costs, back order costs and production costs. With the necessary inclusion of production costs, additional variables to track the modes of production at the plant are also required. Due to limitations at some unit operations and the sequential nature of the mills, the plant may only be operated at full, half, or null capacity. Changing between modes has a fixed switching cost. Additionally, each of these production modes has a unique marginal cost of production that must be considered.

Holding costs are defined as the money required to store a unit of inventory, typically for one year. The factors that contribute towards holding cost are numerous including the cost of money, taxes, insurance, warehousing, physical handling, inventory control, obsolescence, and deterioration. Table 3.1 shows the estimated range of contribution of each of these categories as a percentage of the selling cost. Textbooks use the range between 12% and 34% of the selling price of the product (Berling, 2008). The model used in this thesis will assume a value of 30% of the selling price. The back order cost has units of dollars per week per ton ($\frac{\$}{wk \cdot ton}$).

It is well known in the literature that it is difficult to estimate back order costs due to intangible factors like the loss of goodwill. The tangible components include administration fees when the customer is retained, and the cost of a lost sale when the

Cost of Money	6%-12%
Taxes	2%-6%
Insurance	1%-3%
Warehouse	2%-5%
Physical handling	2%-5%
Inventory Control	3%-6%
Obsolescence	6%-12%
Deterioration	3%-6%
Total Cost (per year)	25%-55%

Table 3.1: Estimated category contribution towards holding cost (Richardson, 1995)

customer is not. It is reasonable to assume that the back order cost must be larger than the holding cost or else no company would hold inventory at all. Additionally, it is relatively difficult for a steel consumer to switch steel suppliers without incurring significant costs and changes in product quality and consistency. For this reason, back order is assumed to be 50% of the selling price of the product for every week that the product is late. The units on the back order cost coefficient are identical to holding cost coefficient.

The production cost is of the linear form $y = mx + b$. This allows for a fixed cost to start up each production line, as well as a marginal cost of production per ton of steel. The production facility is assumed to have 3 operation modes, at full, half and zero capacity. Each of these modes has its own linear cost function. Specifically, each production line has its own start up costs dependent on the unit operations involved. However, this information is not available for the steel mill under question. The assumptions made regarding the linear production costs are provided in Table 3.2. The marginal cost of production is assumed to be 15% of the overall selling price of the steel when the plant is at full capacity. At half capacity, the y-intercept value is $60,000. This is more than half of the full capacity value as there is fixed costs

Capacity	m	b
Zero	-	\$10,000
Half	16.5%	\$60,000
Full	15%	\$100,000

Table 3.2: Linear cost coefficient for different plant capacities of the form $y = mx + b$

that cannot be avoided. The marginal cost of production is set at 16.5%. At zero capacity, the y-intercept value is \$10,000.

The final assumption is made with regard to production uncertainty. Since this model is taking into consideration production costs, it is logical to also include the possibility that the amount of material requested to be produced may not match the actual production values. This situation may arise due to unexpected maintenance shutdowns or the batch nature of the casting process. In order to handle this uncertainty, an additional parameter is added that is normally distributed with a mean of zero and a standard deviation that allows the actual production to be within ±10% of the full plant capacity 99.7% (three standard deviations) of the time. This is allowable as the capacity constraints can often be violated if required. A sample of the created normal distributions for one data set is provided in Figure 3.7. The distributions are sampled in the scenario generation phase and this value is incorporated into the mass balance for both the hot mill and finishing lines as will be shown in the next section.

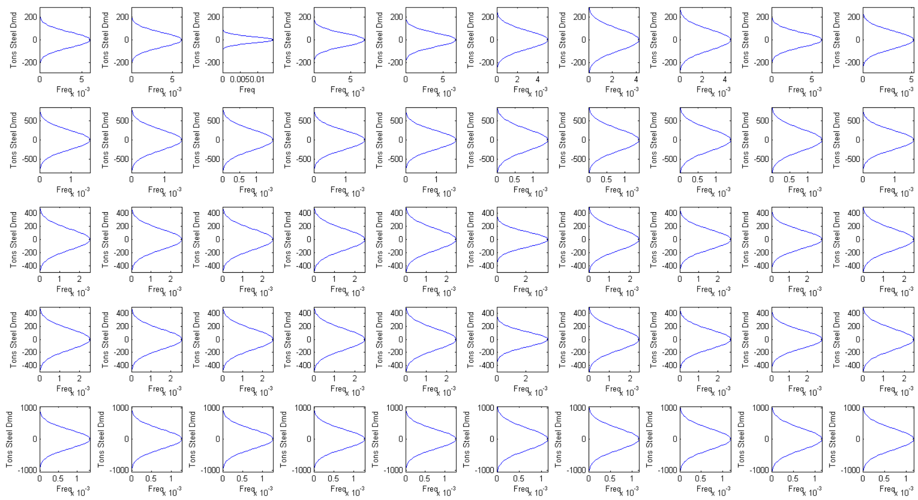

Figure 3.7: Noise distributions in production for a sample data set

3.4.1 Objective Function

The objective function of the SAPP model was the deviation between the current inventory levels and the inventory targets provided by the Long Term Planning group that was to be minimized. Under the new CBSAPP model, the objective function becomes a weighted minimization of costs based on the individual contribution of holding costs, back order costs, startup costs and marginal production costs.

$$
\begin{aligned}
\text{Min } & \sum_p \sum_t \sum_s hc_p(WIP_{p,t,s} + HB_{p,t,s}) \\
& + \sum_p \sum_t \sum_s bc_p(B_{p,t,s}) \\
& + \sum_p \sum_t \sum_s 100000F^f_{p,t,s} + 60000F^h_{p,t,s} + 10000F^n_{p,t,s} \\
& + \sum_p \sum_t \sum_s 100000H^f_{p,t,s} + 60000H^h_{p,t,s} + 10000H^n_{p,t,s}
\end{aligned} \tag{3.40}
$$

$$+\sum_{p}\sum_{t}\sum_{s} sp_p \frac{DR_p}{24}(0.15Fh^f_{p,t,s} + 0.165Fh^h_{p,t,s})$$
$$+\sum_{p}\sum_{t}\sum_{s} sp_p \frac{DR_p}{24}(0.15Hh^f_{p,t,s} + 0.165Hh^h_{p,t,s})$$

Equation 3.40 starts with a minimization of the holding costs for work in progress and hot band inventory storage. The second line minimizes the back order costs. The third line represents a minimization of the start up costs. The variable $F^f_{p,t,s}$, $F^h_{p,t,s}$, and $F^n_{p,t,s}$ represent binary variables for starting up the Finishing line p at full, half, and no capacity respectively. The fourth line performs a similar function for the hot mill. The fifth line represents the marginal cost of production. The sp_p parameter represents the selling cost of each of the products. The variables $Fh^f_{p,t,s}$ and $Fh^h_{p,t,s}$ represent the number of hours of production that the mills operate at full and half capacity respectively. This number is multiplied by the hourly rate to give units in tons rather than hours. The final line performs a similar function for the hot mill production.

3.4.2 Constraints

There are a number of constraints that remain the same when compared to that of the previous models. Equations 3.41 and 3.42 represent the mass balances for the system. These equations now include $n^f_{p,t,s}$ and $n^h_{p,t,s}$ to represent the uncertainty in production. These parameters have units of tons and are allowed to be both negative and positive.

$$WIP_{p,t,s} - B_{p,t,s} = WIP_{p,t-1,s} - B_{p,t-1,s} + L_{p,t,s} + n^f_{p,t,s} - D_{p,t,s} \qquad \forall p \in P, t \in T, s \in S \tag{3.41}$$

$$HB_{p,t,s} = HB_{p,t-1,s} + P_{p,t,s} + n^h_{p,t,s} - L_{p,t,s} - n^f_{p,t,s} \qquad \forall p \in P, t \in T, s \in S \quad (3.42)$$

Equations 3.43 through 3.45 are limitations in inventory and production levels. They remain unchanged from previous models.

$$\sum_p L_{p,t,s} \leq CRC_t \qquad \forall t \in T, s \in S \quad (3.43)$$

$$\sum_p P_{p,t,s} \leq HBC_t \qquad \forall t \in T, s \in S \quad (3.44)$$

$$\sum_p HB_{p,t} \leq HBIC \qquad \forall t \in T, s \in S \quad (3.45)$$

Equations 3.46 and 3.47 replace 3.33 from the SAPP model. This addition was necessary to account for the splitting of hours at full and half production.

$$L_{p,t,s} = \frac{DR_p}{24}(Fh^f_{p,t,s} + \frac{Fh^h_{p,t,s}}{2}) \qquad \forall p \in P, t \in T, s \in S \quad (3.46)$$

$$Fh^f_{p,t,s} + Fh^h_{p,t,s} + Fh^n_{p,t,s} = 168 - MD_{p,t} - CD_{p,t} \qquad \forall p \in P, t \in T, s \in S \quad (3.47)$$

Equation 3.47 no longer contains the variable $ID_{p,t,s}$. This function has been replaced by the variable $Fh^n_{p,t,s}$ which is functionally equivalent. Taking an inventory downtime is equivalent to an hour of production at no capacity. Equation 3.48 and 3.49 perform a similar function for the cold mills.

$$\sum_{p \in CM} L_{p,t,s} = \frac{DR^c}{24}(Ch^f_{t,s} + \frac{Ch^h_{t,s}}{2}) \qquad \forall t \in T, s \in S \quad (3.48)$$

$$Ch^f_{p,t,s} + Ch^h_{p,t,s} + Ch^n_{p,t,s} = 168 - MD^c_t - CD^c_t \qquad \forall t \in T, s \in S \quad (3.49)$$

Equation 3.50 and 3.51 perform a similar function for the hot mills.

$$\sum_p P_{p,t,s} = \frac{DR^h}{24}(Hh^f_{t,s} + \frac{Hh^h_{t,s}}{2}) \qquad \forall t \in T, s \in S \tag{3.50}$$

$$Hh^f_{p,t,s} + Hh^h_{p,t,s} + Hh^n_{p,t,s} = 168 - MD^h_t - CD^h_t \qquad \forall t \in T, s \in S \tag{3.51}$$

Equations 3.52, 3.53, and 3.54 restrict the number of hours at full production to zero when the production line has not been set up.

$$Fh^f_{p,t,s} \leq 168F^f_{p,t,s} \qquad \forall t \in T, s \in S \tag{3.52}$$

$$Ch^f_{p,t,s} \leq 168C^f_{p,t,s} \qquad \forall t \in T, s \in S \tag{3.53}$$

$$Hh^f_{p,t,s} \leq 168H^f_{p,t,s} \qquad \forall t \in T, s \in S \tag{3.54}$$

Equations 3.55, 3.56, and 3.57 perform a similar function at half capacity.

$$Fh^h_{p,t,s} \leq 168F^h_{p,t,s} \qquad \forall t \in T, s \in S \tag{3.55}$$

$$Ch^h_{p,t,s} \leq 168C^h_{p,t,s} \qquad \forall t \in T, s \in S \tag{3.56}$$

$$Hh^h_{p,t,s} \leq 168H^h_{p,t,s} \qquad \forall t \in T, s \in S \tag{3.57}$$

Equation 3.58 set the initial back order to zero.

$$B_{p,0,s} = 0 \qquad \forall p \in P, s \in S \tag{3.58}$$

Equations 3.59 and 3.60 set the non-anticipativity for the first one and three weeks respectively.

$$L_{p,1,s} = L_{p,n,s} \qquad \forall p \in P, s \in S, n \in 1 \tag{3.59}$$

$$P_{p,1,s} = P_{p,n,s} \qquad \forall p \in P, s \in S, n \in 1,2,3 \tag{3.60}$$

Section Nomenclature

The cost-based model uses the following additional parameters and variables:

Parameters

$n^F_{p,t,s}$	production uncertainty at the finishing lines
$n^H_{p,t,s}$	production uncertainty at the hot mill
hc_p	holding cost
bc_p	backorder cost
sc_p	start-up cost
pc_p	production cost

Variables

Finishing Lines

$F^f_{p,t,s}, F^h_{p,t,s}, F^n_{p,t,s}$	Production at full, half or no capacity$\{0,1\}$
$Fh^f_{p,t,s}, Fh^h_{p,t,s}, Fh^n_{p,t,s}$	Hours of production at full, half, or no capacity

Cold Mill

$C^f_{t,s}, C^h_{t,s}, C^n_{t,s}$	Production at full, half or no capacity$\{0,1\}$
$Ch^f_{t,s}, Ch^h_{t,s}, Ch^n_{t,s}$	Hours of production at full, half, or no capacity

Hot Mill

$H^f_{t,s}, H^h_{t,s}, H^n_{t,s}$	Production at full, half or no capacity$\{0,1\}$
$Hh^f_{t,s}, Hh^h_{t,s}, Hh^n_{t,s}$	Hours of production at full, half, or no capacity

3.5 Production Planning Model Comparison Case Studies

In this section, two case studies are presented to quantify the improvement each modelling technique developed in the preceding sections has when applied to an industrially relevant problem. The first case study will demonstrate the improvement made when including uncertainty in a two-stage stochastic model compares to its deterministic counterpart. The performance of the models will be quantified using Equation 3.61. This equation is the distance from the inventory targets summed over all products for all weeks.

$$\sqrt{\sum_{p}\sum_{t}(WIP_{p,t,s} - WIP^{tar}_{p,t})^2 + (HB_{p,t,s} - HB^{tar}_{p,t})^2} \quad (3.61)$$

The second case study compares the SAPP and the CBSAPP by quantifying the production costs associated with the two models. A third model that is a compromise between the two extremes is introduced as well. The objective of this case study is to compare each models ability to reduce operating costs and maintain operability in the plant. The models in this section will be quantified by the total cost to operate the plant over a ten week horizon when considering holding costs, backorder costs and production costs which is the objective function of the CBSAPP model in the previous section.

Both of the case studies use actual production data from a steel mill located in Hamilton, Ontario, extracted on February 13, 2013. All models used in this section are modelled with AMPL version 2006.06.26 and CPLEX 12.5.0.0 using up to 8 threads. The simulations were performed on an Dell Vostro 430 computer (Intel Core i7-870, 8GB DDR3 RAM, Nvidia GeForce GT240, Windows 7 x64).

3.5.1 Case Study 1 - Benefits of Stochastic Modelling

The purpose of this case study is to evaluate the ability of a deterministic model and a stochastic model to reach the objective provided in Equation 3.61 in each of the 100 scenarios. This case study begins with a generation of 100 demand scenarios which are populated randomly from the normal distributions created from historical data developed in Section 3.3.3. It is important to note that the models are not privy to these demand scenarios. In particular, the 100 scenarios used in the stochastic model are different from the 100 scenarios used to test the models ability to meet the objective.

The plant data that is available upon data extraction includes the Long Term Planning inventory target trajectories, the upcoming 10 weeks of demand forecasts, an estimate of the maximum amount of production capacity of each mill and line, an estimate of the current levels of inventory, limitations on the amount of inventory the warehouses can hold, projected maintenance downtime in the upcoming 10 weeks, and the number of scheduled employee hours.

To begin, the deterministic model is evaluated. The model is given only the average value of demand from the order books. This is equivalent to the mean of the normal distributions provided in Section 3.3.3. The model is required to make production decisions for the non-anticipative period (week 1 at the finishing lines and weeks 1-3 at the hot mill) based on this most likely value of demand. The true value of demand (based on the randomly generated scenario) is then provided to the model, and it is allowed to make its production selection for the remaining weeks. When using this optimization approach, the DAPP is able to achieve a total distance from inventory target per week of 2,297. Additionally, this method does not become infeasible under any scenario.

Finally, the stochastic model is evaluated. This model is provided with 100 randomly generated scenarios of demand and its production values for the non-anticipative weeks are locked in. Subsequently, the model is provided with one of the randomly generated scenarios, and the remaining weeks are optimized until all scenarios are considered. When considering the SAPP, it is found to have a total average weekly distance of 2,299 which represents a negligible difference over its deterministic counterpart. This is due to the fact that the large majority of the uncertainty occurs outside of the non-anticipative 1-3 week period. This can be very clearly seen in Figure 3.4. The width of the distribution is very small when compared to the absolute value of the demand. This means a sample from anywhere inside the 3 standard deviation window is essentially equivalent to taking the average. As such, the stochastic model and the deterministic model are provided with nearly identical demand data sets. It is expected that there would be little improvement in the stochastic model, as there is essentially no uncertainty. In fact, when considering the data set provided, 98.7% of the uncertainty (open quota) occurs in weeks 4 through 10. Another reason for the lack of improvement is that the model does not run near its constraints. One of the main benefits of a stochastic model is the ability to predict infeasibilities under certain scenarios and hedge against them in the first stage. If there is no risk of infeasibility, then there is no hedging, and no benefit to a stochastic model. The following section will develop a model that pushes the boundaries of feasibility in an effort to reduce costs. It is expected that a stochastic model will be able to provide value in this situation.

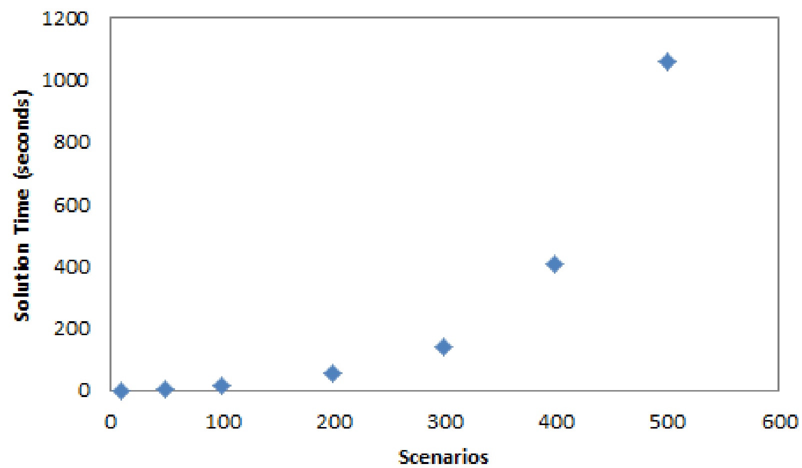

Figure 3.8: Stochastic solution time

An additional expected result of the stochastic case study is the increase in solution time as a function of the number of scenarios. This can be seen in Figure 3.8. This figure shows an exponential relationship. This is expected as the number of variables doubles with each additional scenario.

3.5.2 Case Study 2 - Cost Reductions and Operability

The purpose of this case study is to quantify the benefit of the CBSAPP over the inventory targeted SAPP. This is accomplished by first calculating the total production costs under the SAPP model. The holding cost and back order cost per 100 tons of production under the SAPP model was found to be \$6,833 which comprises 2.2% of the total cost of production. With this base line set, the total production costs are calculated under the CBSAPP model. This was found to be \$526.16 per 100 tons. The breakdown of the individual relative costs for these models are presented in Figure 3.9. The main takeaway from this Figure is the reduction from the Work-In-Progress and Hot Band holding costs from a marginal value, to a negligible

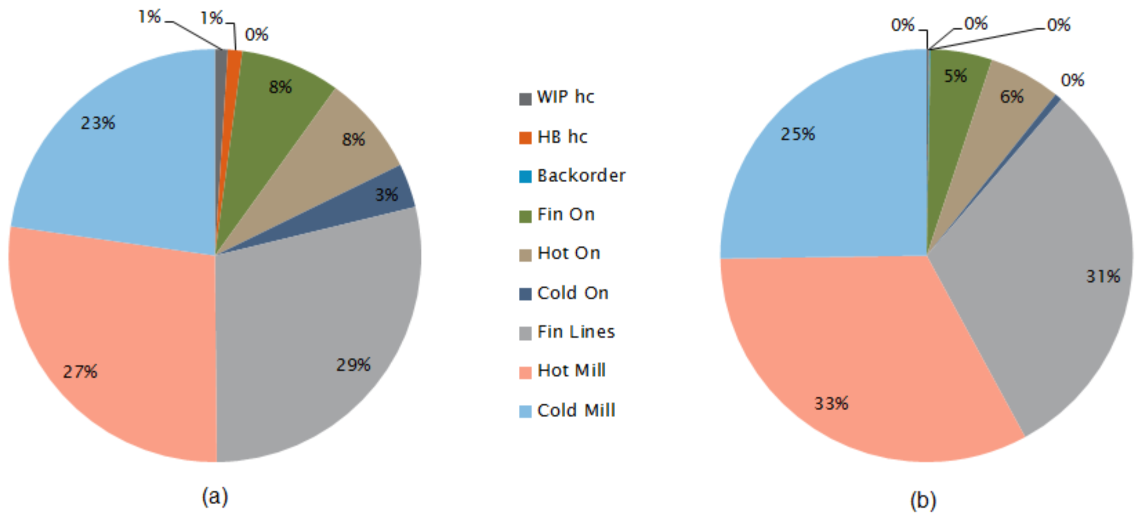

Figure 3.9: Cost break down and comparison for the inventory target SAPP (a) and CBSAPP (b)

one. In total, this represents a 2.1% reduction in operating costs under the CBSAPP. Another result of this change in modelling techniques is the introduction of oscillatory behaviour into the production profile when using the CBSAPP model. Figure 3.10 shows the oscillatory behaviour at the Hot Mill. This behaviour is a result of the different linear cost coefficients for each mode of production. Essentially, it is cost optimal to turn a production line on to full capacity to stock up inventory, and then shut the line down completely while meeting orders with the inventory stock. This mode of operation may be cost optimal, but is undesirable from a plant operability stand point. The plant managers must constantly reallocate staff to meet fluctuating production levels. A solution to this problem is presented by reintroducing the concept of inventory targets, but allowing it to be a variable in the CBSAPP model rather than a parameter. This is shown in Equation 3.62 and is referred to here as

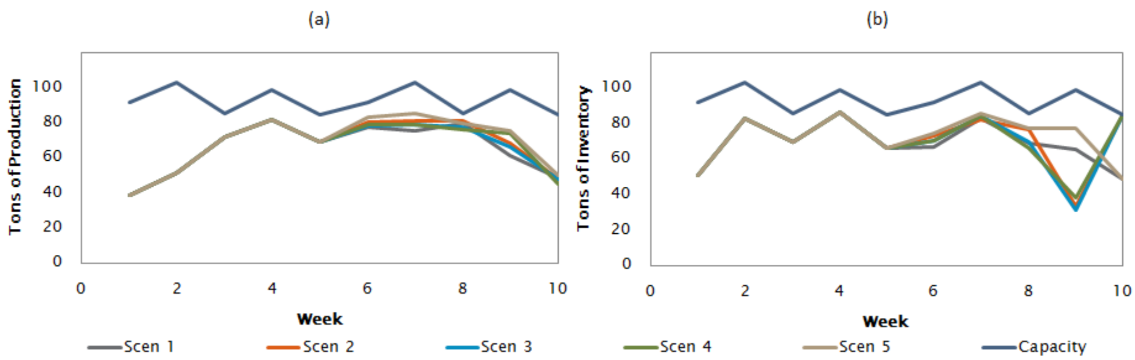

Figure 3.10: Baseline(a) and oscillatory behaviour introduced at the Hot Mill due to the CBSAPP(b)

the Cost Optimal Inventory Target (COIT) model.

$$\begin{aligned}\text{Min} \sum_p \sum_t \sum_s & hc_p(WIP_{p,t,s} + HB_{p,t,s}) + bc_p(B_{p,t,s} + B^h_{p,t,s}) \\ & + sc_p(F^f_{p,t,s} + 0.6sc_pF^h_{p,t,s} + 0.1sc_pF^n_{p,t,s}) \\ & + sc_p(H^f_{p,t,s} + 0.6sc_pH^h_{p,t,s} + 0.1sc_pH^n_{p,t,s}) \\ & + (WIP_{p,t,s} - WIP^{tar}_p)^2 + (HB_{p,t,s} - HB^{tar}_p)^2\end{aligned} \tag{3.62}$$

Note that HB^{tar}_p and WIP^{tar}_p are no longer functions of the time period. This requires the model to select one optimal level of inventory to attempt to meet while simultaneously minimizing holding costs. The square nature of this variable in the objective function means meeting this target is of high importance when the distance from the target is large, but becomes insignificant when the actual inventory level is close to the target. At this point, minimizing costs become the priority. Figure 3.11a shows the oscillatory nature in the CBSAPP and the reduction after reintroducing the inventory targets in Figure 3.11b. There is a small cost to return operability to the plant. This comes in the form of additional holding and back order costs to a total of \$1,048 per 100 tons. The inventory levels shown for the three models discussed in this

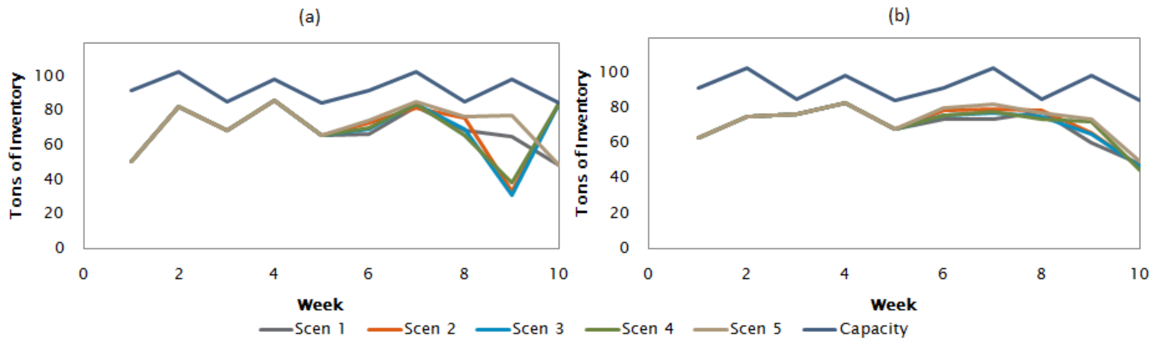

Figure 3.11: Oscillation reduction at the Hot Mill shown in (b) when compared to the CBSAPP(a)

case study for a sample product are presented in Figure 3.12. Graph 3.12(a) shows the initial high levels of inventory introduced by the Long Term Planning target. The inventory storages are sufficiently high enough to meet all uncertainty introduced by demand and production with additional unnecessary inventory. The opposite of this is shown in 3.12(b). The fictitious inventory target is at zero, and the amount of inventory held is strictly sufficient enough to meet the maximum uncertainty provided in the worst scenario generated. A compromise between these two extremes is seen in 3.12(c) when the inventory targeted SAPP and CBSAPP modeling paradigms are merged. The optimization model chooses an inventory level that is large enough to meet all uncertainty while returning operability to the plant. A summary of the holding and back order costs per 100 tons, the percentage reduction in operating cost, and pros/cons of all three models discussed in this case study are presented in Table 3.3. A comparison of the problem size is shown in Table 3.4. Finally, a table of the solution time and optimality gap is presented in Table 3.5.

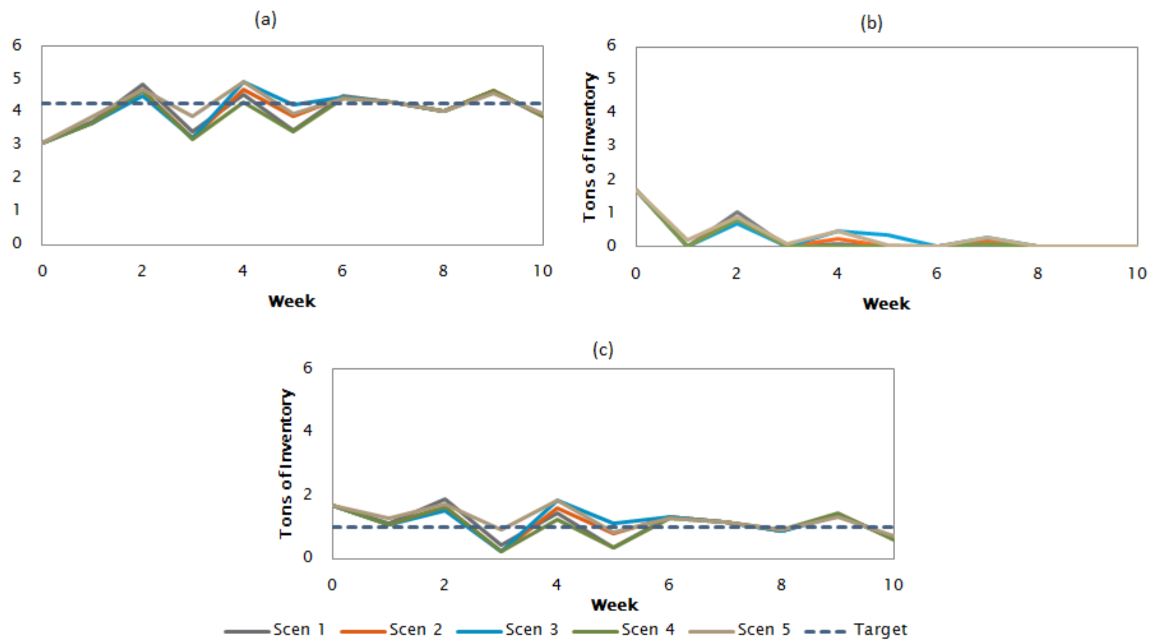

Figure 3.12: Comparison of the selected inventory targets under the SAPP(a), CB-SAPP(b) and COIT(c)

	SAPP	CBSAPP	COIT
Holding Cost and Backorder Cost per 100 tons	$6,833.36	$526.15	$1048.44
Percent Reduction in Operating Cost	-	2.1%	1.9%
Pros	Small chance of missing shipping dates	Significant cost savings	Significant cost savings, Non-oscillatory production
Cons	Significant holding costs	Oscillatory production behavior	Loss of 0.2% in savings

Table 3.3: Summary of stochastic production planning models

	SAPP	CBSAPP	COIT
Type	QP	MILP	MIQP
Percent Reduction in Operating Cost	-	2.1%	1.9%
Scenarios	100	20	10
Binary Variables	0	10200	5100
Continuous Variables	138000	27600	16600
Constraints	125860	25060	15232

Table 3.4: Comparison of stochastic production planning models problem size

	SAPP	CBSAPP	COIT
Percent Optimality	0.0001%	2%	2%
Solution Time	21.72 seconds	10 minutes (cap)	30 minutes (cap)

Table 3.5: Comparison of stochastic production planning models solution time

Chapter 4

Scheduling Using Mathematical and Constraint Programming

The previous chapter addressed the medium term planning problem. This chapter addresses the scheduling problem that resides one temporal level below Medium Term Planning. A visual representation this problems location in the planning hierarchy can be seen in Figure 3.2. The objective of this chapter is to develop a mathematical optimization model for the scheduling of groups of orders at a typical integrated steel mill. Scheduling at a steel mill involves allocating orders to one of many parallel production lines, and sequencing the orders so they are optimal with respect to order due dates and cost. Allocation and sequencing problems are combinatorial in nature and are difficult to solve computationally. As a result, many literature sources have investigated methodologies for reducing the solution time required for these problems. An important factor in computation time is the choice of the mathematical model. The mathematical model dictates the number of variables, the number of constraints, and the solution algorithm used to solve the problem. The selection of an appropriate mathematical model is not always clear. As a result, the first section of this chapter develops several models, including one MINLP, three distinct MILP models and a constraint programming model that uses a significantly different solution algorithm. The purpose of creating these models is for a comparison of their ability and computation time to reach an optimal solution. The results of this comparison are presented in the first case study. Once the most efficient model is selected, it is extended to include constraints that are unique to the integrated steel mill scheduling problem. The second case study investigates this model's ability to solve a problem of industrial scale. Finally, the chapter concludes with a verification of optimality.

4.1 Application Context: Campaign Scheduling

Although the steel mill formally calls this campaign *planning*, it is truly a *scheduling* problem. A scheduling problem involves selecting the precedence between activities given a finite amount of resources with an objective that usually involves minimizing duration, lateness or cost. The "activity" in a steel mill can be a customer order, a steel coil, or in this situation, a campaign. A campaign is defined as a block of time that the production line uses to process steel with specific coil characteristics. A visual reference for the following description can be found in Figure 4.1. Campaigns typically consist of a series of steel coils welded end-to-end. The coils have been allocated customer orders that are grouped together to be sent through the production line in one continuous batch. Campaigns are further classified into campaign types. Each campaign type typically has its own unique combination of rollers, annealing specifications, galvanizing liquid, thickness/width range, etc. Switching between campaign

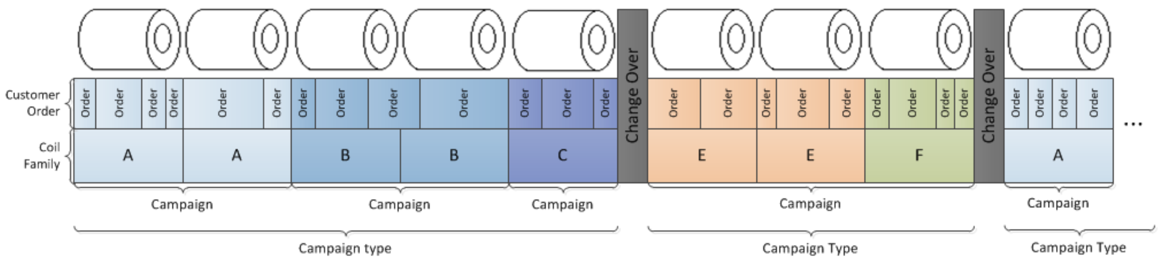

Figure 4.1: Classification of customer orders into campaigns and campaign types

types involves shutting down the line for a set amount of time in order to change rollers and liquid. A product line can have between 6 and 15 campaign types. The campaign types are sequenced and repeated, forming a campaign cycle. If a customer order is not allocated within a campaign it might be weeks before the cycle is repeated and the order can be allocated again.

The objectives of the campaign scheduling problem are three-fold. The main objective is to maximize on-time delivery, or minimize lateness. The secondary objective is to minimize operational costs by maximizing throughput. The tradeoffs are clear when considering all factors in the above paragraph. In order to maximize throughput, all campaigns of one type would be grouped together in continuous operation. This would minimize the amount of time that the line would have to be shutdown for changeover activities. However, this would typically result in a large number of late orders since one campaign cycle can last for several weeks. Conversely, ordering the campaigns based strictly on due date would result in a large number of changeovers. There exists some optimum in the middle.

Concerning the classifications of scheduling problems introduced by Mendez et al. (2006) and replicated in the literature review, the campaign scheduling presented here has a sequential process topology with single stage and sometimes parallel units. This means each batch must be processed following a sequence of stages defined by a recipe. The inventory storage policies are finite and consist of dedicated and shared storage units. Inventory storage is ignored in this case. The material transfer is considered instantaneous, that is, the transport of material between processing units is neglected. The batch size of each campaign is fixed and the processing time is fixed, but unit dependent. The demand patterns are based on due dates and each order has its own date. The changeovers are product and sequence dependent. There exists a continuous constraint in the form of roller usage. Maintenance time constraints are ignored in this problem. There are costs associated with changeovers and inventory holding. There is uncertainty in demand and unscheduled shutdowns, however for simplicity this problem is considered deterministic. The problem under consideration is a simplified version of an actual problem solved at a steel mill located in Hamilton, Ontario. The ultimate goal of the work described in this chapter to replace the manually intensive heuristic with a mathematical optimization model that is capable of producing optimal decisions in a shorter amount of time. In order to simplify the problem to a tractable size, only one product line is considered. This negates the need to consider line switching and significantly reduces the number of variables and constraints in the problem. The product line under consideration involves galvanized steel production.

Galvanized steel is steel that has been immersed in molten zinc at approximately 850 degrees Fahrenheit to provide protection against corrosive environments. Galvannealed steel goes through a similar process to Galvanized steel, with the addition of a heat treatment. This additional heat brings iron to the surface making it harder. The additional iron on the surface prevents scratching, increases the bonding with paint products, and makes the steel better for welding applications. In order to switch between production of galvanized and galvannealed steel, as period of shutdown time is required at the production line.

The data set used is 1227 actual customer orders divided amongst 33 campaigns which represents approximately 6 weeks of production. The information provided includes the size in tons of each order, the due date of each order, the prepossessing time required per order, and the type of campaign; either galvanneal (type 1) or galvanize (type 2). As discussed in the literature review, an optimization model that chooses the precedence of all 1227 orders is likely to be computationally intractable. Instead, this model will focus on the precedence of the 33 campaigns. An advanced production planning program is used to allocate customer orders into campaigns. An algorithm is run over-night that considers all customer due-dates in the system as well the available capacity of each of the product lines. The system takes into consideration the allowable sequence of orders and fits each customer order into a campaign slot. The algorithm is run for customer orders in the time horizon from 6 weeks to 6 months. Once an order is within a 5 week time horizon, the algorithm is removed and the campaign planning is manually sequenced by an experienced operator. This is done to prevent the planning program from constantly adjusting the campaign plan as manufacturing and shipping need a fixed plan to properly allocate resources. This manual allocation of orders into a campaign is performed weekly with daily adjustments to account for any unplanned manufacturing disturbances or new orders that can satisfy remaining open quota. As such, the precedence of the customer orders is considered fixed and is an input to the model. The due date of the campaign is determined by an average of the due dates of the orders inside the campaign. The due dates and processing times are rounded to the nearest hour. It is assumed that there are no storage constraints at the end of the production line.

This problem falls under the category of ”single machine scheduling” which has been researched extensively by the operations research community, and to a lesser degree, the chemical engineering community.

4.2 Comparison and Selection of Modelling Type

The purpose of this section is to introduce and compare various mathematical optimization and constraint programming models in an effort to determine which model best suits the steel mill campaign planning problem discussed above. The models will be compared on their ability to reach optimality in the shortest amount of time. Each subsection will begin with a classification of the modeling type introduced in Figure 2.6. An explanation of the model's objective and constraints follows.

4.2.1 Continuous Time Models

Analyzing the data provided, the summation of all 33 campaigns processing times equates to 41.89 days worth of orders in the system. The campaign duration and due date parameters are accurate to within the hour. As such, in order to properly represent the system within a discrete time frame, the model would need 1,006 individual discrete time intervals. The discrete time models key variable would have one subscript for the campaign and one for the time interval. This would result in $33 * 1006 = 33,198$ binary variables. Since the solution time of a discrete model scales with the degree of discretization, it is presumed that this would make the problem too large to be solved within a reasonable amount of time. Additionally, if any additional production lines were to be added to the model at a later date, this problem would magnify. For this reason, a discrete time model is not investigated.

A continuous time representation is therefore used for all the following models. A discrete time model employs a single set of binary variables that handles the sequencing and timing decisions. Conversely, a continuous model uses a completely separate set of continuous variables to handle the timing decisions. In general, this provides a significant reduction in the number of variables and provides more accurate solutions (Mendez et al., 2006). As a consequence, continuous time models also require the addition of big-M constraints which tend to increase the complexity in the form of a large integrality gap. The formal name given to the four continuous time models in this section are: Slot-Based, Linearized Slot-Based, Immediate Precedence, and General Precedence. These names are in reference to the type of key binary variable used.

Slot-Based Model

This model assumes there exists a time "slot" for every member in the set of predefined time intervals with an unknown duration, which becomes the key variable of the model. Each unit or processing line has its own set of time slots. However, for the sequential case with one product line, this becomes relatively simple as the number of time slots is fixed and equal to the number of campaigns. The key discrete variable is $X_{i,k}$ which is equal to 1 if campaign i exists in time slot k. The basis for this model can be found in Pinto and Grossmann (1998). Note that the nomenclature used in this model can be found at the end of the subsection.

The objective function is presented in Equation 4.1. This simply shows the main objective as a minimization of lateness and earliness using variable L_i and E_i respectively. It is important to recognize that this is different from a minimization of the number of late jobs, which will be used in later models. This objective uses continuous variables which negates the need for additional binary variables.

$$\text{Min} \sum_i [L_i + E_i] \tag{4.1}$$

Equation 4.2 states that exactly one campaign must be assigned to every time slot.

$$\text{s.t.} \sum_i X_{ik} = 1 \qquad \forall k \in K \tag{4.2}$$

Similarly, Equation 4.3 enforces that every campaign is assigned to one time slot.

$$\sum_k X_{ik} = 1 \qquad \forall i \in I \tag{4.3}$$

This model has separate equations for the timing constraints, which is a feature of continuous time formulations. Extra continuous variables are required to model the timing associated with each campaign. These are Ts_k and Te_k which represent the start and end time of each campaign respectively. Equation 4.4 ensures that there is no overlap between time slots. It also includes a component for the sequence dependent changeovers. The size (in hours) of the changeover is equal to $\tau_{ii'}$. This multiplication of binary variables introduces a bilinear term meaning this model is nonlinear.

$$Te_k = Ts_k + \sum_{i \in I} X_{ik} T_i + \sum_i \sum_{i'} X_{ik-1} X_{i'k} \tau_{ii'} \qquad \forall k \in K \tag{4.4}$$

Equation 4.5 ensures that there is no dead time between campaigns which is required as the steel coils are physically welded together for continuous operation of the production line.

$$Te_k = Ts_{k+1} \qquad \forall k \in K \tag{4.5}$$

The continuous lateness variable is defined in Equation 4.6. The necessary multiplication of the binary variable with its timing variable also adds nonlinearity to this model.

$$L_i - E_i = \sum_k X_{ik}(Te_k - D_i) \qquad \forall i \in I \tag{4.6}$$

Equation 4.7 ensures that the continuous variables are greater than zero.

$$L_i, E_i, Ts_k, Te_k \geq 0 \qquad \forall i \in I, k \in K \tag{4.7}$$

Finally, Equation 4.8 simply states that the timing should start at zero.

$$Ts_1 = 0 \tag{4.8}$$

This model is relatively simple to comprehend but suffers from nonlinearity due to the selection of the objective function and the fact that sequence dependent changeovers are necessary for this problem. Fortunately, there exist linearization techniques which will be addressed in the following subsection.

Nomenclature

Sets

I	All Campaigns
K	Time Slots

Parameters

T_i	Processing Time
$\tau_{i,i'}$	Changeover Time
D_i	Campaign Due Date

Variables

$X_{i,k}$	Binary: Assignment
Ts_k	Campaign start time
Te_k	Campaign end time
L_i	Lateness
E_i	Earliness

Linearized Slot-Based Model

The techniques used to linearize the slot-based model were originally proposed by Sahinidis and Grossmann (1990). For the sequence dependent changeovers, an additional binary variable can be defined. Let $Z_{ii'k}$ be a binary variable that is equal to 1 when the processing of campaign i occurs immediately before the processing of campaign i' at slot k. Equation 4.9 is added to the model to ensure that $Z_{ii'k}$ is activated when X_{ik} and $X_{i',k+1}$ are equal to one.

$$Z_{ii'k} \geq X_{ik} + X_{i',k+1} - 1 \qquad \forall i \in I, i' \in I, k \in K \tag{4.9}$$

This new variable can now be substituted anywhere the bilinear multiplication of $X_{ik-1}X_{i'k}$ is found. Particularly, constraint 4.4 is modified to create the new Equation 4.10.

$$Te_k = Ts_k + \sum_{i \in I} \sum_{i' \in I} Z_{ii'k}(T_i + \tau_{ii'}) \qquad \forall k \in K \tag{4.10}$$

To deal with the bilinear term involving a multiplication of the X_{ik} and Te_k variables in Equation 4.6, the method shown in Xpress (2009) can be followed. A new continuous variable y_{ik} is created of the form $y_{ik} = X_{ik}Te_k$. This new continuous variable is then substituted into Equation 4.6 to create Equation 4.11.

$$L_i - E_i = \sum_k y_{ik} - D_i \qquad \forall i \; inI \tag{4.11}$$

In order to make the new y_{ik} behave like the bilinear term, an additional two constraints are necessary and shown in Equation 4.12 and Equation 4.13 where M is a large number.

$$y_{ik} \leq MX_{ik} \qquad \forall i \in I, k \in K \tag{4.12}$$

$$Te_k - M(1 - X_{ik}) \leq y_{ik} \leq Te_k \qquad \forall i \in I, k \in K \tag{4.13}$$

The first of these constraints ensures that the new variable is equal to zero when the binary term is equal to zero, and is unbounded otherwise. The size of the big-M does not need to be any larger than the sum of all the processing times plus changeovers. To make this as tight as possible without being restrictive, it is set to the sum of all the processing times of the campaigns plus 33 changeover periods which would be the maximum number possible. The second constraint ensures that y_{ik} is equal to Te_k when the binary is equal to one. As can be seen, all of these constraints are linear. As such, the nonlinearities have been removed from this formulation at the expense of an increase in the number of variables and constraints. Since the number of campaigns is equal to the number of time slots in the single-unit case, this problem has increased by n^2 continuous variables and n^3 binary variables where n is the number of campaigns. This is a significant increase. However, the benefit to a linear program when compared to its nonlinear counterpart is that there exists a guarantee of optimality. Additionally, superior solution algorithms are available for linear programs. It is unclear whether the increase in the number of variables will offset these positives. The first case study will investigate this concept.

Immediate Precedence Model

The mathematical model used in this subsection is largely based on the one available in Radhakrishnan and Ventura (2000). The key decision variable is Z_{ijm} which is equal to one if job i is immediately followed by job j on machine m. This is fundamentally similar to the variable introduced in the linearized slot-based model. However, the immediate precedence model is completely based on this variable. Additionally, the slot-based models' third subscript is k, representing the number of time slots of which there is 33. Conversely, the precedence models' third subscript is m, representing the machine or production line. Since the reduced steel mill problem only considers one product line, the m subscript takes on only one value. This results in a significantly smaller amount of variables when compared to the slot-based models. The precedence model will be extended to multiple production lines in the future, so it is helpful to include the m subscript in this formulation.

The objective function in 4.14 is unchanged from the first model.

$$\text{Min} \sum_i [L_i + E_i] \tag{4.14}$$

The slot-based models used the X_{ik} variable to assign a campaign to a slot. The precedence models use instead use the variable y_{im} to assign a campaign to a production line. Constraint sets in Equation 4.15 and 4.16 combine to affirm that all campaigns are assigned to a production line. They also ensure that if campaign i precedes campaign j, then they both must be assigned to the same machine. They also ensure that a campaign can only be followed by one campaign with the exception of the last campaign which is followed by nothing.

$$\text{s.t.} \sum_{i \in I, i \neq j} Z_{ijm} = y_{jm} \qquad \forall j \in I, m \in M \tag{4.15}$$

$$\sum_{j \in I, i \neq j} Z_{ijm} \leq y_{im} \qquad \forall i \in I, m \in M \tag{4.16}$$

Constraint 4.17 ensures that every job is assigned to exactly one machine.

$$\sum_m y_{im} = 1 \qquad \forall i \in I \tag{4.17}$$

The timing constraints begin with Equation 4.18. This equation ensures that the completion time of campaign j will occur after campaign i plus any setup time and processing time.

$$Te_j + W(1 - Z_{ijm}) \geq Te_i + \tau_{ij} + T_j \qquad \forall i \in I, j \in I, m \in M : j \neq i \tag{4.18}$$

The definition for lateness is in Equation 4.19.

$$L_i - E_i = Te_i - D_i \qquad \forall i \in I \tag{4.19}$$

By the definition of Z_{ijm}, all campaigns must follow a previously sequenced campaign. Therefore, in order to facilitate the start of the sequence, a dummy campaign is introduced called campaign '0'. Equation 4.20 initializes this first fictitious campaign.

$$\sum_i Z_{0im} = 1 \qquad \forall m \in M \tag{4.20}$$

Equation 4.21 ensures that the continuous variables are greater than zero.

$$L_i, E_i, Te_i \geq 0 \qquad \forall i \in I \tag{4.21}$$

Nomenclature

Sets

I	All Campaigns
M	Machines (Product lines)

Parameters

T_i	Processing Time
$\tau_{ii'}$	Changeover Time
D_i	Campaign Due Date
W	Large Number

Variables

$Z_{ii'm}$	Binary seq: $= 1$ if campaign i' scheduled after campaign i
y_{im}	Binary assigning campaign i to product line m
Te_i	Processing end time
L_i	Lateness
E_i	Earliness

General Precedence Model

The mathematical model used in this subsection is based on the model given in Zhu and Heady (2000). This model separates the sequencing of the campaigns from the assignment of campaigns to machines. The key decision variable is Z_{ij} which is equal to one if campaign i is followed by campaign j. This definition is functionally different from the immediate precedence model in that campaign i does not need to *immediately* precede j. The same variable y_{im} is used for assigning campaigns to production lines.

The objective function in 4.22 is unchanged from the previous models.

$$\text{Min} \sum_i [L_i + E_i] \tag{4.22}$$

The sequencing constraints in 4.15 and 4.16, as well as the timing constraint in 4.18 are replaced with two complex constraints shown in 4.23 and 4.24.

$$Te_j \geq Te_i + T_j + \tau_{ij} + B(Z_{ij} + y_{im} + y_{jm} - 3) \qquad \forall i \in I, j \in i+1..N, m \in M, j \neq i \tag{4.23}$$

$$Te_j \geq Te_i + T_j + \tau_{ji} - B(Z_{ji} - y_{im} - y_{jm} + 2) \qquad \forall i \in 0..N, j \in i+1..N, m \in M, j \neq i \tag{4.24}$$

The first of which ensures that the end of campaign j is greater than the end of campaign i plus the processing and changeover time if and only if campaign i precedes j and both are assigned to the same production line. The latter utilizes the concept that if Z_{ij} is equal to 0, than Z_{ji} must be equal to 1. It performs the same function as 4.23, but for when Z_{ji} is active. Constraint 4.25 through 4.27 remain unchanged from the previous model.

$$\sum_{m} y_{im} = 1 \qquad \forall i \in I \tag{4.25}$$

$$L_i - E_i = Te_i - D_i \qquad \forall i \in I \tag{4.26}$$

$$Te_i, L_i, E_i \geq 0 \qquad \forall i \in I \tag{4.27}$$

Using this slight change in notation, Zhu and Heady (2000) have seen on average a reduction by a factor of five for many problems. The authors also provided additional constraints to help tighten the search space based on the triangle inequality which did not prove to make a significant difference in this application. The reduction in the number of binary variables from the immediate precedence model is given by 4.28, where N is the number of campaigns and M is the number of machines.

$$Reduction = N(N+1)M - \frac{N(N-1)}{2} \tag{4.28}$$

This concept is further discussed in the first case study.

Nomenclature

Sets

I	All Campaigns
I_{GI}	Galvanize Campaigns
I_{GA}	Galvaneal Campaigns
m	Machines (Product lines)

Parameters

T_i	Processing Time
$\tau_{ii'}$	Changeover Time
D_i	Campaign Due Date
W	Large Number

Variables

$Z_{ii'm}$	Binary seq: $= 1$ if campaign i' scheduled after campaign i
y_{im}	Binary assigning campaign i to product line m
Te_k	Processing end time
L_i	Lateness

4.2.2 Constraint Programming Model

The constraint programming model uses IBM's optimization programming language (OPL) to create a mathematical model that is solvable with their constraint programming solver. The constraint programming solver is unique in that it divides the search space into interval variables and only performs calculations for solutions inside the feasible domain. The OPL itself is unique and behaves partly like a traditional optimization language and partly like a programming language. Logical constraints can be used to restrict the feasible domain in addition to mathematical constraints.

The variables that are required to replicate a mathematical optimization model include a set of *interval* variables and one *sequencing* variable. An interval variable behaves much like a slot. It has a unique start time, end time, size and type. Its position in time is the variable. The "type" characteristic allows the distinguishing between campaigns without the need to introduce a binary changeover variable. The sequence variable represents the position of campaigns in relation to each other. The values that the sequence take on are the order of the intervals within the sequence. For more information regarding the types of variables that IBM uses, the reader is directed towards the IBM OPL Reference Manual (2009). The interval variable C_i representing campaign i with size T_i representing the tonnage of steel in the campaign is shown in Equation 4.29.

$$C_i \text{ size } T_i \tag{4.29}$$

The sequencing variable is entered into OPL as shown in Equation 4.30. This says that sequence S represents the relative position in time of campaigns C_i. Each of the campaigns are given a type J_i. If the types of back-to-back campaigns are different, the sequence adds in changeover time between the campaigns.

$$S \text{ in } C_i \text{ types } J_i \tag{4.30}$$

The objective function used in this model is the same as in all the previous models.

$$\text{Min} \sum_i [L_i + E_i] \tag{4.31}$$

The constraints required in the OPL are compact due to the built in function *noOverlap* shown in Equation 4.32. This equation states that the intervals inside of the sequence S may not overlap. Additionally, since we provided an interval type to the sequence, the built in function will add the changeover time $\tau_{ii'}$ if two intervals sequenced together are of different type.

$$\text{s.t.} \quad \text{noOverlap } (S, \tau_{ii'}) \qquad \forall i \in I, i' \in I \tag{4.32}$$

The OPL uses built in functions to access values "inside" an interval. In Equation 4.33, we require the end of the interval to determine its lateness or earliness. This value is simply accessed by the *endOf* function.

$$\text{endOf } (C_i) - D_i = L_i - E_i \tag{4.33}$$

The benefits of this modelling paradigm are obvious. It is extremely compact due to the use of the built in functions and its unique definition of variables. It is unclear whether the constraint programming solver will outperform CPLEX. This will be investigated in the first case study.

Nomenclature

Sets

I	All Campaigns
J	Campaign type

Parameters

T_i	Processing Time
$\tau_{ii'}$	Changeover Time
D_i	Campaign Due Date
N	Number of campaigns

Variables

L_i	Lateness
E_i	Earlienss
I	All Campaigns
C_i size T_i	Interval variable: One interval per Campaign
S in C_i types J_i	Sequence variable: Represents the sequence of campaigns C

4.2.3 Case Study 3 - Model Solution Time

The purpose of this case study is to investigate the preceding models in their ability to solve an industrially relevant problem. The mathematical models are compared by incrementally increasing the problem size until the full problem size is reached. A time limit is set at 5 minutes of computation time. The MINLP model is solved using the Knitro solver, version 9.0.1 with multi-start enabled. The MILP slot-based, immediate precedence, and general precedence models are solved using CPLEX 12.5.0.0. All of these model are coded in Ampl version 2006.06.26. The constraint programming model uses the OPL and CP solver. All models are solved to an optimality gap of 0.01%.

The data set used for this case study is provided by an integrated steel maker in Hamilton, Ontario. The data provided includes 33 campaigns, 1227 customer orders, steel tonnage per order, processing time and product type. The data represents 6 weeks of production.

Before beginning the experiment, the size of the problems can be viewed and a hypothesis made. The problem size in terms of number of variables and constraints for the full 33 campaigns is given in Table 4.1. Note that variables and constraints are not considered in the same way in Constraint Programming. From this table, we can see that the MILP slot-based model is large in both variables and constraints. The large number of binary variables is due to the presence of the $Z_{ii'k}$ to handle changeovers. The increase in constraints over the MINLP slot-based model is due to the bilinear relaxation. This model is not expected to perform well for this reason. The Immediate Precedence model has the same number of variables as the MINLP slot model, but has additional constraints. It is unclear whether the additional constraints will

Model	Binary Variables	Continuous Variables	Constraints
MINLP Slot	1089	99	164
MILP Slot	37026	1188	38312
Immediate Precedence	1089 (1089)	99	1189
General Precedence	1056 (528)	164	2145
Constraint Programming*	100	-	34
* Variables not comparable			

Table 4.1: Problem sizes and specifications for different models

slow down or speed up solution time. CPLEX has the feature of presolving the set of equations and reducing the size of the MILP. The reduced size of the MILP is displayed in parentheses in Table 4.1. As can be seen, the General Precedence model can be reduced to 528 binaries which is consistent with substituting $N = 33$ and $M = 1$ into Equation 4.28 as expected. The Constraint Programming software provided by IBM does not calculate variables and constraints in a similar manner to mathematical optimization. It is not practical to compare them. However, it is presented in the table for completeness.

The results for the solution times as a function of problem size are presented in Figure 4.2. The Knitro solver used for the MINLP slot model uses a mutli-start feature which occasionally stops at a local and infeasible solution. The MILP slot model and constraint programming models reach the maximum solution time at 12 campaigns. The immediate precedence model performs significantly better, but is not able to solve the entire problem within the allotted time. The constraint programming model does not perform well in this case study. It is assumed that the lack of constraints in this model create a search space that is too large to be efficient under the constraint programming paradigm that benefits from tight problems. The most efficient model used is the general precedence model. It is able to solve the full problem size in 127 seconds.

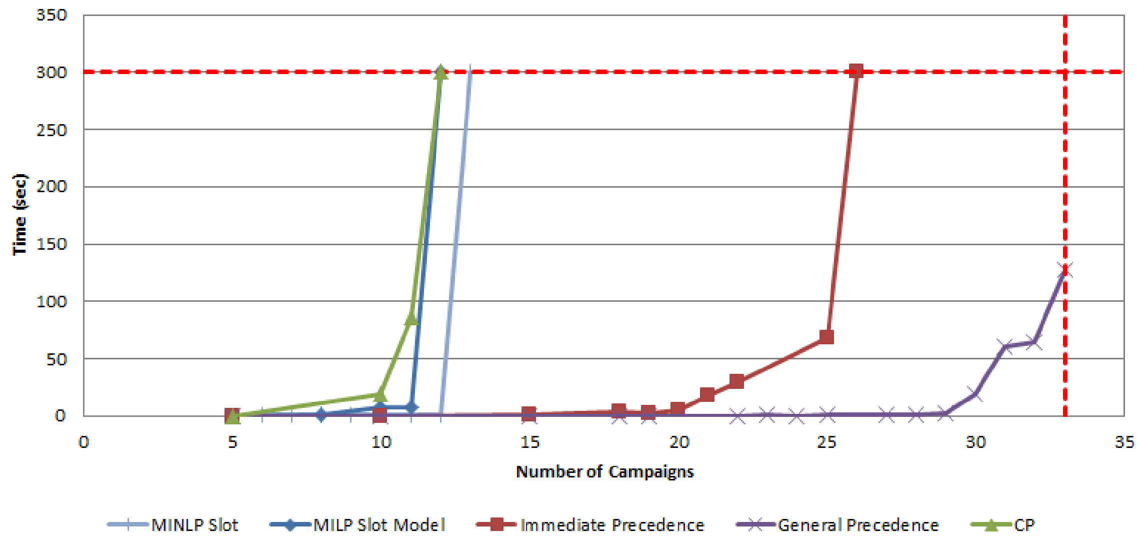

Figure 4.2: Solution time as a function of problem size for different model types

Model	Best Feasible Solution	Time to Best Feasible Solution	Optimality Guaranteed?
MINLP Slot	30289	300	N
MILP Slot	N/A	No Sol'n Found	N
Immediate Precedence	3101.48	300	N
General Precedence	3094.62	127	Y
Constraint Programming	3119	1.76	N

Table 4.2: Time taken to reach the best feasible solution for different models

Another interesting result can be seen in Table 4.2 which shows the time taken to reach the best feasible solution when each model is solving for the full 33 campaigns. From this table it can be seen that the true optimum exists at a combined earliness and lateness minimum of 3094.62 hours. However, the Constraint Programming model is able to find a solution that is only 0.78% away from the true optimum in under two seconds. This shows that there is promise in the Constraint Programming formulation.

4.3 Complete ISP Mathematical and Constraint Programming Models

The purpose of this section is to develop a mathematical and constraint programming model that more closely replicates the efforts that are performed by the expert system at a steel mill. This is accomplished by adding constraints that are specific to a steel making process. Although the constraint programming model did not perform particularly well in the previous section, it is believed that it will perform significantly better under this paradigm. The reason for this belief is that the solution algorithm finds the true optimum (as determined by the general precedence model) in under two seconds, but cannot prove optimality until the entire remaining feasible search space is enumerated. This is one of the drawbacks of the constraint programming methodology. Since the above model is very loosely constrained, the remaining search space is very large. By adding complex and restrictive constraints, it is hypothesized that the search space will be significantly smaller, and the solution time will decrease. This model is compared to that of the immediate precedence model that performed well in the previous case study.

The additional constraints that are necessary are depicted in Figure 4.3. This shows the cyclical nature of the manufacturing process. The previous model makes a distinction between two types of product, type 1 and type 2. In actuality, there are seven subcategories of type 1 (type 1a, b, c, d, e, f, g) and four subcategories of type 2 (type 2a, b, c, d). The highest value product is the "exposed" product. This is the product that is used for automotive exterior purposes. It is required to have the least amount of defects. As such, a certain type of material is run for a specified number of tons to prepare the production line for this exposed product. Then, the high value

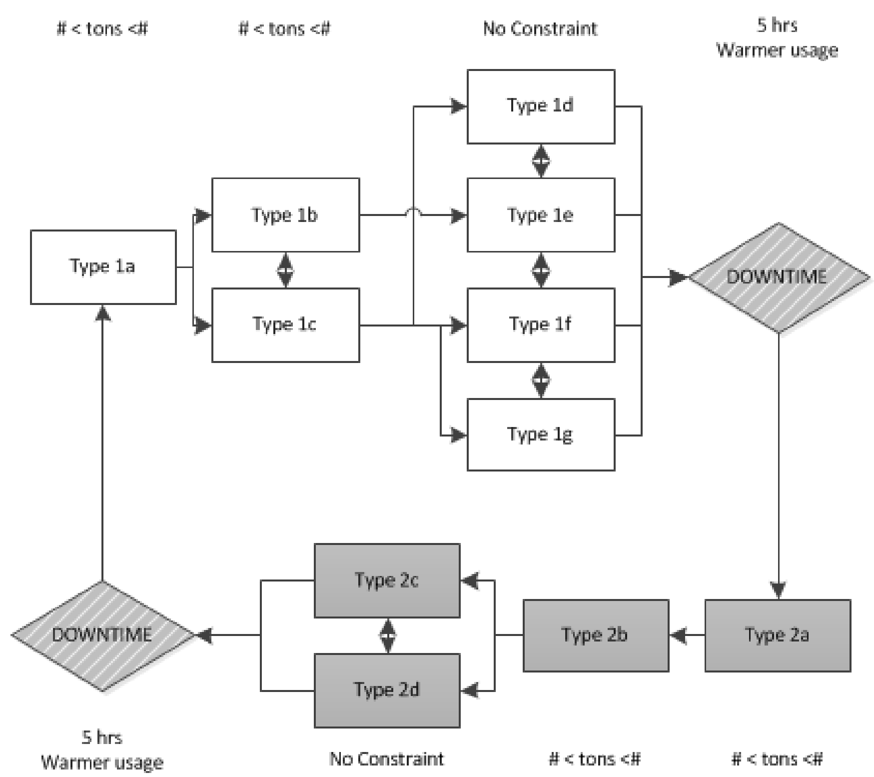

Figure 4.3: Cycle of campaigns and specific steel mill constraints

exposed material can be run for a specified maximum. From a cost efficiency point, manufacturing stipulates that this material should be run at specific minimum capacity. Any less would waste roller usage. At this point, the rollers used to move the material are considered to be sub-par for this application. Then the remaining type 1 material can be run. A similar situation occurs for type 2 material. Once this is completed, the type 1 product can be sequenced again and a cycle is created.

4.3.1 ISP Immediate Precedence Model

In order to handle the cycling of the campaigns, we use the subscript m in the y_{im} variable to allocate a campaign to a cycle instead of a production line. This is possible since only one production line is being considered. The set k represents the campaign cycles (i.e. cycle 1, cycle 2, etc.) and the assignment variable becomes y_{ik}.

The objective function of this model is changed to a minimization of the number of late campaigns in 4.34. N_i is a binary variable that takes on the value of 1 if an order is late, and 0 otherwise. In reality, a customers goodwill is lost only if an order is late. Producing a product early is not valuable to a consumer.

$$\text{Min} \sum_i N_i \tag{4.34}$$

Constraint 4.19 is modified to reflect this change in objective function as shown in 4.35 where B is a sufficiently large number.

$$Te_i \leq D_i + BN_i \qquad \forall i \in I \tag{4.35}$$

The only additional constraint needed keeps track of the timing of the campaigns within the cycles. This is shown in Equation 4.36

$$Te_j + B(2 - y_{ik} - y_{ik'}) \geq Te_i + \tau_{ij} + T_j \qquad \forall i \in I, j \in I, k \in K, k' \in K : k' \neq k \quad (4.36)$$

This equation states that the ending time of campaign j is greater than the ending time of campaign i plus its processing and changeover time if and only if campaign j occurs at a later cycle than campaign i

These relatively simple modifications are all that is necessary to convert the immediate precedence model into one that also considers cycling. The number of cycles however is an input to the model and must be estimated. An estimation that is too small can result in additional late campaigns. Conversely, overestimating the number of cycles increases the number of variables resulting in slower solution speed as a key variable y_{ik} scales with the number of cycles. An iterative procedure is necessary to narrow in on the optimal number of cycles. For this problem, it has been determined that 6 cycles is ideal for 33 campaigns.

4.3.2 ISP Constraint Programming Model

In order to model the cyclical nature in the IBM Optimization Programming Language, new functions, variables and sets must be defined. First, it is necessary to explain that the set i of campaigns is in fact a tuple. A tuple is a data structure that stores an identifier and a list of elements that are linked with that identifier. In programming, a classic example of a tuple is to store contact information. The identifier would be the contacts name, and the elements linked with that identifier would be their phone number and address. This allowable set notation is one of the benefits of constraint programming. The tuple created for this problem stores the campaign processing time, due date, weight, and type. An example of the tuple is provided below:

$$< \mathbf{C1}, 5 \text{ hours}, 144 \text{ hours}, 2592 \text{ tons}, \text{type 1b} >$$

$$< \mathbf{C2}, 28 \text{ hours}, 321 \text{ hours}, 391 \text{ tons}, \text{type 2c} >$$

$$< \mathbf{C3}, 7 \text{ hours}, 168 \text{ hours}, 1473 \text{ tons}, \text{type 1a} >$$

In words, Campaign 1 takes 5 hours to be processed, it is due in 144 hours, the steel has a weight of 2592 tons, and the product type dictates it is to be used for automotive exposed purposes. With this definition of a tuple, individual elements of campaign i can be accessed with the dot nomenclature. For example, entering "**C1.type**" would return "type 1b".

An additional added set k represents the campaign cycles (i.e. cycle 1, cycle 2, etc.) and the set j represents the campaign types (type 1a, type 2b, etc.). A new interval variable that uses this set is defined in OPL as

$$C^{a}_{ik} \text{ optional, size } i.pt \tag{4.37}$$

This variable states that the interval representing campaign i exists in every cycle k with processing time pt. However, the campaign's existence is optional. Figure 4.4 explains this concept. This figure shows a small problem including five campaigns and two cycles. Each campaign must be allocated to a cycle. The program begins by allocating all five campaigns to each cycle as shown in part (a). A constraint shown later will restrict C^{a}_{ik} to exist in only one cycle. The solution algorithm will then test different locations for each campaign until the sample solution shown in part (b) is

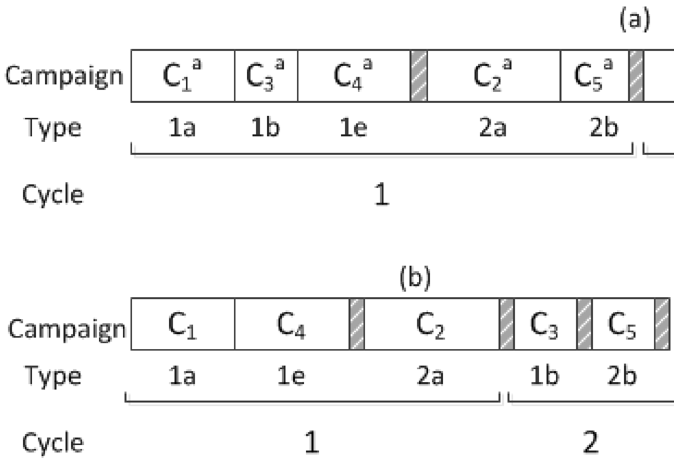

Figure 4.4: Selection of sequenced campaigns from the list of alternates

found. This becomes the final location of campaign C_i.

Finally, the OPL allows us to use built-in functions, or define functions of our own. The first built-in function used is called *cumulFunction* and is shown in Equation 4.38.

$$\text{cumulFunction } RollerUsage_{jk} = \sum_i \text{stepAtStart}(C^a_{ik}, i.tons) \tag{4.38}$$

This function behaves similarly to a step function in the process control literature. It tracks the value that the step is currently at. This works in conjunction with the *stepAtStart* function. This adds a step change to the *cumulFunction* at the start of the interval C^a_{ik}. The size of the step that is made is equal to the weight (in tons) of the campaign using parameter $i.tons$. This function will be used in constraint 4.44.

The objective function of this model is identical to that of the ISP Immediate Precedence model and is shown in 4.39.

$$\text{Min} \sum_i N_i \tag{4.39}$$

Equation 4.40 remains the same as the previous CP model. This ensures the sequence of campaigns do not overlap and includes changeover time if a type 1 product is sequenced next to a type 2 product or visa versa.

$$\text{s.t. noOverlap}(S, \tau_{ii'}) \qquad \forall i \in I, i' \in I \tag{4.40}$$

The *alternative* function in Equation 4.41 is boolean in nature and states that if interval C_i is present, then one and only one of the optional intervals C_{ik}^a must be present as well.

$$\text{alternative } (C_i, C_{i,k}^a) \qquad \forall i \in I, k \in K \tag{4.41}$$

Practically, this constraint allocates interval C_i to one and only one of the cycles k. For example, if campaign 3 is allocated to cycle 4, then the alternative function would read

$$\text{alternative } (C_3, C_{3,4}^a) = TRUE$$

Equation 4.42 uses the built in function *endBeforeStart*. This function states that the first term in the parenthesis must end before the second term in the parenthesis. In this instance, this function is used for the restrict the timing of the cycles. It ensures that all campaigns allocated to cycle k are completed before all campaigns in cycle k' start, where $k' > k$.

$$\text{endBeforeStart } (C_{i,k}^a, C_{i',k'}^a) \qquad \forall i \in I, k \in K, i' \in I, k' \in K : k' > k \tag{4.42}$$

Similarly, Equation 4.43 restricts the timing of campaigns inside a cycle. This ensures that all type 1a material occurs before type 1b material, which occurs before all type

1c, 1d, etc. Here, the dot operator is used again to accesses the type of the campaign.

$$\text{endBeforeStart}\left(C^a_{i,k}, C^a_{i',k}\right) \qquad \forall i \in I, k \in K, i' \in I, i'.type \geq i.type \tag{4.43}$$

The constraint in Equation 4.44 is used in conjunction with the *cumulFunction* defined previously. This ensures that the quality restriction on the capacity of each roller is not violated.

$$\text{RollerUsage}_{jk} \leq M_j \qquad \forall j \in J, k \in K \tag{4.44}$$

Finally, Equation 4.45 defines the lateness binary variable (N_i) where B is a sufficiently large number. Intuitively, the *endOf* function returns the ending time of the campaign interval i.

$$\text{endof}\left(C_i\right) \leq D_i + BN_i \qquad \forall i \in I \tag{4.45}$$

The following case study will investigate these models' performance at solving the industrial campaign planning problem.

Nomenclature

Sets	
I	Campaigns
J	Campaign type
K	Campaign cycles
Parameters	
T_i	Processing Time
$\tau_{ii'}$	Changeover Time
D_i	Campaign Due Date
W_i	Weight in tons of each campaign
M_j	Capacity limit on each type per cycle
B	Large Number
Variables	
C_i	Int var: One interval per Campaign
C^a_{ik} optional, size T_i	Int var: Campaign C can optionally exist in alt cycle locations
S in C types J	Seq var: Represents the sequence of campaigns C with type J
N_i	Binary variable: equal to 1 if campaign i is late

4.3.3 Case Study 4 - Industrial Solutions

The same data set that is used in Case Study 3, is used here. The immediate precedence model contains 3644 constraints, 2349 binary variables, and 35 continuous variables. This is a significant increase in the number of binaries compared to original immediate precedence model. The constraint programming model contains 18753 constraints and 265 variables as dictated by the IBM OPL. The important factor is that the number of constraints is significantly larger than the previous problem, thereby reducing the search space. The progression towards the optimum for both of these models can be seen in Figure 4.5. This figure reports a data point for every feasible solution found. From this figure it can be seen that the constraint programming formulation outperforms the mathematical programming model. After a similar amount of time, CPLEX is able to return a best solution of 10 late campaigns. The optimality gap at this point in time is 60%. Conversely, the constraint programming model is able to reach the true optimum in 8 seconds, and only takes 60 seconds to prove optimality.

The solution to the full ISP model as reported by the IBM Suite can be seen in Figure 4.6. There are six late campaigns in this solution and 5 changeovers.

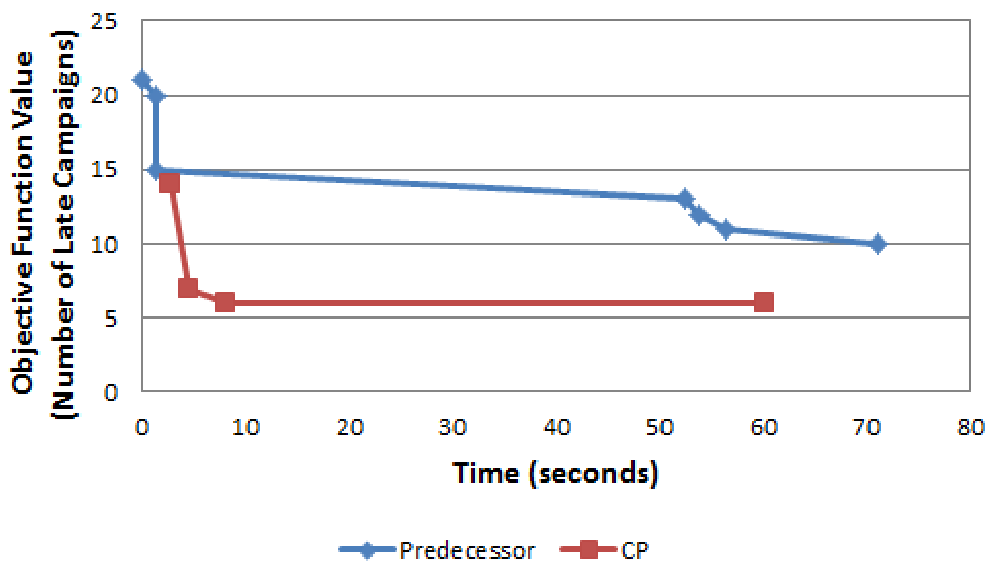

Figure 4.5: Reduction in objective function value as a function of solution time for the number of late campaigns

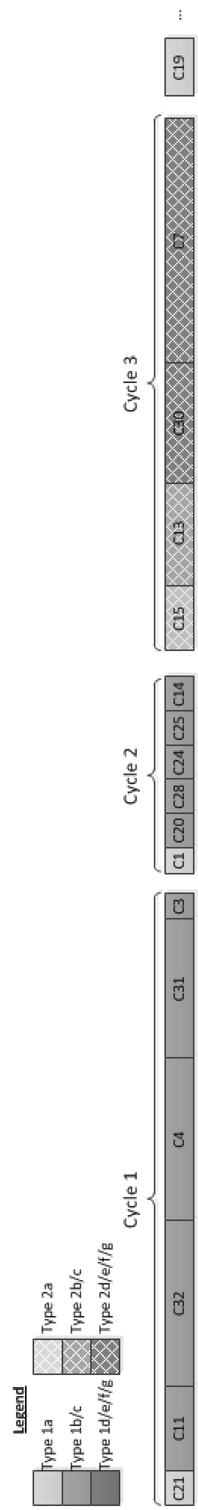

Figure 4.6: Gantt chart of OPL solution for 33 campaigns

Chapter 5

Conclusions and Recommendations

5.1 Conclusions

The main conclusions to be drawn from this thesis are as follows:

1. **Stochastic aggregate production planning formulation.** After introducing a deterministic aggregate production planning formulation, a two-stage stochastic model was developed in an effort to include uncertainty in demand. It was found that the stochastic model does not provide any improvement when the objective is to meet inventory targets set out by the long term planning group. The is due to the fact that 98.7% of the demand uncertainty occurs outside of the non-anticipative window. Additionally, the inventory target objective does not push the model to its feasibility limits. The benefits of the stochastic models ability to predict and hedge against infeasibility are nullified.

2. **Cost savings by objective function selection in stochastic production**

planning. It was hypothesized that the inventory targets proposed by the long term planning group are causing an excessive amount of inventory to be held. The two-stage stochastic production planning model developed is extended to include a cost minimization through the development of a cost objective and its corresponding constraints. It was found that this modelling technique was able to reduce operation costs by 2.1%. However, this aggressive cost minimization introduced oscillations in the production profile. A compromise was reached by reintroducing the idea of inventory targets by making them a variable of the optimization model. This removed production oscillations with only a 0.2% reduction in cost savings.

3. **Analysis of mathematical programming formulations for use in scheduling problems.** Solving scheduling problems for industrially sized problems is computationally difficult to solve. The computational efficiency is largely dependent on the choice of the mathematical model used. As such, several continuous time mathematical programming formulations were introduced and tested using data from the steel industry. The problem under consideration is a sequencing problem with 33 campaigns and sequence dependent changeovers. The objective function used is a minimization of lateness. There are no constraints on the allowed precedence of the jobs. It was found that a general precedence model is the most efficient model available for this particular problem. The model is able to solve 6 weeks of campaigns for one product line in approximately 2 minutes of computation time.

4. **Constraint programming formulation alternative.** A constraint programming formulation was developed as an alternative model to the mathematical programming formulations. This formulation was selected as it had been

shown to perform well for combinatorial optimization problems, particularly when considering scheduling problems. The model considered 33 campaigns with sequence dependent changeovers. Additional constraints were added to restrict the feasible precedence of jobs. The allowable sequence forms a cyclical structure that is type and family dependent. Additional constraints were added to this problem that include the quality of steel produced dependent on its sequence. The constraint programming model was able to computationally outperform the mathematical formulations when considering an industrially sized steel making problem.

5.2 Recommendations for Further Work

Further avenues to explore are summarized in the following list:

1. **Rolling Horizon Planning.** The stochastic production planning models investigated in this thesis assume that an initial first stage decision must be made, and then one final second stage reactionary decision can be implemented after the uncertainty is realized. In actuality, demand forecasts are updated every week and the operators are allowed to reevaluate their planning decisions. This is commonly referred to in the research literature as a rolling horizon. A modelling extension that can replicate this is multi-period, multi-stage stochastic formulation.

2. **Reactive Scheduling.** An extension that would benefit from additional research is in the area of reactive scheduling. The steel mill planners are often faced with scenarios where unexpected occurrences necessitate the immediate

change of the current schedule. This may be a result of an unexpected shutdown or a last-minute customer order. A useful modelling extension would be to re-optimize the previously solved schedule to include the new variables while deviating from the previous plan as minimally as possible.

3. **Campaign Planning for Multiple Production Lines.** Currently, the model is able to provide a solution for one production line. In reality, there are 10 production lines running in parallel at the steel mill. Additionally, some campaigns are able to be produced on multiple production lines. This adds a significant amount of complexity and room for optimality that is not currently included in the model. However, adding multiple production lines will increase the number of variables of the optimization model substantially. This work will need to investigate other methods of speeding up solution times such as disaggregation and decomposition.

4. **Integration of Planning and Scheduling.** The planning problem in the first portion of this thesis and the scheduling portion in the second are currently independently solved. In reality, the decisions made in each of these respective layers affect decisions of the other layers. The first step in integrating the multiple layers should be to include important constraints used in primary planning into the campaign planning model. The next step towards integration would be passing the results of the scheduling model to the planning model and iterating until a plant-wide feasible solution is found.

List of References

A. Atighehchian, M. Bijari, and G. Tarkesh. A novel hybrid algorithm for scheduling steel-making continous casting production. *Computers and Operations Research*, 36:2450–2461, 2009. 8

A. Balakrishnan and J. Geunes. Production planning with flexible product specifications: An application to specialty steel manufacturing. *Operations Research*, 51: 94–112, 2003. 19

P. Baptiste, C. Le Pape, and W. Nuijten. Constraint-based scheduling: applying constraint programming to scheduling problems. In *Vol. 39*. Springer Science and Business Media, 2001. 16, 17

A. Bellabdaoui and J. Teghem. A mixed-integer linear programming model for the continuous casting planning. *Internation Journal of Production Economics*, 104: 260–270, 2006. 7, 8

P. Berling. Holding cost determination: An activity-based cost approach. *Production Economics*, 112:829–840, 2008. 48

R. Bisdorff and S. Laurent. Industrial linear optimization problems solved by constraint logic programming. *European Journal of Operational Research*, 84:92–95, 1995. 17

M. Chen and W. Wang. A linear programming model for integrated steel production and distribution planning. *International Journal of Operations and Production Management*, 17(6):592–610, 1997. 15, 19

T. Fabian. A linear programming model of integrated iron and steel production. *Management Science*, 4(4):415–449, 1958. 15

T. Fabian. Blast furnace production planning-A linear programming example. *Management Science*, 14(2):B1–B27, 1967. 15

C. A. Floudas and X. Lin. Continous-time versus discrete-time approaches for scheduling of chemical processes: a review. *Computers and Chemical Engineering*, 28: 2109–2129, 2004. 22

I.E. Grossmann. Challenges in the new millennium: product discovery and design, enterprise and supply chain optimization, global life cycle assessment. *Computers and Chemical Engineering*, 29:29–39, 2004. 2

I. Harjunkoski and I. E. Grossmann. A decomposition approach for the scheduling of a steel plant production. *Computers and Chemical Engineering*, 25:1647–1660, 2001. 15, 33

I. Harjunkoski, I.E. Grossmann, M. Friedrich, and R. Holmes. A Multi-week scheduling approach for the steel-making process. In *Proceedings of FOCAPO 2003*, pages 315–318, 2003. 7, 8, 26

A. C. Hax and H. C. Meal. Hierarchial integration of production planning and scheduling. In Geisler, editor, *Logistics: Studies in the Management Science*. Elsevier, North Holland, 1975. 18

T. L. Jacobs, J. R. Wright, and A. E Cobbs. Optimal inter-process steel production scheduling. *Computers and Operational Research*, 15:497–507, 1988. 14

K. Kim and U. Diwekar. Efficient combinatorial optimization under uncertainty. 1. Algorithmic Development. *Industrial and Engineering Chemistry Research*, 41(5): 1276–1284, 2002. 29

P. Li, H. Arellano-Garcia, and G. Wozny. Chance constrainted programming approach to process optimization under uncertainty. *Computers and Chemical Engineering*, 32:25–45, 2008. 28

IBM OPL Reference Manual. IBM ILOG OPL Language Reference Manual. In IBM, editor, *IBM ILOG OPL V6.3*, pages 1–264. International Business Machines Corportation, 2009. 83

C. A. Mendez, Jaime Cerda, I. E. Grossmann, I. Harjunkoski, and M. Fahl. State-of-the-art review of optimization methods for short-term scheduling of batch processes. *Computers and Chemical Engineering*, 30:913–946, 2006. viii, x, 20, 21, 22, 24, 25, 26, 67, 71

H. Meyr, H. Wagner, and J. Rohde. Structure of advanced planning systems. In H. Stadtler and C. Kilger, editors, *Supply chain management and advanced planning: Concepts, models software and case studies*, pages 99–104. Berlin: Springer-Verlag, 2002. viii, 2

R. P. Mohanty and R. Singh. A hierarchical production planning approach for a steel manufacturing system. *International Journal of Operations and Production Management*, 12.5:69–78, 1992. 19

S. Nahmias and T.L. Olsen. Sales and Operations Planning. In *Production and Operations Analysis*. Waveland Press Inc. 20

S. Nam and R. Logendran. Aggregate production planning - A survey of models and methodologies. *European Journal of Operational Research*, 61:255–272, 1992. 17

H. Okano, A. J. Davenport, M. Trumbo, C. Reddy, K. Yoda, and M. Amano. Finishing line scheduling in the steel industry. *IBM Journal of research and development*, 48 (5.6):811–830, 2004. viii, 8, 9, 10, 25

L. G. Papageorgiou. Optimal campaign planning/scheduling of multipurpose batch/semicontinuous plants. 1. Mathematical formulation. *Industrial Engineering and Chemical Research*, 35:488–509, 1996. 25

C. M. Petersen, K. L. Sorensen, and R. V. V. Vidal. Inter-process synchronization in steel production. *International Journal of Production Research*, 30(6):1415–1426, 1992. 14

J. M. Pinto and I. E. Grossmann. Assignment and sequencing models for the scheduling of process systems. *Annals of Operations Research*, 81:433–466, 1998. 71

Y. Pochet and L. A. Wolsey. Production Planning and MIP. In *Production planning by mixed integer programming*. Springer Science and Business Media. 20

S. Radhakrishnan and J. A. Ventura. Simulated annealing for parallel machine scheduling with earliness-tardiness penalties and sequence-dependent set-up times. *International Journal of Production Research*, 39(10):2233–2252, 2000. 76

C. N. Redwine and D. A. Wismer. A mixed integer programming model for scheduling orders in a steel mill. *Journal of optimization theory and applications*, 14(3):305–318, 1974. 14

H. Richardson. Control your costs–then cut them. *Transportation and Distribution*, 36:94, 1995. x, 49

N. V Sahinidis and I. E. Grossmann. MINLP model for cyclic multiproduct scheduling on continuous parallel lines. *Computers and Chemical Engineering*, 15(2):85–103, 1990. 75

B. Sasidhar. A multiple arc network model of production planning in a steel mill. *International Journal of Production Economics*, 22:195–202, 1991. 15

C. Schulz, S. Engell, and R. Rudolf. Scheduling of a multi-product polymer batch plant. *Technikum Joanneum*, 1998. 26

K. Sekiguchi, Y. Seki, N. Okitani, and M. Fukuda. The advanced set-up and control system for Dofasco's tandem cold mill. *IEEE Transactions on Industry Applications*, 32(3):608–616, 1996. 8

N. K. Shah. Process industry supply chains: Advances and challenges. *Computers and Chemical Engineering*, 29(6):1225–1235, 2005),. 27

J. Snyman. Practical mathematical optimization: an introduction to basic optimiztion theory and classical and new gradient-based algorithms. In *Vol. 97*. Springer Science and Business Media, 2005. 12

M. S. Suh, A. Lee, Y. J. Lee, and Y. K. Ko. Evaluation of ordering strategies for constraint satisfaction reactive scheduling. *Decision Support Systems*, 22:187–197, 1998. 17

L. Tang, J. Liu, A. Rong, and Z. Yang. A mathematical programming model for scheduling steelmaking-continuous casting production. *European Journal of Operational Research*, 120:423–435, 2000. 8

L. Tang, J. Liu, A. Rong, and Z. Yang. A review of planning and scheduling systems and methods for integrated steel production. *European Journal of Operational Research*, 133:1–20, 2001. 2, 6, 7

M. C. Wellons and G. Reklaitis. Scheduling of multipurpose batch chemical plants. 1. Formation of single-product campaigns. *Industrial Engineering and Chemical Research*, 30:671–688, 1998. 25

J. R. Wright, M. H. Houck, L. L. Archibald, D. W. H. Chong, J. T. Diamond, S. S. Egly, H. W. Gallivan, T. L. Jacobs, and R. P Peterson. The application of systems engineering to hot mill production scheduling. *Rep. to Bethlehem Steel Corp.*, Ind, 1984. 15

FICO Xpress. MIP formulations and linearizations - Quick Reference. In FICO, editor, *FICO Xpress Optimization Suite*, page 7. Fair Isaac Corporation, 2009. 75

S. Zanoni and L Zavanella. Model and analysis of integrated production-inventory system: The case of steel production. *International Journal of Production Economics*, 93–94:197–205, 2005. 19

Z. Zhu and R. B. Heady. Minimizing the sum of earliness/tardiness in multi-machine scheduling: a mixed integer programming approach. *Computers and Industrial Engineering*, 38:297–305, 2000. 80, 81