CHAPTER 1: INTRODUCTION

This chapter aims to introduce the reader to the Sasol Mining and Sasol Coal Supply environments and to explain the scope, goal and objectives of this project. A brief introduction to the operations under consideration will be given and the structure of the rest of this document will be discussed.

1.1. BACKGROUND

Sasol Mining (Pty) Ltd (hereafter called Sasol Mining) supplies coal to Sasol Synfuels (Pty) Ltd (hereafter called Synfuels) to be used in the coal-to-liquids technology. These operations are situated in Secunda, Mpumalanga, South Africa, and are part of the Sasol Group of Companies.

Sasol Mining produces approximately 45 million ton of coal annually, of which approximately 40.2 million ton are consumed by the Synfuels factory. Sasol Mining's operations in Secunda consist of six underground coal mines. These mines are in a radius of approximately 30 km around the Synfuels factory.

The conveying of coal from the different mines to Synfuels, which needs a 24 hour supply of coal, poses a very specific logistic challenge. Sasol Mining has a coal conveying and handling facility, Sasol Coal Supply (hereafter called SCS), which conveys coal from the mines, supplies coal to Synfuels, and acts as a supply buffer between the mines and Synfuels.

1.2. PROJECT GOAL AND OBJECTIVES

The goal of this project is to develop the following:

An operational scheduling model to be used at Sasol Coal Supply.

The scheduling model must meet the following requirements:

- Optimised schedule for extracting coal from the mines' bunkers to SCS, minimising the occurrences of throwing out coal at the bunkers.
- Optimised schedule for stacking coal on the stockpiles and the strategic stockpiles.
- Optimised schedule for reclaiming coal from the stockpiles or conveying coal directly from the mines to the factory.

- Optimised coal supply to the factory by minimising deviation from the blend plan.
- Optimised infrastructure utilisation by integrating equipment availability, operations and maintenance activities and minimising machine runtime.

In short, the model must indicate which mine's coal is to be conveyed at which time, to which stockpile and how it should be reclaimed to meet the factory's coal requirements.

1.3. APPROACH

A mathematical approach is used to develop the scheduling model described above. The problem is defined as a *Mixed Integer Non-linear Programming (MINLP)* problem. The General Algebraic Modelling System (GAMS) is the software used to model and solve the problem. The technique and the software will be discussed in detail in Chapter 2.

1.4. ENVIRONMENT

In this section, the operating environment for the scheduling model will be discussed briefly.

Table 1.1: Sasol Mining's operations in Secunda

Mine / plant	Type of operation	Coal product delivered
Brandspruit mine	Underground mine	Coal for Synfuels market*
Middelbult mine	Underground mine	Coal for Synfuels market*
Bosjespruit mine	Underground mine	Coal for Synfuels market*
Syferfontein mine	Underground mine	Coal for Synfuels market*
Twistdraai complex: 1. Central shaft 2. East shaft 3. West shaft	 Underground mine Underground mine Underground mine	 1. Coal for Synfuels market* 2. Coal for Export coal market Coal for Export coal market Coal for Export coal market
Export Plant	Coal beneficiation plant	1. Pure export coal 2. Middlings to Synfuels* 3. Waste
Sasol Coal Supply (SCS)	Coal handling and blending	A homogenous blend of coal to Synfuels

**Indication of the six coal streams conveyed to SCS to be supplied to Synfuels*

1.4.1. Sasol Mining

a. Coal sources

The Sasol Mining complex in Secunda consists of the operations summarised in Table1.1. This project focuses on the operations supplying coal to Synfuels, and therefore, the *export operations will not be discussed*. Thus, for the purpose of this project, there are 6 sources of coal in the Secunda complex supplying coal to satisfy Synfuels' demand for coal:

1. Brandspruit mine.
2. Middelbult mine.
3. Bosjespruit mine.
4. Syferfontein mine.
5. Twistdraai Central mine.
6. Middlings from the Export plant.

b. Coal quality and blend

Coal has different quality properties which are classified as physical and chemical properties. The coal from each of the above mentioned sources differ in quality. These quality properties and its variation have a significant impact on the Synfuels gasification process, both in terms of production volume and process stability. Therefore, Sasol's Research and Development department has set maximum limits for the amount of coal from a certain source that the factory receives at any given time. These limits can change over time, as the sources' coal quality change.

In addition to the maximum limits, a weekly operating plan is calculated, determining the best combination of each source's contribution to the coal supplied to Synfuels. This plan is referred to as the 'blend plan'. The blend plan takes the following factors into consideration:

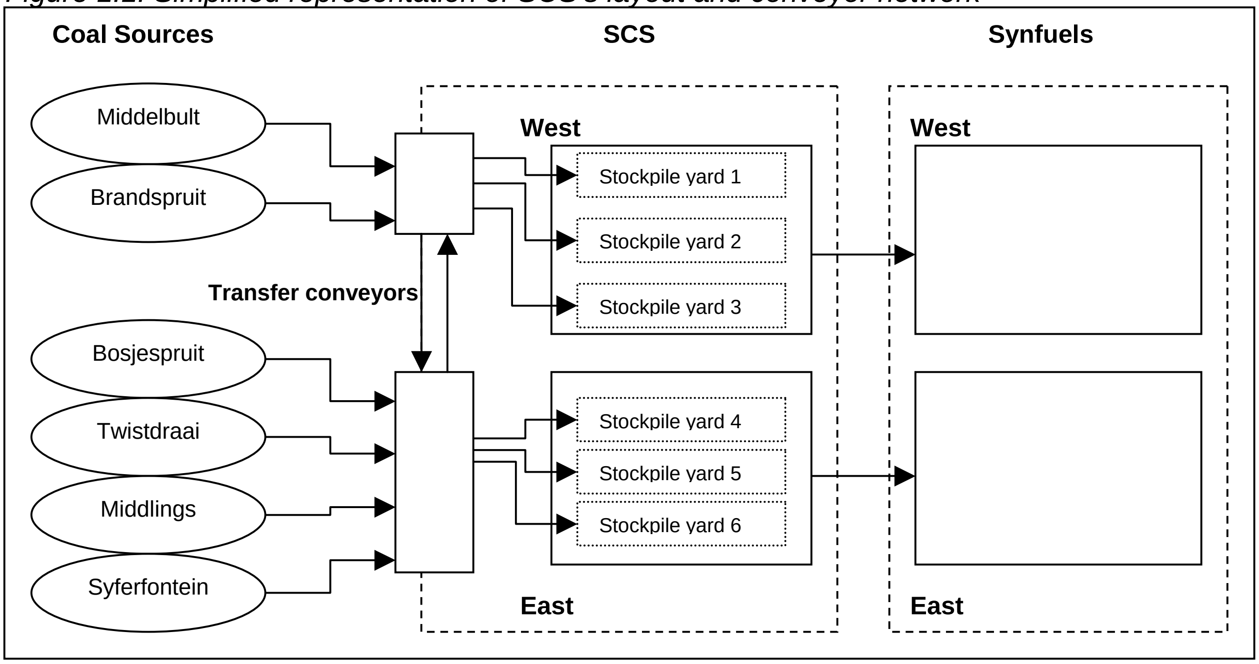
- The planned mine production for the week.
- The predicted coal quality from the different mines.
- The balance of stock between the East and the West plant.
- Other operational factors.

The blend plan is the guideline during the stacking and reclaiming processes, to ensure optimum gas production at Synfuels.

c. SCS

SCS is Sasol Mining's coal conveying and blending facility. It consists of a large conveyor network, connecting the coal sources to a stockpiling facility. This facility is divided into two identical sides: East and West. Each side has three stockpile yards where coal from the sources are stacked. From the stockpiles, the coal is reclaimed and conveyed to Synfuels. Synfuels is also divided into two identical sides: East and West. The Eastern side of SCS only supplies coal to the Eastern side of Synfuels and vice versa. Figure 1.1 illustrates this situation.

Figure 1.1: Simplified representation of SCS's layout and conveyor network



SCS has two main roles:

1. *To be a supply buffer between the mines and Synfuels:*

   Synfuels has a 24 hour per day continuous operation, resulting in a continuous demand for coal. However, the mines' supply are not continuous. The mines have a two shift system during the week, have only one shift on a Saturday and none on a Sunday and public holidays.

   In order to supply the continuous demand of coal, SCS has two types of coal in stock:

   - Normal stockpiles:

     The coal on these stockpiles is used to supply Synfuels' continuous demand of coal from day to day. Coal is stacked and reclaimed on a daily basis on these stockpiles. The stock turnover on these stockpiles is approximately 2.5 times per week.

   - Strategic stockpiles:

     The strategic stockpile coal is used when the supply from the mines does not match the factory demand. Periods of low production by the mines and periods with public holidays such as Christmas and Easter are examples of times when the coal from the strategic stockpiles are loaded back onto the normal stockpiles and reclaimed to supply the factory's demand. The reverse also holds. In times of high production by the mines and periods of low factory consumption such as the annual shutdown, the excess coal is thrown out on the strategic stockpiles, compacted and stored for later use.

2. *To minimise the quality variation of coal supplied to Synfuels:*
   As described in b. above, the mines' coal quality differ from one to another. SCS uses the normal stockpiles to blend the coal from the different mines. By blending, quality variation of the coal sent to the factory is minimised. The detail operations are described in section 1.5.1.

1.4.2. Sasol Synfuels

For a complete overview of the environment, a brief description of a few Synfuels activities is necessary.

a. Coal Processing

When the coal arrives at Synfuels from SCS, it goes through a wet screening process where most fine coal is removed. The excess fine coal is sent to the Steam Plant.

b. Gasification

At Gasification the coal is pressurised under high temperatures. Steam and oxygen are added. The coal is gasified and the products are raw gas and ash. The gas is used further down stream in Synfuels' unique coal-to-liquid technology.

c. Steam Plant

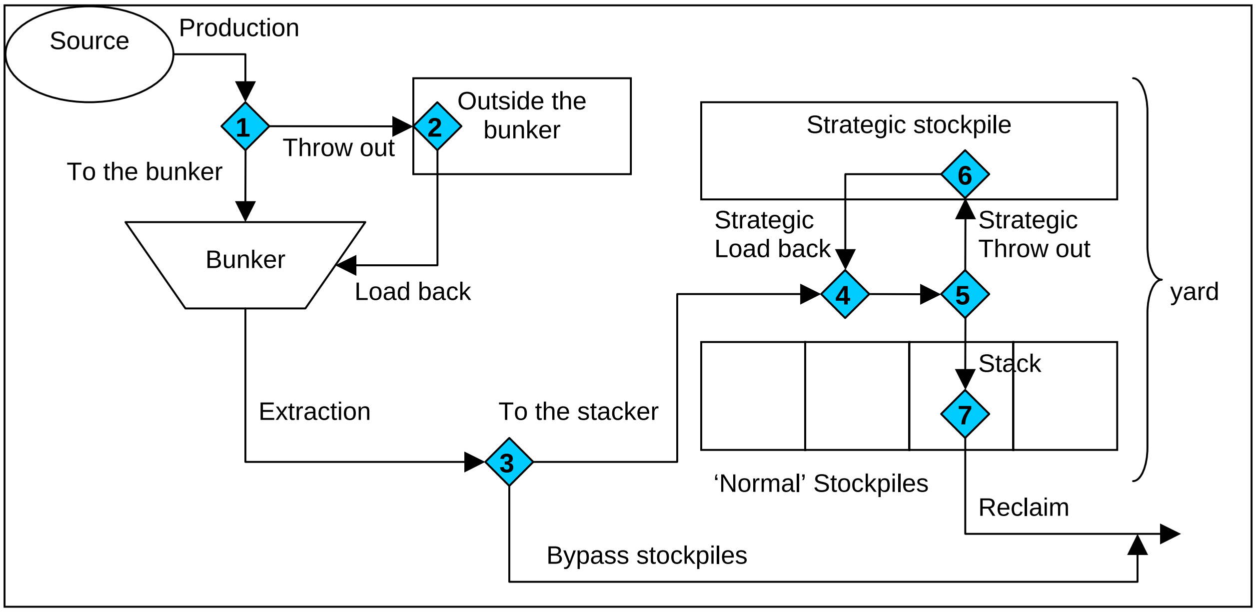
The Steam Plant uses fine coal to generate steam. The steam is used in the gasification process and in the rest of the Synfuels plant.

1.5. PROBLEM DESCRIPTION

1.5.1. Detail operations

To get better insight into the scheduling problem, the operations at SCS must be described in more detail. The basic coal flow options for one coal source are illustrated by Figure 1.2. The decision points are indicated with blue diamond blocks. These blocks represent ‘or-gates’ where only one of the relevant options may be activated.

Figure 1.2: Coal flow options and decision points for one source



A source supplies coal at a certain production rate (which varies from mine to mine and from day to day). The coal is normally accumulated in a temporary storage place, called a bunker. When the bunker reaches capacity, the coal from the source may be directly thrown out outside the bunker, bypassing the bunker totally (decision point 1). This precaution only acts as an emergency measure to prevent stopping the production of a mine. The coal outside the bunker must be loaded back into the bunker with front-end loaders when the emergency has passed. Throwing out and loading back may not happen simultaneously (decision point 2). The handling of coal outside the bunker has two disadvantages, namely the additional handling cost of the front-end loaders (cost penalty) and the additional fine coal created by the additional handling of the coal (quality penalty).

When coal is extracted from the bunkers and conveyed to the stockpile yards, there are two options for the coal flow at the stockpile yard. The coal can either be conveyed to the stacker or it can be directed to bypass the stockpiles totally (decision point 3). *Note that, if coal from one mine is bypassed at a certain stockpile, coal from another mine may still be conveyed to the stacker to be thrown out or to be stacked.*

If the coal is conveyed to the stacker, there are again two options for the coal flow. The coal can either be stacked on one of the individual stockpiles on that specific yard or it can be thrown out on the strategic stockpile to be compacted and stored for the longer term (decision point 5). Coal can also be loaded back from the strategic stockpiles. In this case, coal is loaded onto the conveyor feeding the stacker, and the coal is stacked on one of the individual stockpiles on that specific yard. Note that loading back coal from the strategic stockpile and stacking coal that was extracted from a bunker may not happen simultaneously (decision point 4).

Throwing out on the strategic stockpile and loading back from the strategic stockpile may not happen simultaneously (decision point 6). Again, both of these actions require the coal to be handled with front-end loaders and compacting rollers, which add cost and create additional fine coal.

At SCS there are three stockpiles yards East and three stockpile yards West. Each stockpile yard has normal stockpiles and a strategic stockpile (refer to section 1.4.1.c). There is only one stacker and one reclaimer per stockpile yard. The stockpile yard can be divided to accommodate four individual stockpiles per yard. In a typical operational example, the stacker may be stacking on a stockpile at the one end of the yard and simultaneously the reclaimer may be reclaiming coal from another stockpile at the other end of the same yard. However, one individual stockpile may not be stacked and reclaimed simultaneously (decision point 7).

As mentioned above, the stacking and reclaiming process is used to blend the coal from the different sources. The blending method can be summarised as follows:

- When starting a new stockpile, coal is stacked on the stockpile in horizontal layers across the length of the stockpile. As the stockpile progresses, different mines' coal make up the different layers of coal within the stockpile. The stacking process stops as soon as the stockpile reaches a certain capacity (ton per meter).
- A stockpile can be reclaimed as soon as it is stacked to capacity. A stockpile is reclaimed steadily from one side, resulting in the stockpile being reclaimed in vertical 'slices'. This process ensures that the coal conveyed to the factories at any given time contains a portion of every layer that was stacked on that specific stockpile.

The reclaimed coal together with the bypassed coal are sent to the factory, to meet the factory demand.

1.5.2. Problem Statement

The description of the coal flow above (with the exception of the bypass option), applies to all six sources and the six stockpile yards at SCS (each stockpile yard with its own strategic stockpile and four possible normal stockpiles). As a result, the amount of coal flow options to be managed escalates dramatically (Table 1.2).

Table 1.2: Coal flow options for the total scheduling problem:

Action considered	Number of sources	Number of stockpiles	Total options
Throw out at bunker	6	-	6
Load back at bunker	6	-	6
Extracting coal from a bunker to a normal stockpile	6	6 x 4	144
Extracting coal from a bunker to a strategic stockpile	6	6	36
Extracting coal from a bunker to bypass	5	4	20
Loading back from a strategic stockpile to a normal stockpile	-	6 x 4	24
Reclaiming coal from a normal stockpile	-	6 x 4	24
TOTAL options available:			**260**

In addition to the complexity illustrated in Table 1.2, the following must be kept in mind:

- SCS must comply with the blend limits set by Synfuels.
- SCS must supply Synfuels with a constant supply of coal.
- The bunkers at the mines must be operated as empty as possible to prevent throw out.
- Maintenance schedules must be incorporated in the daily operations at SCS.

It is very difficult for a control room operator to manage a complex system like this, let alone to optimise the operation. Therefore, SCS requires a scheduling model to assist the control room operators in operating the SCS plant efficiently and optimally.

1.5.3. Project scope

The boundaries of this project are defined as follows:

a. Included in the scope:

- The detail coal flow management as described in 1.5.1.
- Scheduling to comply to blend limits and minimising the deviation from the weekly blend plan.
- Minimising coal thrown out at the mine bunkers.
- Optimising the incorporation of maintenance schedules.

b. Excluded from the scope:

- Detail mining operations.
- Any operation beyond SCS (after reclaiming).
- The detail coal quality from each source.

1.6. DOCUMENT OUTLINE

The rest of this document is divided in another four chapters. Chapter 2 introduces the reader to the specific scheduling techniques with an explanatory example to compare the different techniques' performance, advantages and disadvantages. Chapter 3 explains the basic mathematical formulation of the scheduling problem described above. In Chapter 4, some improvement techniques are applied to ensure that the model solves in a time appropriate for the operational use of the model. Finally, Chapter 5 contains some implementation comments and concluding remarks.

Chapter 2: Literature Study and Technique Evaluation

The literature study aims to introduce the reader to the most recent Mixed Integer Non-Linear Programming (MINLP) scheduling techniques, listing their advantages and disadvantages. An example problem is used to illustrate the application of the different techniques. Each model's size and performance are evaluated. Finally, an improved binary variable formulation and an improved non-linear solving method are presented and the results illustrated with the same example problem. A motivation is given for the technique chosen to formulate the SCS scheduling model.

2.1. INTRODUCTION

The general scheduling problem entails the determination of *when*, *where* and *how* to produce a set of products, given requirements in a specific time horizon, a set of limited resources, and processing recipes (Floudas and Lin, 2004). Scheduling focuses on the daily operational aspects which are of concern in order to achieve production targets. These targets are the output of a planning process and therefore it is important to distinguish between planning and scheduling. Planning is aimed at long-term economic issues (Zhu and Majozi, 2001b) such as market demand, supply patterns and total plant capacity. Planning tends to overlook detail operational aspects such as change-over time, detail product flow and sequencing. Therefore, scheduling provides the capability to turn high-level production targets into a series of practical, operational tasks.

A lot of research work has been done in the field of scheduling in the past two decades. The increasing attention to scheduling problems is motivated by three main factors (Floudas and Lin, 2004):

1. Pressure from the industry to improve efficiency and reduce cost.
2. Significant advances in relevant modelling and solution techniques.
3. Rapidly growing computational power.

In scheduling literature, there are two main aspects of scheduling receiving attention. These aspects are the time representation of the model and the use of binary variables in the mathematical formulation. Both aspects are critical in the performance of the model in terms of solution time, accuracy and its application to industrial-sized problems.

Floudas and Lin (2004) provided a classification structure according to the different time representation techniques developed in the past two decades. It can be summarised as follows:

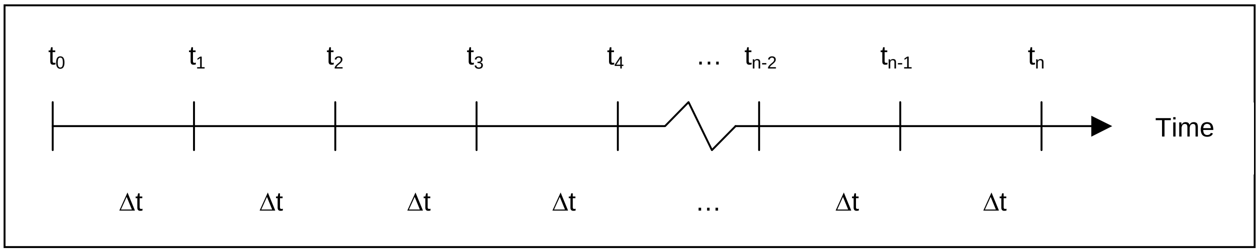
1. Discrete time formulation.
2. Continuous time formulations:
    2.1. Time-slot formulation.
    2.2. Global event based formulation.
    2.3. Unit-specific event based formulation.

In the rest of this chapter, an overview of development in scheduling techniques as well as progress made with binary variable formulation, are discussed according to the structure above. The applicable modelling and solving software are discussed briefly, along with the reporting method for model solutions. Finally, the techniques are compared and evaluated by using a simple example. The relevant technique for the scheduling problem described in Chapter 1 is identified and motivated. At the end, some improved binary and non-linear formulations are discussed.

2.2. DISCRETE TIME

The discrete time approach for time representation in a scheduling problem originates from early attempts to model such problems (Kondili et al., 1993). The applicable time horizon is divided into a number of time intervals of equal duration (Δt), as illustrated in Figure 2.1:

Figure 2.1: Discrete time representation



Events such as the start or the end of a task are associated with the boundaries of these fixed-length intervals. An advantage of this formulation is that a time reference-grid is provided, which simplifies the modelling effort of scheduling constraints.

However, these Δt intervals should be sufficiently small to give an accurate representation of the original problem. Thus, by decreasing the length of the intervals, the error between the model representation and the original problem also decreases. By implication, decreasing the interval length will increase the number of intervals. The number of intervals has a direct effect on the size of the model. Inevitably, this situation leads to a trade-off between model accuracy and model size (solution time). This is the main disadvantage of the discrete time formulation.

Ierapetritou and Floudas (1998a) summarise the main limitations of the of time discretisation method as follows:

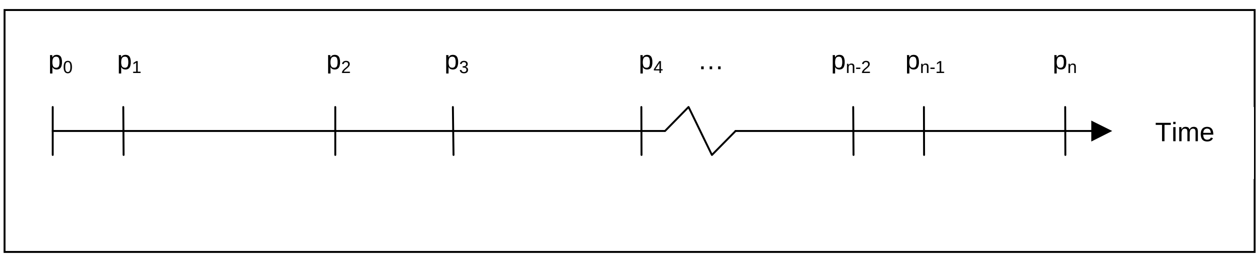
1. It corresponds to an approximation of the time horizon.
2. It results in an unnecessary increase of the number of binary variables in particular and of the overall size of the mathematical model.

Due to the reasons described above, the application of the discrete time approach is limited. Problems with fixed processing or production times may be modelled accurately, but industrial-sized problems will still require a large number of intervals to ensure accuracy. Systems which require more flexible operation times will also result in very large models with substantial deviations from the true solutions.

2.3. CONTINUOUS TIME

As a result of the limitations of the discrete time representation, a new time representation technique was developed during the past decade. Instead of using uniform time intervals, events are potentially allowed to take place at any point in the continuous domain of time (Floudas and Lin, 2004). This concept is illustrated in Figure 2.2:

Figure 2.2: Continuous time representation



Event points are defined at any point in the time horizon, ensuring flexibility and accuracy. The continuous time formulation eliminates a major part of the inactive time intervals used in the discrete time formulation. Therefore, continuous time model sizes are generally much smaller and solution times faster than their discrete time counterparts.

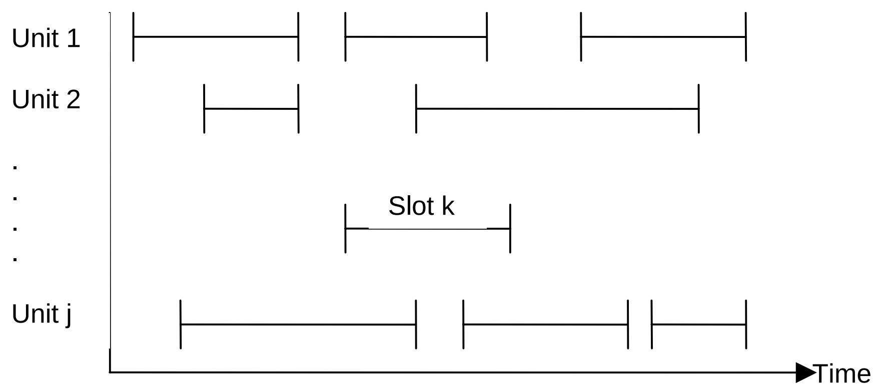
However, the continuous time representation results in more complicated model structures (Floudas and Lin, 2004) which again places a constraint on the size of the problem to which the technique may be applied.

The continuous time representation development can be divided into three sub-groups, namely time-slots, global event based and unit-specific event based, which will be discussed briefly.

2.3.1. Time-slots

Some of the earliest attempts to formulate continuous time scheduling problems were based on the time-slot approach. The basic principle of the time-slot approach is illustrated in Figure 2.3 (Pinto and Grossmann, 1995):

Figure 2.3: Time-slot approach per unit



A slot corresponds to a single event in a certain unit. The start, end and duration of the slot will be the start, end and duration of that specific event. Depending on the formulation of the problem, a slot can also include the set-up time for that specific event. These time-slots can potentially be activated at any point in the continuous time horizon.

This approach was applied to sequential processes (Pinto and Grossmann, 1995) by using a four index binary variable w_{ijkl}, to assign stage l of order i to slot k of unit j. This formulation required a very large number of binary variables (l x i x k x j), which increased the overall size of the mathematical model.

The disadvantages of the time-slot based formulations are:

1. The number of time-slots in these models is pre-defined. Thus, optimality cannot be guaranteed (Floudas and Lin, 2004).
2. Time-slot formulations restrict the time representation of a model and therefore result, by

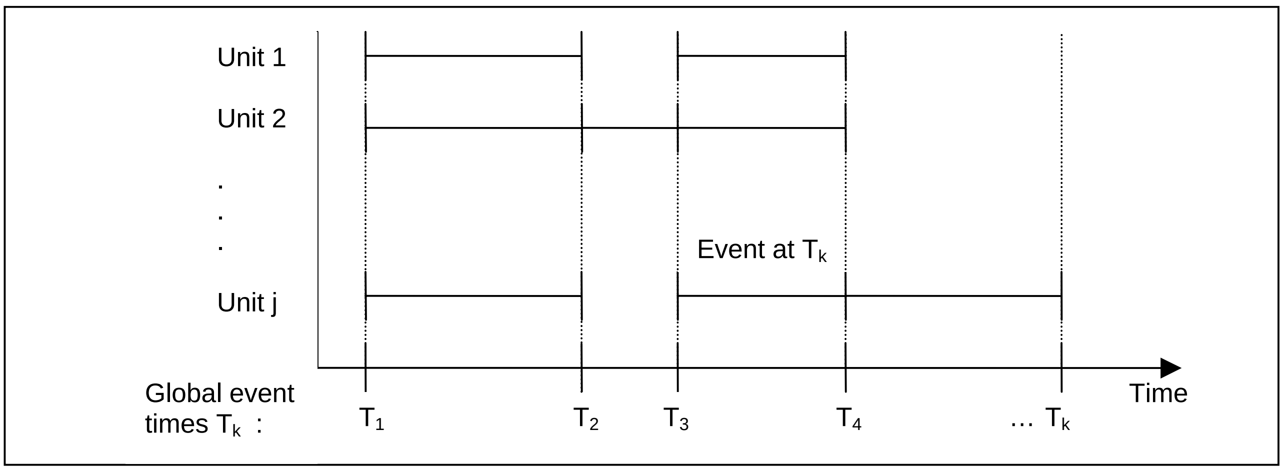
definition, in sub optimal solutions (Ierapetritou and Floudas, 1998a).

2.3.2. Global events

The approach of using global events entails the use of one set of events for *all* tasks and *all* units. Continuous variables are used to determine the timing of events and binary variables are used to assign state changes such as the start or end of an event (Floudas and Lin, 2004).

The first attempt to model a scheduling problem with this approach was made by Zhang and Sargent (1996). Their time representation philosophy is illustrated in Figure 2.4:

Figure 2.4: Global event approach per unit



Zhang and Sargent (1996) proposed the use of the binary variable $X_{ijkk'}$ to indicate the event of task *i* starting in unit *j* at time T_k and ending at $T_{k'}$, with $T_{k'} > T_k$. This formulation leads to a mixed integer non-linear programming (MINLP) model, which can be linearized using Glover's (1975) exact linearization techniques.

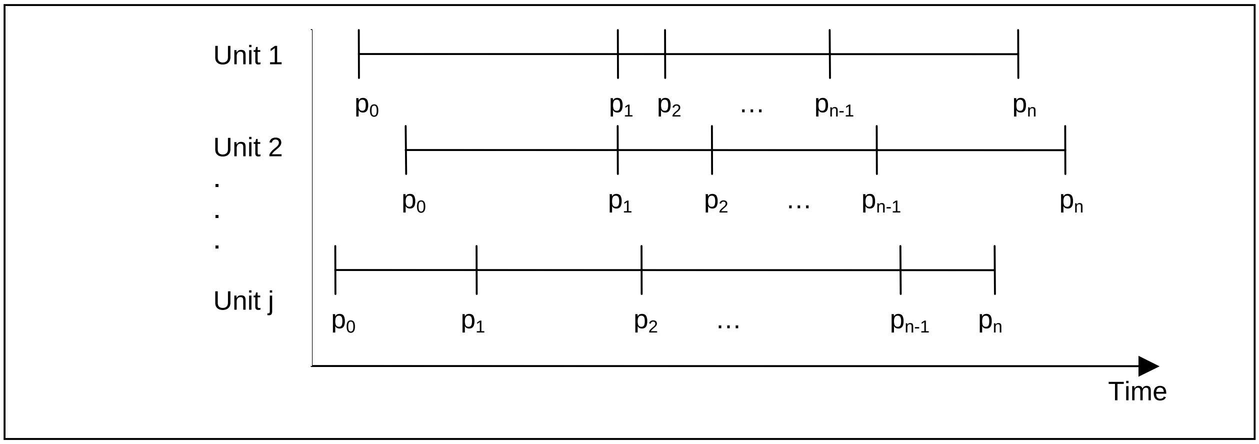
This formulation has the advantage that the event times T_k are not restricted to discrete durations, but can potentially be allocated anywhere in the continuous time horizon. However, for large problems, the constraint that all events must use the same global start and finish times, places a heavy burden on the model solution time. This burden results in very long solution times, not appropriate for operational use (refer to section 2.7.3).

Another major disadvantage of the proposed formulation is the large number of binary variables that results from the four indexed binary variable $X_{ijkk'}$ (i x j x k x k'). The overall size of the model is greatly increased by this binary variable.

2.3.3. Unit-specific events

Unit-specific events refer to an approach where different tasks in different units can refer to one event point *p*, and yet take place at different times in the continuous time horizon. Figure 2.5 illustrates this principle (Floudas and Lin, 2004):

Figure 2.5: Unit-specific approach



The main advantage of this approach is the scheduling flexibility provided by the unit-specific time allocation. This approach does not restrict the time representation of the problem, and an optimal solution can be guaranteed. Ierapetritou and Floudas (1998a,b) introduced this concept of time representation, which leads to a smaller amount of event points than the global event approach. Therefore, the overall size of a model is reduced.

Another contribution made by Ierapetritou and Floudas (1998a,b) was the introduction of a new concept for the use of binary variables. Previously researchers used one binary variable X_{ijp}, indicating task *i* to be done in unit *j* at point *p*. The new idea was to separate the tasks from the units. They used two binary variables: W_{ip}, indicating task *i* to be done at point *p* and Y_{jp} indicating unit *j* to be used at point *p*. The two variables are connected by the following allocation constraint:

$$\sum_{i \in I_j} W_{ip} = Y_{jp} \qquad \forall j, \quad \forall p \tag{2.3.1}$$

This constraint ensures that, in the event of a unit being activated for use ($Y_{jp} = 1$), at least one task *i* of the tasks that can be performed in unit *j* (subset I_j) should also be activated ($W_{ip} = 1$).

This concept reduces the number of binary variables from (i x j x p) to [(i x p) +(j x p)], which greatly improves the solution time of a model.

Floudas and Lin (2004) note that, in the case where a task can be performed in more than one unit, the task should be split into different tasks, one specifically allocated to each unit. This action will increase the number of W_{ip} binary variables. In the case where every task i can be performed by every unit j, the number of split tasks will equal the original tasks times the number of units (i x j).

It is therefore clear that the advantage of using the binary variable philosophy proposed by Ierapetritou and Floudas (1998a,b) will only realize in the instance where specific tasks i are allocated to specific units j. In the case where no such allocation is made and all tasks i can be performed by all units j, the number of binary variables will be the same as using the original binary variable X_{ijp}.

2.4. HARDWARE AND MODELLING SOFTWARE

2.4.1. Mathematical modelling

The General Algebraic Modelling System (GAMS 21.1) software from GAMS Development Corporation is used as the modelling interface. GAMS is a software package specifically designed for modelling linear, non-linear and mixed integer optimisation problems. The system is especially useful with large, complex problems (Majozi, 2003).

The developers state that GAMS was developed to do the following (Brooke et al, 1998):

- To provide a high-level language for the compact representation of large and complex models.
- To allow changes to be made in model specifications simply and safely.
- To allow unambiguous statements of algebraic relationships.
- To permit model descriptions which are independent of solution algorithms.

The detail GAMS listings for the models described in this chapter are included in Appendix A.

2.4.2. Solving the model

GAMS uses a variety of solvers to solve different types of models. The solvers of interest for this document are as follows:

a. DICOPT for MINLP (Grossmann et al., 2002)

DICOPT (DIscrete and Continuous OPTimizer) is an optimisation program which is used by GAMS to solve mixed integer non-linear programming (MINLP) problems. The MINLP algorithm inside DICOPT solves a series of non-linear programming (NLP) and mixed integer linear programming (MILP) sub-problems, resulting in various solution cycles. The solvers used by DICOPT to get the results in this document, are CONOPT as the NLP solver and CPLEX as the MILP solver.

b. CONOPT for NLP

CONOPT is a solver suited for non-linear constraints. It uses mathematical algorithms to find a local optimum solution to the NLP problem.

c. CPLEX for LP/MILP

CPLEX is used to solve linear programming (LP) as well as mixed integer linear programming (MILP) problems.

2.4.3. Hardware

The computer used to get the results in this document has a 2.0GHz CPU and 768MB RAM.

2.5. REPORTING MODEL RESULTS

For the purposes of this document, the reporting criteria used by Floudas and Lin (2004) will be used to report the results of any model discussed. These criteria are listed in Table 2.1.

2.6. EXAMPLE FOR TECHNIQUE COMPARISON

A smaller version of the problem described in Chapter 1 is used as an example problem. The application and detail mathematical formulation for each of the techniques introduced in section 2.2 and 2.3 are illustrated and discussed. Different linearization techniques are also highlighted.

Table 2.1: Reporting criteria for model results

Reporting criteria	Description
Number of points p	The number of entities in the set of points p.
Binary variables	The number of binary variables used in the model.
Continuous variables	The number of continuous variables used in the model.
Constraints	The total number of constraints in the model.
Objective value$_{max}$	The maximum value of the objective function in the solved state of the model.
CPU time	The length of the model solution time (sec).
Optcr	This is a GAMS function that sets a termination tolerance for Mixed Integer Linear Programming (MILP) problems. The objective function is compared to the best possible solution (in relaxed MILP mode). When the objective function is within the tolerance set by Optcr, the solver will terminate and report that specific solution.
DICOPT Cycles	As explained in section 2.4.2, DICOPT solves a series of MILP and NLP sub-problems. This setting prescribes how many times DICOPT should iterate between the NLP and MILP problems.

The results from the different scheduling technique models are compared and evaluated. The purpose of the comparison is to illustrate the effect of the different time representations on the solution time and the accuracy of the model. From the results of this example problem, a technique is chosen to apply to the SCS scheduling problem described in Chapter 1.

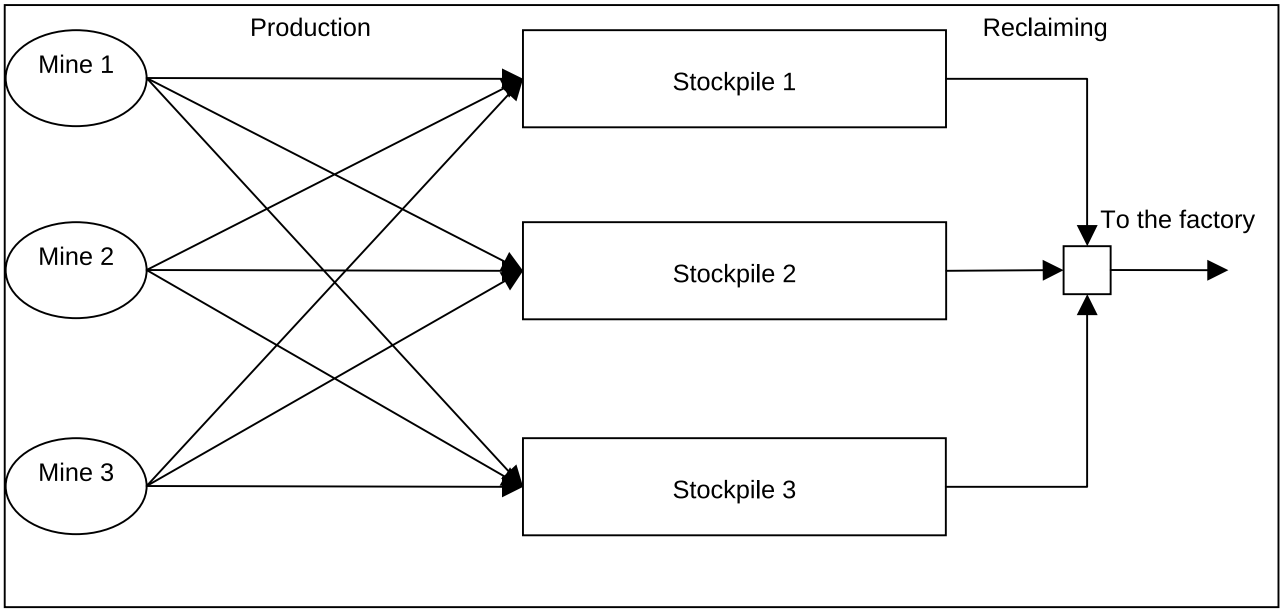
The strategy is to formulate a base model with binary variables, material balances, infrastructure constraints, demand constraints and an objective function. The base model remains unchanged for all the comparison models, thus the binary variable formulation also remains unchanged for all the models. This approach simplifies the comparison effort and highlights the specific performance contribution of each time representation technique.

For each of the different time representation formulations, specific equations are highlighted, focusing on timing restrictions. It includes time, duration, sequencing and quantity constraints. The definition of the set of time points or slots p will also change for each comparison model.

2.6.1. Problem description

The basic coal flow diagram for the example problem is illustrated by Figure 2.6:

Figure 2.6: Example problem



In the example problem there are three mines producing coal. From the mines, the coal is conveyed to a stockpiling area. The coal from any of the mines can be conveyed to any of the stockpiles. From the stockpiles, the coal is reclaimed, and the different streams are combined to supply coal to the factory.

The following infrastructure restrictions are applicable:

- A mine can only supply coal to one stockpile at a time.
- A stockpile can only receive coal from one mine at a time.
- The amount of coal allowed on a stockpile is restricted to its maximum capacity.
- Stacking coal on a stockpile and reclaiming coal from the stockpile may not happen simultaneously.
- When reclaiming coal, an equal portion of each mine's coal on the specific stockpile is reclaimed. Thus, when reclaiming 20% of the total stockpile, 20% of mine 1's coal, 20% of mine 2's coal and 20% of mine 3's coal on the specific stockpile is reclaimed (refer to section 1.5.1 for a detailed explanation).

The following simplifying assumptions are applicable:

- The mines' production are inexhaustible. Thus, there will always be coal available.
- No mine bunkers are taken into account.
- A stockpile consists of only one large stockpile, not four individual stockpiling areas as described in Chapter 1.
- A stockpile does not have to be stacked to capacity before reclaiming can start.

- A stockpile does not have to be finished with reclaiming before stacking can start.
- The factory has an unlimited capacity to receive coal.

The following cost assumptions were made:

- Income is generated by every ton of coal delivered to the factory.
- Cost is incurred when coal is conveyed from the different mines to the stockpiles.
- No other costs were taken into account.

2.6.2. Base model

The base model will be explained by stating the relevant sets, variables, constants and the detail mathematical formulation. Since the formulation for the different time representation techniques were not explained in sections 2.2 and 2.3, this description will clarify the application of the binary variables in the formulation. The different comparison models will focus only on the time representation constraints.

a. Sets

i = set of mines $i \in \{ mine_1, mine_2, mine_3\}$

j = set of stockpiles $j \in \{ sp_1, sp_2, sp_3\}$

p = set of event points / time-slots $p \in \{ p_1, p_2, p_3, \ldots, p_n\}$

b. Variables

b.1. Binary Variables:

w_{ijp} = 1 when conveying coal from mine i to stockpile j at p
= 0 otherwise

x_{jp} = 1 when reclaiming coal from stockpile j at p
= 0 otherwise

b.2. Positive variables:

q_b_{ijp} Amount of coal conveyed from mine i to stockpile j at p (kt)

q_r_{ijp} Amount of coal from mine i reclaimed from stockpile j at p (kt)

ST_s_{ijp} Amount of coal from mine i on stockpile j at p (kt)

Note that the quantity variables are given in ‘kilo-ton’. This ensures that the values in the model are scaled.

Ts_b_{ijp}	Time value if conveying coal from mine i to stockpile j *starts* at p (h)
Tf_b_{ijp}	Time value if conveying coal from mine i to stockpile j *stops* at p (h)
Dur_b_{ijp}	Duration of conveying coal from mine i to stockpile j if it starts at p (h)
Ts_r_{jkp}	Time value if reclaiming coal from stockpile j *starts* at p (h)
Tf_r_{jkp}	Time value if reclaiming coal from stockpile j *stops* at p (h)
Dur_r_{jkp}	Duration of reclaiming coal from stockpile j if it starts at p (h)
T_p	Time value of a global event point at p (h)

b.3. Variable:

z_{max}	Objective function variable to maximise profit

c. Parameters:

In the context of this document, a parameter is a constant value that has an index. Parameters with two or more indices result in tables.

$ST0\_s_{ij}$	Starting level of mine i contribution to stockpile j (kt)
Cap_s_j	Capacity of stockpile j (kt)
$Rate\_b_i$	Rate for conveying coal from mine i (kt/h)
$Rate\_r_j$	Rate for reclaiming coal from stockpile j (kt/h)
$Cost_i$	Cost of conveying coal from mine i (R/kt)
Delta	The size of an interval in the discrete time representation (*Delta* = *H* / number of points *p*)

d. Constants:

In the context of this document, a constant is a value that remains the same through out the scheduling model's time horizon, but does not have an index. In the GAMS context these constant values are called scalars.

H	Scheduling time horizon (chosen to be 12 hours)

Demand Coal demand of factory during time horizon

Income Income per kt reclaimed (R/kt)

e. Mathematical formulation:

e.1. Allocation constraints and binary linearization:

Coal conveyed from mine i to stockpile j is indicated with the binary variable w_{ijp}. The following equation enforces the infrastructure restriction that a mine's coal may only be conveyed to one stockpile:

$$\sum_j w_{ijp} \leq 1 \qquad \forall i, \quad \forall p \qquad (2.6.1)$$

The inverse of the restriction, namely that only one mine at a time may be stacked at a stockpile j, is enforced by the following equation:

$$\sum_i w_{ijp} \leq 1 \qquad \forall j, \quad \forall p \qquad (2.6.2)$$

The equation above forces only one conveying route from the mines to a specific stockpile j to be activated.

To ensure that stacking and reclaiming does not happen simultaneously, the following constraint is needed:

$$(\sum_i w_{ijp}) \times x_{jp} = 0 \qquad \forall j, \quad \forall p \qquad (2.6.3.a)$$

This equation ensures that one of the binary terms will always be equal to 0, thus ensuring that stacking and reclaiming will never happen simultaneously. However, this equation is non-linear. Non-linear equations should be linearized as far as possible, to enhance the model structure and solution time. The linearization of equation 2.6.3.a results in the following equation:

$$(\sum_i w_{ijp}) + x_{jp} \leq 1 \qquad \forall j, \quad \forall p \qquad (2.6.3.b)$$

The two binary terms are not multiplied anymore, but added and set less than or equal to one. This formulation also ensures that only one of the terms will be activated (set equal to one), thus reclaiming and stacking will not happen simultaneously.

This linearization technique will also be used in the model development and improvement phases described in Chapters 3 and 4.

e.2. Storage constraints:

The amount of coal from each mine on the different stockpiles at the start of the scheduling horizon is given by the user:

$$ST\_s_{ijkp} = ST0\_s_{ijk} \qquad \forall i, \quad \forall j, \quad p = p_1 \tag{2.6.4}$$

The material balance for the amount of coal from a specific mine i on a specific stockpile j can be stated as follows:

$$\begin{aligned} ST\_s_{ijp} &= ST\_s_{i,j,p-1} - q\_r_{i,j,p-1} + q\_s_{i,j,p-1} \\ &\forall i, \quad \forall j, \quad \forall p > p_1 \end{aligned} \tag{2.6.5}$$

Equation 2.6.5 starts with the amount of coal from mine i on a specific stockpile j at the previous point (p-1). The amount of that specific mine's coal reclaimed from the stockpile since the previous point (p-1) is subtracted. Lastly, the amount of coal from that specific mine stacked on the specific stockpile j since the previous point (p-1) is added.

Note that, since no previous point (p-1) exists at the first point, equation 2.6.5 does not hold for the first point p_1.

The capacity limit per stockpile serves as the upper limit for the total amount of coal stacked on stockpile j:

$$\sum_i ST\_s_{ijp} \leq Cap\_s \qquad \forall j, \quad \forall p \tag{2.6.6}$$

The upper limit for the amount of coal from a specific mine i which may be reclaimed, is the amount of coal from a specific mine i stacked on a stockpile j:

$$q\_r_{ijp} \leq ST\_s_{ijp} \qquad \forall i, \quad \forall j, \quad \forall p \tag{2.6.7}$$

To ensure that the same portion of each mine's coal on the stockpile is reclaimed, the following non-linear equation was used:

$$\frac{q\_r_{ijp}}{ST\_s_{ijp}} = \frac{q\_r_{ii,j,p}}{ST\_s_{ii,j,p}} \qquad \forall i \neq ii, \quad \forall j, \quad \forall p \tag{2.6.8}$$

This equation enforces the fifth infrastructure restriction described in section 2.6.1.

No linearization technique was applied to equation 2.6.8. Therefore, equation 2.6.8 is the *only non-linear equation* in the base model's formulation described above. This resulted in the final models being mixed integer non-linear programming (MINLP) models.

e.3. Demand constraint:

The total amount of coal reclaimed from the stockpiles must satisfy the coal demand from the factory:

$$\sum_{ijp} q\_r_{ijp} \geq Demand \tag{2.6.9}$$

e.4. Objective function:

The objective of the example problem is to maximise profit, thus maximising the income from coal supplied to the factory and minimising the cost of conveying coal to the stockpiles:

$$z_{max} = \sum_{ijp} \left(q\_r_{ijp} \times Income \right) - \sum_{ijp} \left(q\_b_{ijp} \times Cost_i \right) \tag{2.6.10}$$

2.6.3. Discrete model

In the discrete model, the time horizon is divided into a number of time intervals of equal duration (Δt = *Delta*), as illustrated in Figure 2.1. Thus, the set of points *p* represents the discretisation of the time horizon.

The only constraints that need to be added to the base model in section 2.6.2, are the two quantity constraints for stacking and reclaiming.

The amount from a specific mine i stacked on a specific stockpile j depends on the activation of the binary variable:

$$q\_b_{ijp} = rate\_b_i \times Delta \times w_{ijp} \qquad \forall i, \quad \forall j, \quad \forall p \tag{2.6.11}$$

Note that *Delta* represents the size of the discrete intervals. Therefore, if w_{ijp} is activated, the conveying rate (kt/hr) is multiplied by the applicable time interval (hr) to get the amount of coal (kt) conveyed from mine i to stockpile j in the particular discrete interval p. This is not a non-linear equation, since both the rate and the *Delta* variable is constant.

Similarly, the amount of coal reclaimed from a stockpile j is represented by:

$$\sum_{i} q\_r_{ijp} = rate\_r_j \times Delta \times x_{jp} \qquad \forall j, \quad \forall p \tag{2.6.12}$$

Since the discrete variable x_{jp} is activated for the stockpile in total, and not for the coal of one specific mine, the calculation in equation 2.6.12 is made for the total amount of coal reclaimed from that specific stockpile for a specific discrete interval p. Equation 2.6.8 will ensure that the correct amount of each mine's coal on the stockpile is reclaimed.

2.6.4. Time-slot model

In the time-slot model, the set of points *p* represents the time-slots that can be placed anywhere in the time horizon (refer to Figure 2.3). A slot corresponds to a single event in a certain unit.

Because the slots can be placed anywhere in the continuous time domain, the mathematical formulation includes constraints to ensure the correct sequencing of events. The result is a more complex model structure than the discrete model.

a. Stacking events

a.1. Duration and quantity constraints

The time balance of an event is represented as follows:

$$Tf\_b_{ijp} = Ts\_b_{ijp} + Dur\_b_{ijp} \qquad \forall i, \quad \forall j, \quad \forall p \qquad (2.6.13)$$

Equation 2.6.13 calculates the finish time value of a specific slot p, given the associated starting time value of slot p, plus the duration of the event. This will cause the starting and finishing times to be equal when the duration of an event is 0, that is, the event did not take place. Note that the start time, the duration and the finish time are all allocated to one specific slot p.

The following equation links the event duration to the activation of the binary variable w_{ijp}:

$$Dur\_b_{ijp} \quad \leq \quad H \times w_{ijp} \qquad \forall i, \quad \forall j, \quad \forall p \qquad (2.6.14)$$

The constraint above ensures that the duration of an event is less than the scheduling time horizon H, if the binary variable is activated. Thus, in the case where w_{ijp} = 1, equation 2.6.14 sets an upper limit for the duration of a stacking event. Otherwise, when w_{ijp} = 0, the duration is forced to be 0, since Dur_b_{ijp} is a positive variable.

To calculate the amount of coal conveyed from a mine to a stockpile, the following equation holds:

$$q\_b_{ijp} \quad = \quad rate\_b_i \times Dur\_b_{ijp} \qquad \forall i, \quad \forall j, \quad \forall p \qquad (2.6.15)$$

Note that the fixed-length *Delta* parameter used in the discrete model (equation 2.6.11), is replaced with the duration variable in equation 2.6.15. This gives more flexibility to the model and better time representation.

a.2. Sequencing constraints and binary/continuous linearization:

As explained earlier, the flexible time representation of the slot based model, results in the following additional sequencing constraints.

To ensure the sequence of consecutive slots when the same mine's coal is conveyed to different stockpiles (j and jj):

$$Ts\_b_{ijp} \geq Tf\_b_{i,jj,p-1} \times w_{ijp} \qquad \forall i, \quad \forall j, jj, \quad \forall p > p_1 \tag{2.6.16.a}$$

Equation 2.6.16.a ensures that the start time value of slot p is after the finish time value of the previous slot (p-1), if the binary variable for slot p is indeed activated. Note that this constraint is non-linear as a result of the product of the binary variable w_{ijp} and the continuous variable $Tf\_b_{i,jj,p-1}$.

With the linearization technique described by Glover (1975), the non-linear constraint of 2.6.16.a was transformed to the following linear constraint:

$$Ts\_b_{ijp} \geq Tf\_b_{i,jj,p-1} - H(1 - w_{ijp}) \qquad \forall i, \quad \forall j, jj, \quad \forall p > p_1 \tag{2.6.16.b}$$

The term $H(1 - w_{ijp})$ acts as a relaxation term. If $w_{ijp} = 1$, the term becomes 0, therefore enforcing the constraint. If $w_{ijp} = 0$, the term takes on a large value, the equation is relaxed, and trivially solved.

Similarly, to ensure the sequence of consecutive slots when the different mines' (i and ii) coal is conveyed to the same stockpile:

$$Ts\_b_{ijp} \geq Tf\_b_{ii,j,p-1} - H(1 - w_{ijp}) \qquad \forall i, ii, \quad \forall j, \quad \forall p > p_1 \tag{2.6.17}$$

This linearization technique will be used in the rest of the example problem's formulation as well as in the model development and improvement phases described in Chapters 3 and 4.

The following equations ensure that the start and finish times of consecutive time-slots are in sequence:

$$Ts\_b_{ijp} \geq Ts\_b_{i,j,p-1} \qquad \forall i, \quad \forall j, \quad \forall p > p_1 \tag{2.6.18}$$

$$Tf\_b_{ijp} \geq Tf\_b_{i,j,p-1} \qquad \forall i, \quad \forall j, \quad \forall p > p_1 \tag{2.6.19}$$

Finally, the finish time of any time-slot should not exceed the length of the time horizon:

$$Tf\_b_{ijp} \leq H \qquad \forall i, \quad \forall j, \quad \forall p \tag{2.6.20}$$

b. Reclaiming events

b.1. Duration and quantity constraints

Similar to the equations described above, the timing constraints for the reclaiming action from a certain stockpile to the factory is formulated as follows:

$$Tf\_r_{jp} = Ts\_b_{jp} + Dur\_b_{jp} \qquad \forall j, \quad \forall p \tag{2.6.21}$$

$$Dur\_r_{jp} \leq H \times x_{jp} \qquad \forall j, \quad \forall p \tag{2.6.22}$$

$$\sum_i q\_r_{ijp} = rate\_r_j \times Dur\_r_{jp} \qquad \forall j, \quad \forall p \tag{2.6.23}$$

Note that the total amount of coal reclaimed from stockpile j is calculated in equation 2.6.23, resulting in the summation of all the mines' coal reclaimed from that specific stockpile (refer to equation 2.6.12).

b.2. Sequencing constraints

The sequencing constraints for the reclaiming action at a stockpile also follow the same formulation described in section a.2 above:

$$Ts\_r_{jp} \geq Tf\_r_{j,p-1} - H(1 - x_{jp}) \qquad \forall j, \quad \forall p > p_1 \tag{2.6.24}$$

$$Ts\_r_{jp} \geq Ts\_r_{j,p-1} \qquad \forall j, \quad \forall p > p_1 \tag{2.6.25}$$

$$Tf\_r_{jp} \geq Tf\_r_{j,p-1} \qquad \forall j, \quad \forall p > p_1 \tag{2.6.26}$$

$$Tf\_r_{jp} \leq H \qquad \forall j, \quad \forall p \tag{2.6.27}$$

c. Sequencing between stacking and reclaiming

To ensure that reclaiming starts only after the stacking ended, the following sequencing constraint is necessary:

$$Ts\_r_{jp} \geq Tf\_b_{i,j,p-1} - H(1 - x_{jp}) \qquad \forall i, \quad \forall j, \quad \forall p > p_1 \tag{2.6.28}$$

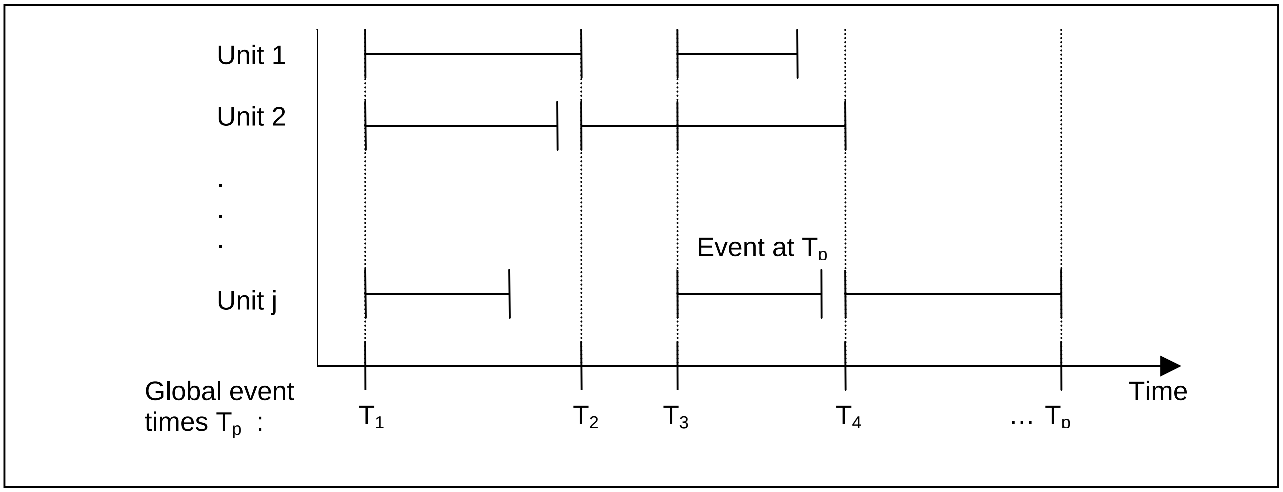
The constraint above ensures that the reclaiming start time value of the time-slot p is not before the stacking end time value of the previous time-slot (p-1). This equation only holds if the reclaiming binary variable x_{jp} is activated, otherwise the constraint is relaxed.

2.6.5. Global event based model

In the global event based approach discussed in section 2.3.2, the global events entail the use of a set of event points applying to *all* tasks and *all* units. These event points indicate the start *as well as* the end of any event taking place in the time horizon (refer to Figure 2.4).

An alternative global event based approach is to let the event points only indicate the starting time value of the events (not the finish times as well), but allowing the duration to be any length between the different starting points. This philosophy is illustrated by Figure 2.7:

Figure 2.7: Global starting time approach per unit



This alternative approach may have the advantage of slightly more flexible duration times which is not restrained by the placement of global time points.

Both these global event approaches (global starting and finishing points and global starting points only) will be illustrated by the example model.

The following characteristics hold for the global event based representation:

- The set of points *p* represents the global event points that can be placed anywhere in the time horizon (refer to Figure 2.4 and Figure 2.7).
- The global time variable T_p is used, instead of the individual timing variables such as Ts_b_{ijp}.
- Events only start or finish at an event point, contrary to the slot based approach where both the start and the finish of an event is represented by one event point.

a. Global starting and finishing points

a.1. Allocation constraints

When an event is allocated to point p, the finish time value is allocated to the next point (p+1). As a result, the following additional allocation constraints are necessary for the last point p_n:

$$w_{ijp} = 0 \qquad \forall i, \quad \forall j, \quad \forall p = p_n \qquad (2.6.29)$$

$$x_{jp} = 0 \qquad \forall j, \quad \forall p = p_n \qquad (2.6.30)$$

The constraints above set both binary variables to 0 at the last time point. This ensures that no event starts on the last time point. If these constraints are excluded, events will be allocated to the last time point, and there will be no event point for the finish times of such events, leaving such events open-ended.

a.2. Timing, duration and quantity constraints and linearization extension

To calculate the global event points T_p, the following sets of constraints are used:

$$\begin{aligned} &T_p \geq T_{p-1} + Dur\_b_{ijp} - H(1 - w_{ijp}) \\ &T_p \leq T_{p-1} + Dur\_b_{ijp} + H(1 - w_{ijp}) \\ &\forall i, \quad \forall j, \quad \forall p > p_1 \end{aligned} \tag{2.6.31}$$

This set of constraints is an extension of the linearization explained in section 2.6.4. It ensures that the global time point T_p is equal to the previous time point T_{p-1} plus the duration of the stacking event. This only holds if the stacking binary variable is activated, thus resulting in the relaxation term $H(1 - w_{ijp})$ being equal to 0. In this case, the 'less than' and 'greater than' signs in the two equations have the effect of an 'equal to' sign. If $w_{ijp} = 0$, the set of equations is again relaxed and trivially solved.

Similarly, the following set of equations holds for the reclaiming events:

$$\begin{aligned} &T_p \geq T_{p-1} + Dur\_r_{jp} - H(1 - x_{jp}) \\ &T_p \leq T_{p-1} + Dur\_r_{jp} + H(1 - x_{jp}) \\ &\forall j, \quad \forall p > p_1 \end{aligned} \tag{2.6.32}$$

Similar to the time-slot model, the duration and quantity constraints for the stacking events are defined as follows (refer to equations 2.6.14 and 2.6.15):

$$Dur\_b_{ijp} \leq H \times w_{ijp} \qquad \forall i, \quad \forall j, \quad \forall p \tag{2.6.33}$$

$$q\_b_{ijp} = Dur\_b_{ijp} \times rate\_b_i \qquad \forall i, \quad \forall j, \quad \forall p \tag{2.6.34}$$

The constraints for the reclaiming events are formulated in the same way:

$$Dur\_r_{jp} \leq H \times x_{jp} \qquad \forall j, \quad \forall p \tag{2.6.35}$$

$$q\_r_{jp} = Dur\_r_{jp} \times rate\_r_j \qquad \forall j, \quad \forall p \tag{2.6.36}$$

To ensure the time points do not exceed the time horizon, the following constraint is necessary (similar to equation 2.6.20):

$$T_p \leq H \qquad \forall p \tag{2.6.37}$$

Finally, to ensure the sequence of the global time points, the following equation holds:

$$T_p \geq T_{p\text{-}1} \qquad \forall p > p_1 \tag{2.6.38}$$

b. Global starting points only

With this approach, the allocation constraints 2.6.29 and 2.6.30 are replaced by the following timing constraints applicable to the last point p_n:

$$T_p + Dur\_b_{ijp} \leq H \qquad \forall i, \quad \forall j, \quad \forall p = p_n \tag{2.6.39}$$

$$T_p + Dur\_r_{jp} \leq H \qquad \forall j, \quad \forall p = p_n \tag{2.6.40}$$

These constraints ensure that the stacking and reclaiming events at the last time point do not exceed the time horizon.

The other change in this alternative approach is the elimination of the second constraint in each of the two sets of equations 2.6.31 and 2.6.32. This ensures that the next time point can be placed anywhere in the time horizon after the duration of the current event. This is an important relaxation of the previous global event formulation.

Equations 2.6.33 to 2.6.38 also hold for this alternative global based approach.

2.6.6. Unit-specific event based model

The formulation of the unit-specific event based approach is very similar to that of the time-slot approach. Both require various additional equations to ensure sequenced events. The only difference is that events in the unit-specific approach start at an event point p and finish at the next event point p. The slot based approach uses one slot p to indicate both the start and the end of an event.

a. Allocation constraints

As with the first global event approach, the following allocation restrictions hold for the last point p_n (refer to equation 2.6.29 and 2.6.30):

$$w_{ijp} = 0 \qquad \forall i, \quad \forall j, \quad \forall p = p_n \tag{2.6.41}$$

$$x_{jp} = 0 \qquad \forall j, \quad \forall p = p_n \tag{2.6.42}$$

b. Stacking events

b.1. Duration and quantity constraints

Similar to equation 2.6.13, the following equation holds:

$$Tf\_b_{ijp} = Ts\_b_{i,j,p\text{-}1} + Dur\_b_{i,j,p\text{-}1} \qquad \forall i, \quad \forall j, \quad \forall p > p_1 \qquad (2.6.43)$$

Note that the event finish time is allocated at the next time point p, if the event started at point p-1. As a result of this formulation, equation 2.6.43 does not hold for p_1.

Similar to equations 2.6.14 and 2.6.15, the following duration and quantity constraints hold:

$$Dur\_b_{ijp} \leq H \times w_{ijp} \qquad \forall i, \quad \forall j, \quad \forall p \qquad (2.6.44)$$

$$q\_b_{ijp} = rate\_b_i \times Dur\_b_{ijp} \qquad \forall i, \quad \forall j, \quad \forall p \qquad (2.6.45)$$

b.2. Sequencing constraints

The same event based timing principle holds for the following constraints (refer to 2.6.16 to 2.6.17 for explanations):

$$Ts\_b_{ijp} \geq Tf\_b_{i,jj,p} - H(1 - w_{ijp}) \qquad \forall i, \quad \forall j, jj, \quad \forall p \qquad (2.6.46)$$

$$Ts\_b_{ijp} \geq Tf\_b_{ii,j,p} - H(1 - w_{ijp}) \qquad \forall i, ii, \quad \forall j, \quad \forall p \qquad (2.6.47)$$

Note that the start of the next event and the end of the previous event will take place at the same time point p.

Similar to equations 2.6.18 to 2.6.20, the following general sequencing constraints hold:

$$Ts\_b_{ijp} \geq Ts\_b_{i,j,p\text{-}1} \qquad \forall i, \quad \forall j, \quad \forall p > p_1 \qquad (2.6.48)$$

$$Tf\_b_{ijp} \geq Tf\_b_{i,j,p\text{-}1} \qquad \forall i, \quad \forall j, \quad \forall p > p_1 \qquad (2.6.49)$$

$$Tf\_b_{ijp} \leq H \qquad \forall i, \quad \forall j, \quad \forall p \qquad (2.6.50)$$

b. Reclaiming events

b.1. Duration and quantity constraints

The following timing constraint for the reclaiming action from a certain stockpile to the factory is formulated similar to the stacking action's time balance (equation 2.6.43):

$$Tf\_r_{jp} = Ts\_b_{j,p\text{-}1} + Dur\_b_{j,p\text{-}1} \qquad \forall j, \quad \forall p > p_1 \qquad (2.6.51)$$

Similar to Equations 2.6.22 and 2.6.23, the following duration and quantity constraints hold:

$$Dur\_r_{jp} \leq H \times x_{jp} \qquad \forall j, \quad \forall p \qquad (2.6.52)$$

$$\sum_i q\_r_{ijp} = rate\_r_j \times Dur\_r_{jp} \qquad \forall j, \quad \forall p \qquad (2.6.53)$$

b.2. Sequencing constraints

The reclaiming action's sequencing is formulated similar to the stacking action's sequence in equation 2.6.46:

$$Ts\_r_{jp} \geq Tf\_r_{jp} - H(1 - x_{jp}) \qquad \forall j, \quad \forall p \tag{2.6.54}$$

Similar to equations 2.6.25 to 2.6.27, the following constraints hold:

$$Ts\_r_{jp} \geq Ts\_r_{j,p-1} \qquad \forall j, \quad \forall p > p_1 \tag{2.6.55}$$

$$Tf\_r_{jp} \geq Tf\_r_{j,p-1} \qquad \forall j, \quad \forall p > p_1 \tag{2.6.56}$$

$$Tf\_r_{jp} \leq H \qquad \forall j, \quad \forall p \tag{2.6.57}$$

c. Sequencing between stacking and reclaiming

As stated above, equation 2.6.28 changes as follows to apply to this model:

$$Ts\_r_{jp} \geq Tf\_b_{ijp} - H(1 - x_{jp}) \qquad \forall i, \quad \forall j, \quad \forall p \tag{2.6.58}$$

The change ensures that the next reclaiming event does not start (at point p) before the end (also at point p) of the previous stacking event.

2.7. COMPARISON RESULTS

Each model was tested with a different number of points/slots p. It is important to note the increase in solution time (CPU time) and the improvement of the objective function value, as the number of points/slots p increased. The results will be reported in the format explained in section 2.5.

The results is summarised in section 2.7.5 by comparing the different model sizes (with equal number of points/slots p for each model), and the different model performances (solution time and objective function value at a certain optimum number of points/slots p).

2.7.1. Discrete model

The following results were achieved with the discrete model formulation as described in section 2.6.2 and 2.6.3.

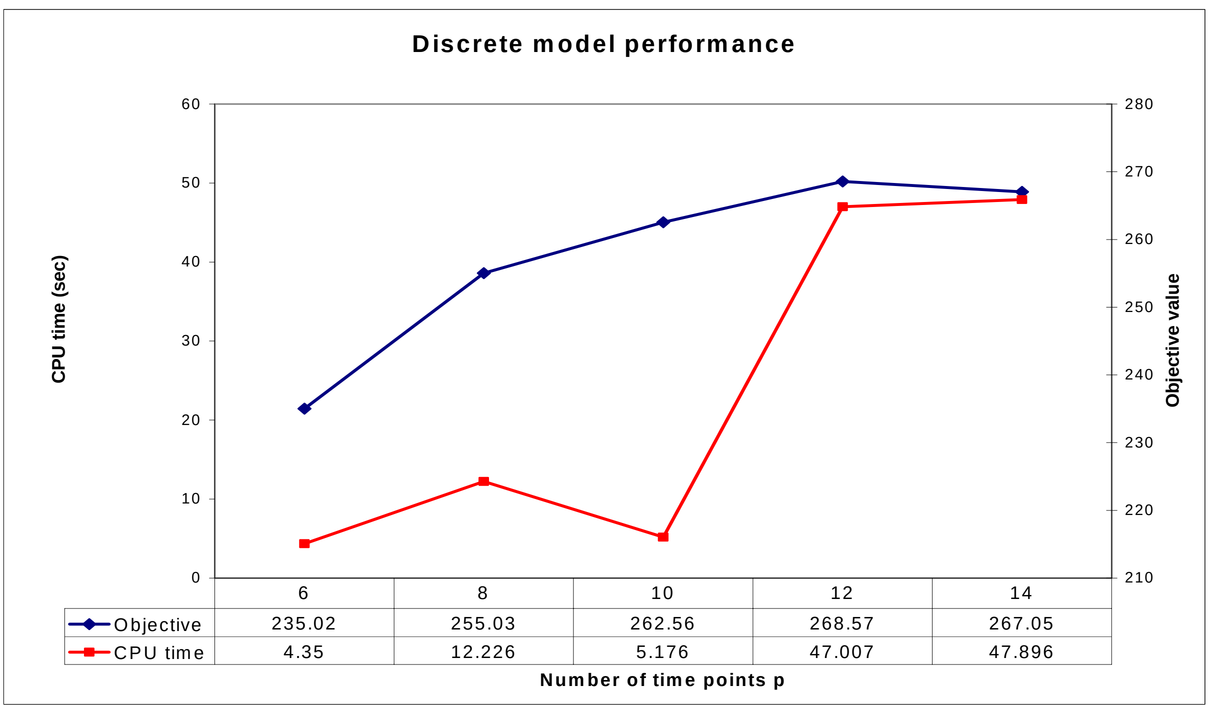
The graphical illustration in Graph 2.1 clearly shows the exponential increase in CPU time as the number of points p increases. However, the value of the objective function increases with the inverse of the CPU time. As a result of this characteristic, the optimum amount of time points results in a trade-off between CPU time and the objective value.

Table 2.2: Discrete model results

Discrete model results	**Run 1**	**Run 2**	**Run 3**	**Run 4**	**Run 5**
Number of points p	6	8	10	12	14
Binary variables	72	96	120	144	168
Continuous variables	235	313	391	469	547
Constraints	362	482	602	722	842
$Objective_{max}$ value	**235.02**	**255.03**	**262.56**	**268.57**	**267.05**
CPU time (sec)	**4.35**	**12.226**	**5.176**	**47.007**	**47.896**
Optcr	10.00%	10.00%	10.00%	10.00%	10.00%
DICOPT Cycles	3	3	3	3	3

Graph 2.1: Discrete model performance:

CPU time and Objective value as a function of the number of time points.



The significant increase in the CPU time with 12 points does not reflect the same increase in the objective function value. Thus, in this case, 10 points will give the best approximation of the objective value, with a fast CPU time.

2.7.2. Time-slot model

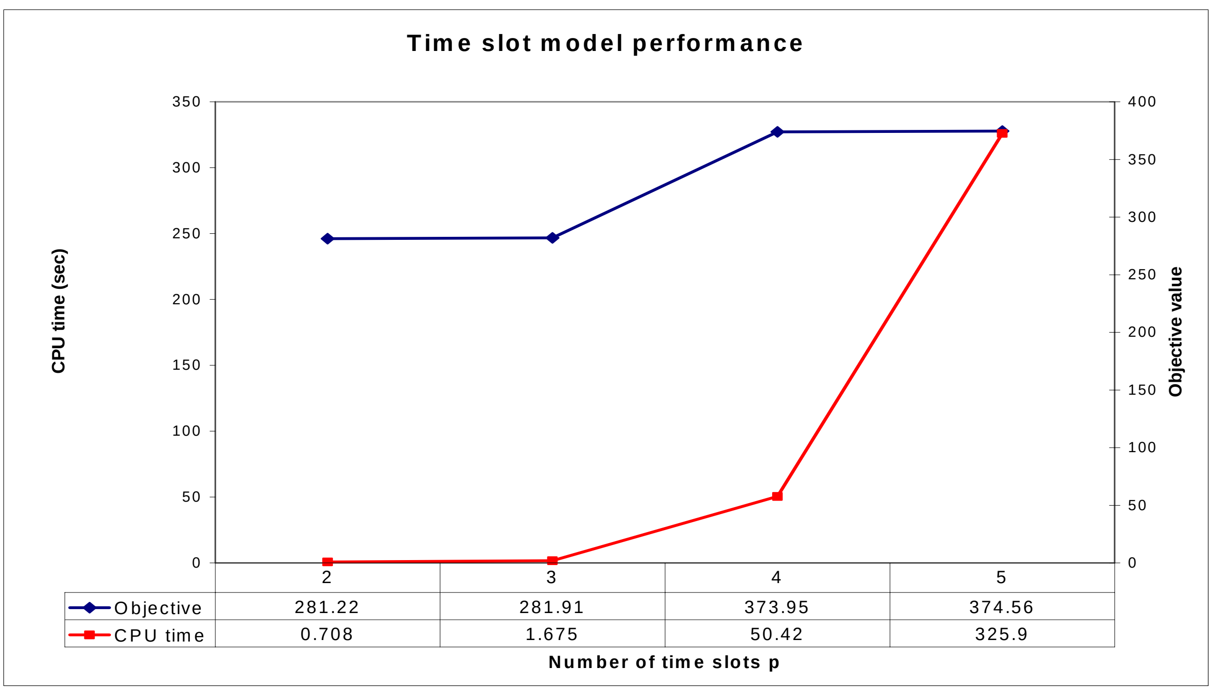
The following results were achieved with the time-slot model formulation as described in section 2.6.2 and 2.6.4.

Table 2.3: Time-slot model results

Time-slot model results	Run 1	Run 2	Run 3	Run 4
Number of slots p	2	3	4	5
Binary variables	24	36	48	60
Continuous variables	151	226	301	376
Constraints	284	470	656	842
$Objective_{max}$ value	**281.22**	**281.91**	**373.95**	**374.56**
CPU time (sec)	0.708	1.675	50.42	325.9
Optcr	10.00%	10.00%	10.00%	40.00%
DICOPT Cycles	3	3	3	3

Graph 2.2: Time-slot model performance:

CPU time and Objective value as a function of the number of time points.



The time-slot approach resulted in a much better object function value (with only four and five slots) than the discrete model. This is the result of the continuous time representation, instead of the discrete approximation of the time horizon. Note that, compared to the discrete model, less slots are needed to give a better representation of the problem.

The CPU time exponentially increases from four to five slots p, with no meaningful increase in the objective function value (refer to Graph 2.2). This dramatic increase in the CPU solution time is a direct result of the more complicated and flexible model structure as explained in section 2.6.4. In the case of six discrete points, the discrete model has 362 individual constraints, whereas the equivalent time-slot model with six slots has 1028 individual constraints.

Note that the Optcr setting (refer to section 2.5) was set to 40% for the model with five slots p. Even with the more relaxed solution criteria, the CPU time still increased dramatically.

As discussed in section 2.7.1 above, the optimal number of slots p results in a trade-off between CPU solving time and model accuracy. In this case, four points gives the best objective function approximation with the best associated CPU time.

2.7.3. Global based model

The following results were achieved with the global event based model formulation as described in section 2.6.2 and 2.6.5. The first model relates to global starting as well as finishing times. The second model relates to global starting times only.

Table 2.4: Global event based model results: Model 1

Global event based model results: Model 1	**Run 1**	**Run 2**	**Run 3**	**Run 4**
Number of points p	3	4	5	6
Binary variables	36	48	60	72
Continuous variables	157	209	360	313
Constraints	283	381	479	577
Objective$_{max}$ value	**253.78**	**276.06**	**285.34**	**286.68**
CPU time (sec)	0.974	1.731	62.417	1190.5
Optcr	10.00%	10.00%	10.00%	50.00%
DICOPT Cycles	3	3	3	3

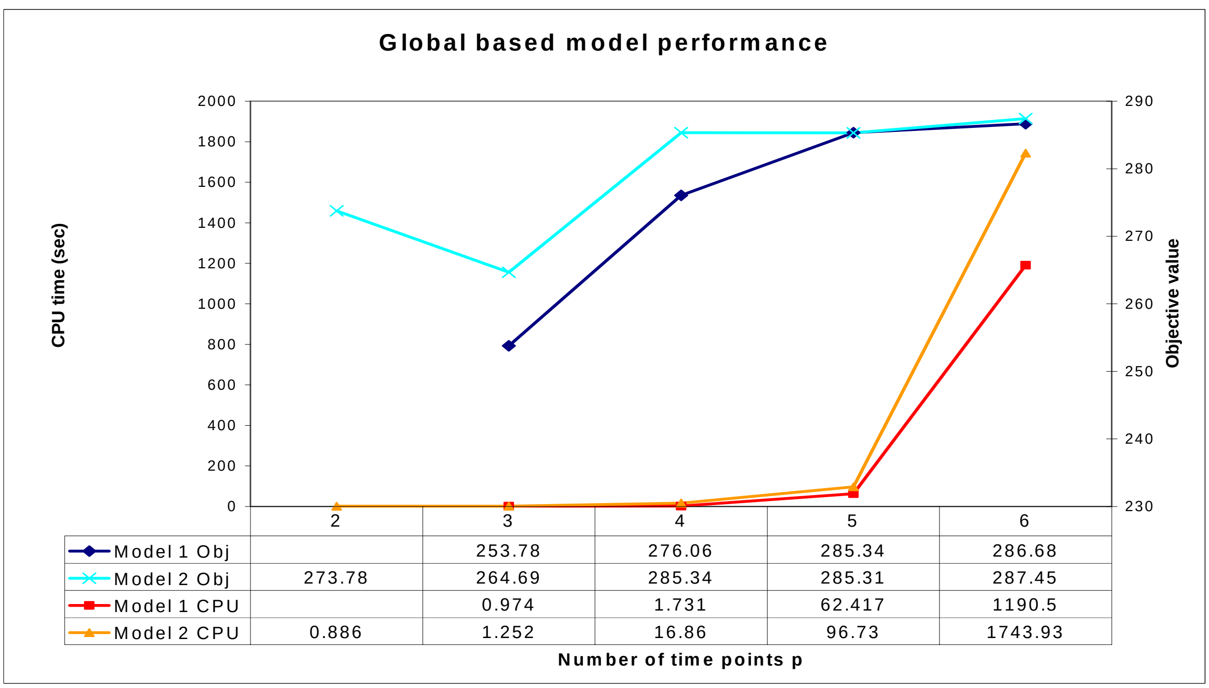
Note that the Optcr function was set to 50% for the run with six points. Even with the more relaxed solution criteria, the CPU time still increased dramatically.

Table 2.5: Global event based model results: Model 2

Global event based model results: Model 2	Run 1	Run 2	Run 3	Run 4	Run 5
Number of points p	2	3	4	5	6
Binary variables	24	36	48	60	72
Continuous variables	105	157	209	261	313
Constraints	173	259	345	431	517
Objective$_{max}$ value	**273.78**	**264.69**	**285.34**	**285.31**	**287.45**
CPU time (sec)	0.886	1.252	16.86	96.73	1743.93
Optcr	10.00%	10.00%	10.00%	10.00%	50.00%
DICOPT Cycles	3		3	3	3

Note that the Optcr function was set to 50% for the run with six points, and the CPU solution time still increased exponentially. Also note that model 2 could run with only 2 points because it does not need an additional point to allocate an event finish time at the end of the time horizon.

Graph 2.3: Global event based model 1 and 2 performance:
CPU time and Objective value as a function of the number of time points.



Initially, model 2 had better objective function results than model 1 (compare run 1, 2 and 3). However, with five and six points each, the two models had very similar results.

The objective function results from the global based models are substantially lower than the time-slot model's objective function values. This can be explained by the fact that the global events place a restriction on the time representation of the problem. Synchronised starting and finishing points for all events in all units cause, by definition, a sub-optimal answer (refer to section 2.3).

The objective function results from the global based models are a bit higher than the same results from the discrete model. It is the result of the continuous time representation which gives more flexibility to the model than in the discrete case.

It should also be noted that in both the global event based models very large exponential increases in CPU time were reported when the points p were increased from five to six time points. In contrast, the objective function value increased minimally.

The hypothesis (set in section 2.6.5) that the second global event based approach could result in better objective function values and faster solution times, was proofed wrong with the results above. Both approaches have very similar performances and no distinct difference can be made between the different models' objective function value. The second approach is therefore disregarded.

2.7.4. Unit-specific event based model

The following results were achieved with the global event based model formulation as described in section 2.6.2 and 2.6.6.

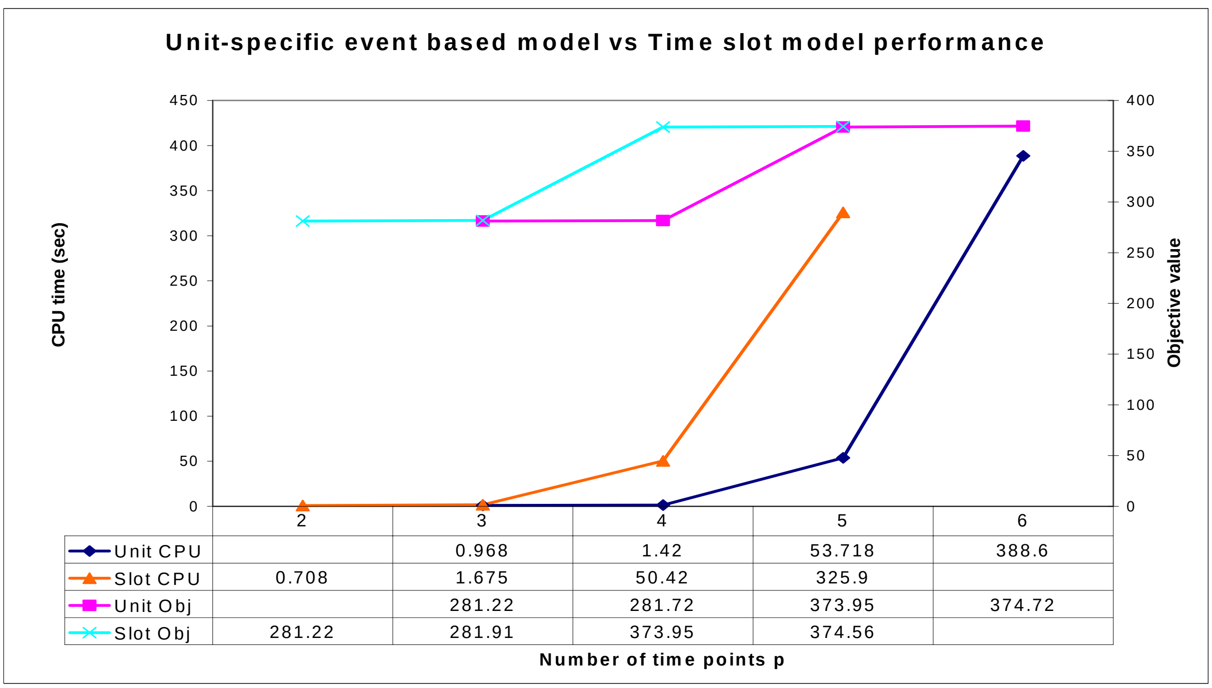
The objective function value of the unit-specific approach, and the results from the time-slot approach were the highest objective function values reported. This comes as no surprise, since both these techniques represent continuous time representation for all units and all tasks. These two models' results show the advantage of continuous time representation as apposed to the discretisation of the horizon and the use of global events that restrict the time representation in a model.

The performances of the unit-specific and the time-slot models were very similar. To illustrate this similarity in performance, the two models are plotted on the same graph in Graph 2.4. Note that the Optcr function was set to 40% for the run with six points. Even with the more relaxed solution criteria, the CPU time still increased exponentially.

Table 2.6: Unit-specific event based model results:

Unit-specific event based model results	**Run 1**	**Run 2**	**Run 3**	**Run 4**
Number of points p	3	4	5	6
Binary variables	36	48	60	72
Continuous variables	226	301	376	451
Constraints	527	713	899	1085
Objective$_{max}$ value	**281.22**	**281.72**	**373.95**	**374.72**
CPU time (sec)	0.968	1.42	53.718	388.6
Optcr	10.00%	10.00%	10.00%	40.00%
DICOPT Cycles	3	3	3	3

Graph 2.4: Unit-specific event based model performance vs Time-slot model performance: CPU time and Objective value as a function of the number of time points.



The unit-specific event based model with five time points will give a good result within a reasonable solution time, thus resulting in the optimum number of time points for this particular case.

From Graph 2.4, it can be noted that the two models' graphs follow the same pattern, with the unit-specific model one time point behind the slot based approach. This is the result of the unit-specific model having one additional time point at the end of the horizon to allocate the finish time of the last event. The time-slot model does not need that additional time point, since event start and finish times are allocated to one slot.

2.7.5. Summary of results

To summarise and compare the results obtained from the different techniques, the following criteria will be used:

- The ***size*** of the different models in terms of number of variables and constraints.
- The ***performance*** of the different models in terms of CPU solution time and the maximum objective value.

Note that, since no significant difference between the two global event models could be found, the first model's results will be used in the summary (refer to section 2.7.3).

The choice of a time representation technique, to be used for the SCS scheduling problem described in Chapter 1, will also be motivated.

a. Model size

To compare the size of the model, the same number of time points was used:

Table 2.7: Comparison of model sizes:

Model size criteria	Discrete	Time-slots	Global events	Unit-specific
Number of points p	6	6	6	6
Binary variables	72	72	72	72
Continuous variables	**235**	**451**	**313**	**451**
Constraints	**362**	**1028**	**577**	**1085**

It is clear that the size of the continuous time representation models is significantly larger than the size of the discrete model. This is the result of a much more complicated model structure to ensure sequencing of the event points. The time-slot and unit-specific models are especially large.

However, a conclusion from these results will be incomplete without reviewing the performance of these models.

b. Model performance

To compare the performance of the model, each model's best performance run was chosen for reporting. As explained earlier, a trade-off between CPU solution time and model accuracy was the key deciding factor.

Table 2.8: Comparison of model performances:

Model performance criteria	Discrete	Time-slots	Global events	Unit-specific
Number of points p	12	4	5	5
Binary variables	144	48	60	60
Continuous variables	469	301	360	376
Constraints	722	656	479	899
$Objective_{max}$ value	**268.57**	**373.95**	**285.34**	**373.95**
CPU time (sec)	47.01	50.42	62.42	53.72

The following conclusions can be made from the above comparison:

- ***Discrete model:***
  Even with 12 time points, the discrete model could not achieve the solution given by the time-slot and unit-specific models. By increasing the number of time intervals, the number of binary variables increased accordingly, resulting in a very large model (compare to Table 2.7 above). It can therefore be concluded that, although the discrete approach yields a much simpler model structure, the number of intervals needed to have a good representation of the problem increases the size of the model dramatically.

- ***Global event based model:***
  The global event based model also yielded inferior results compared to the time-slot and unit-specific event based models. Although the model's optimum performance is at five points, which results in a smaller model compared to the other two continuous approaches, the global model could not achieve the optimum answer.

- ***Time-slot and Unit-specific:***
  The time-slot and unit-specific event based models gave the best objective function values. This result confirms the theory described in section 2.3. These two continuous approaches give flexibility to the models to place the time points at the optimal points in the continuous time domain. Note that the unit-specific approach needed one additional

time point because the finish times for events at point p are allocated at the next point (p+1). Despite the good performance of these approaches, a major disadvantage is the resulting size of the models.

c. Model choice

After evaluating the results from the different models, the time-slot and unit-specific time representation techniques stood out as giving the best results for this specific example problem. The assumption was made that the results achieved with the example problem will be of the same nature when the different techniques were to be applied to the SCS scheduling problem described in Chapter 1.

After careful consideration, the ***unit-specific approach*** was chosen to be used for the problem described in Chapter 1. Although both the unit-specific and the time-slot models yield similar results, the unit-specific approach is the most recent contribution to the scheduling research field. Therefore, the application of this technique to an industrial-sized problem and the evaluation thereof, could add the most value to the scheduling research field.

2.8. SPECIAL ORDERED SETS (SOS1) – BINARY IMPROVEMENT

2.8.1. Basic Principles

There is a number of ways to improve a model's performance, one of which is the reduction of the number of binary variable in the model. The number of binary variables is a very important factor, which slow down the solution time of the model dramatically.

The approach proposed in this section is to use a special variable type available in GAMS. This variable type is called *Special Ordered Sets of Type 1* (hereafter called SOS1 variables). The definition for SOS1 variables from the GAMS user manual is as follows (Brooke et al, 1998):
"*At most one variable within a SOS1 set can have a non-zero value. This variable can take any positive value... The members of the innermost index belong to the same set*."

The following example illustrates the use of the SOS1 variables:

Let	Q_{ijp}	be a positive quantity variable for event i in unit j at point p
Let	w_{ijp}	be a binary variable indicating event i in unit j at point p

As explained in section 2.6.2, the following constraints ensure only one event happening at unit j at a time, and give the upper limit for the quantity processed:

$$\sum_i w_{ijp} \leq 1 \qquad \forall j, \quad \forall p$$
$$Q_{ijp} \leq rate\_b_i \times w_{ijp} \quad \forall i, \quad \forall j, \quad \forall p \tag{2.8.1}$$

To eliminate the use of the binary variable in the formulation above, the quantity variable is ***redefined*** as follows:

QQ_{jpi} A SOS1 quantity variable for event i in unit j at point p

Note that the position of the i-index moved to the end, to be the set under control.

The following constraint illustrates the principle of using SOS1 variables:

$$\sum_i QQ_{jpi} \leq rate\_b \qquad \forall j, \quad \forall p \tag{2.8.2}$$

Equation 2.8.2 ensures that at most one quantity variable of all the events i in a certain unit j has a value (equivalent to equation 2.8.1.a). If it indeed has a value, the upper limit is the rate for that specific event (equivalent to equation 2.8.1.b). By using the SOS1 variable in equation 2.8.2, both the equations in 2.8.1 were replaced.

Therefore, the SOS1-formulation explained above has the following benefits:

- The binary variable is eliminated.
- The number of equations (size of the model) is reduced.

2.8.2. Application to example problem

To illustrate the application and result of the SOS1 approach, the previous example problem will be used. The SOS1 approach will be compared to the performance of the unit-specific model. The results will be discussed in section 2.10.

The binary variable formulation described in section 2.6.2 and the quantity variable formulation described in section 2.6.6 will be reformulated. All the other constraints defined in sections 2.6.2 and 2.6.6 still apply, unless specifically stated otherwise.

a. Variables

The binary variable w_{ijp} and the continuous variable q_b_{ijp} defined in section 2.6.2 are replaced by the following two SOS1 variables:

q_supply_{ipj} The quantity of coal supplied from mine i at point p to the set of stockpiles j.

q_stack_{jpi} The quantity of coal stacked on stockpile j at point p from the set of mines i.

An additional binary variable was needed to ensure stacking and reclaiming do not take place simultaneously. However, this binary variable only has two indices:

y_{jp} = 1 when stacking coal on stockpile j at p

= 0 otherwise

b. Mathematical reformulation

b.1. Allocation constraints

Constraints 2.6.1 and 2.6.2 were replaced by the use of the two SOS1 variables. The reformulation of these constraints is discussed with the quantity constraints.

Equation 2.6.3 was reformulated to include the new binary variable y_{jp}:

$$y_{jp} + x_{jp} \leq 1 \qquad \forall j, \quad \forall p \tag{2.8.3}$$

This constraint ensures that stacking and reclaiming cannot take place at the same time.

Equation 2.6.41 was replaced by the following equation to include the new binary variable y_{jp}:

$$y_{jp} = 0 \qquad \forall j, \quad \forall p = p_n \tag{2.8.4}$$

b.2. Storage constraints

Constraint 2.6.5 was reformulated to include the SOS1 variable in the material balance of the stockpile and to eliminate the previous quantity variable q_b_{ijp}.

$$ST\_s_{ijp} = ST\_s_{i,j,p-1} - q\_r_{i,j,p-1} + q\_stack_{j,p-1,i} \qquad \forall i, \quad \forall j, \quad \forall p > p_1 \tag{2.8.5}$$

b.3. Quantity and Duration constraints

To reformulate the action of conveying coal from the mines to the stockpiles, the following set of equations was used:

$$\sum_j q\_supply_{ipj} \leq rate\_b_i \times H \qquad \forall i, \quad \forall p \tag{2.8.6}$$

$$\sum_i q\_stack_{jpi} \leq HH \times y_{jp} \qquad \forall j, \quad \forall p \tag{2.8.7}$$

$$q\_supply_{ipj} = q\_stack_{jpi} \qquad \forall i, \quad \forall j, \quad \forall p \tag{2.8.8}$$

The first constraint replaces equation 2.6.1, ensuring a mine i send coal to only one stockpile j at a time. An upper limit for the quantity conveyed is set simultaneously. The second constraint replaces equation 2.6.2, ensuring a stockpile j to receive coal from only one mine i at a time. The right hand side of the equation ensures that no coal is conveyed to the stockpile if the stacking action is not activated. Since these two SOS1 variables refer to the same event, namely conveying coal from a mine i to a stockpile j, they should be equal, yielding the third equation. These equations replace equation 2.6.15 used in the unit-specific model.

The duration of a conveying event from mine i to stockpile j is described by the following constraint (replacing equation 2.6.14):

$$q\_supply_{ipj} = rate\_b_i \times Dur\_b_{ijp} \qquad \forall i, \quad \forall j, \quad \forall p \qquad (2.8.9)$$

Note that, with the SOS1 formulation, the quantity conveyed is first determined by the SOS1 variable, and then the duration is calculated. The order is different from the previous unit-specific formulation, where the duration of the event was determined first and then applied to the quantity variable.

b.4. Objective function

Lastly, the previous q_b_{ijp} quantity variable in the objective function had to be replaced by the SOS1 variable:

$$z_{max} = \sum_{ijp}\left(q\_r_{ijp} \times Income\right) - \sum_{ijp}\left(q\_supply_{ipj} \times Cost_i\right) \qquad (2.8.10)$$

Equation 2.8.9 replaces 2.6.10.

2.9. NLP SOLUTION IMPROVEMENT

2.9.1. Basic Principles

Apart from the amount of binary variables in the model, the amount of non-linear equations in the formulation is also an important factor determining the model solution time. This section focuses on the reformulation and alternative solution method for non-linear equations in a model.

Thus far, the following three types of non-linear equations were encountered:

- *Binary $\times$ Binary*

  The case where two binary variables are multiplied was discussed in section 2.6.2.e. This non-linearity was linearized by adding the two binary variables rather than using multiplication (refer to discussion).
- *Continuous $\times$ Binary*

  The case where a continuous variable is multiplied with a binary variable was discussed in section 2.6.4.a. This non-linearity was linearized using Glover's technique (1975).
- *Continuous $\times$ Continuous*

  The case where a continuous variable is multiplied with another continuous variable was encountered in section 2.6.2, equation 2.6.8. This non-linearity could not be linearized, and resulted in the only remaining non-linear equation in the base model. The rest of this section will discuss a technique for the reformulation and an alternative solution method for this type of non-linear equations.

Quesada and Grossmann (1995) suggested a method to simplify the solution process of what they call "bilinear" equations. The method exploits the fact that each of the two continuous variables has an upper and a lower limit. These limits are used in linearization equations (McCormick, 1976) which are solved as part of a MILP pre-model to determine the upper and lower bounds for the real MINLP model.

To implement the proposed method, the following five steps should be followed.

Step 1:

Let the following equation be the non-linear (bilinear) equation under consideration, where m_{ijp}, F_{ip} and G_{ijp} are all positive continuous variables:

$$m_{ijp} = F_{ip} \times G_{ijp} \qquad \forall i, \quad \forall j, \quad \forall p \tag{2.9.1}$$

Step 2:

Consider the upper and lower bounds on each of the continuous variables in the bilinear relation:

$$\begin{aligned} Fmin_i \le F_{ip} \le Fmax_i \qquad & \forall i, \quad \forall p \\ Gmin_{ij} \le G_{ijp} \le Gmax_{ij} \qquad & \forall i, \quad \forall j, \quad \forall p \end{aligned} \tag{2.9.2}$$

Step 3:

Formulate the proposed linearization equations as follows:

$$
\begin{aligned}
m_{ijp} &\geq \quad Fmin_i.G_{ijp} + Gmin_{ij}.F_{ip} - Fmin_i.Gmin_{ij} \\
m_{ijp} &\geq \quad Fmax_i.G_{ijp} + Gmax_{ij}.F_{ip} - Fmax_i.Gmax_{ij} \\
m_{ijp} &\leq \quad Fmax_i.G_{ijp} + Gmin_{ij}.F_{ip} - Fmax_i.Gmin_{ij} \\
m_{ijp} &\leq \quad Fmin_i.G_{ijp} + Gmax_{ij}.F_{ip} - Fmin_i.Gmax_{ij} \\
& \qquad \forall i, \quad \forall j, \quad \forall p
\end{aligned}
\tag{2.9.3}
$$

Step 4:

Define the following two models:

- ***MILP model:*** Include the upper and lower limit equations 2.9.2 and the linearization equations 2.9.3. All other relevant equations as described in the other sections are also included in this model. The non-linear equation 2.9.1 is excluded from this model.
- ***MINLP model:*** Include upper and lower limit equations 2.9.2 and the non-linear equation 2.9.1. All other relevant equations as described in the other sections are also included in this model. This time, the linearization equations 2.9.3 are excluded.

Step 5:

First solve the MILP model. The solution found with this model provides upper and lower bounds to the MINLP problem (Quesada and Grossmann, 1995). The MILP solution is used as input to the MINLP model which is solved after the inputs have been received.

Note that a comprehensive description of the theory behind this method and the NLP linearization will not be discussed in this document, but can be obtained from Quesada and Grossmann's paper (1995) and McCormick's paper (1976). The purpose of this study is only to evaluate the method's performance when applied to an industrial-sized problem.

2.9.2. Application to example problem

To illustrate the use of the method proposed by Quesada and Grossmann (1995), it will be applied to the example problem described in section 2.6.

Step 1:

As mentioned above, the only non-linear equation in the example problem is the reclaiming constraint in equation 2.6.8. However, this equation consists of two bilinear parts, one on the left hand side, and one on the right hand side. In step 1, two new continuous variables, $Ratio1_{i,ii,j,p}$ and $Ratio2_{ii,i,j,p}$ are created and the devisor in each term is multiplied across, to ensure the correct application of the method:

$$
\begin{aligned}
&Ratio1_{i,ii,j,p} = q\_r_{ijp}.ST\_s_{ii,j,p}\\
&Ratio2_{ii,i,j,p} = q\_r_{ii,j,p}.ST\_s_{ijp}\\
&Ratio1_{i,ii,j,p} = Ratio2_{ii,i,j,p}\\
&\quad \forall i \neq ii, \quad \forall j, \quad \forall p
\end{aligned}
\tag{2.9.4}
$$

Step 2:

Only two continuous variables' limits need to be considered, since set i and set ii are the same. Furthermore, these variables do not have a lower bound, and therefore, the upper limits can be set as follows:

$$
\begin{aligned}
&q\_r_{ijp} \leq rate\_r_j \times H \qquad \forall i, \quad \forall j, \quad \forall p\\
&ST\_s_{ijp} \leq Cap\_s \qquad \forall i, \quad \forall j, \quad \forall p
\end{aligned}
\tag{2.9.5}
$$

The upper bound for the amount of coal to be reclaimed is determined by the rate at which coal can be reclaimed. Another upper bound was set in equation 2.6.7, using the amount of that specific mine's coal on the stockpile as an upper limit for the reclaiming action. This still applies. However, for the application of the linearization technique, the limits must be set in terms of constant values.

The upper bound on the amount of coal from a specific mine on that stockpile, is determined by the capacity of the stockpile.

Step 3:

The proposed linearization equations were applied as follows:

$$
\begin{aligned}
&Ratio1_{i,ii,j,p} \geq \quad rate\_r_j.H.ST\_s_{ii,j,p} + Cap\_s.q\_r_{ijp} - rate\_r_j.H.q\_r_{ijp}\\
&Ratio1_{i,ii,j,p} \leq \quad rate\_r_j.H.ST\_s_{ii,j,p}\\
&Ratio1_{i,ii,j,p} \leq \quad Cap\_s.q\_r_{ijp}\\
&\qquad \forall i \neq ii, \quad \forall j, \quad \forall p
\end{aligned}
\tag{2.9.6}
$$

Note that the structure of this formulation is slightly different from the original proposed formulation. This is due to the fact that the continuous variables do not have lower bounds in the example problem.

Similar to equation 2.9.6, the following equations are defined:

$$\begin{aligned} Ratio2_{ii,i,j,p} &\geq rate\_r_j.H.ST\_s_{ijp} + Cap\_s.q\_r_{ii,j,p} - rate\_r_j.H.q\_r_{ii,j,p} \\ Ratio2_{ii,i,j,p} &\leq rate\_r_j.H.ST\_s_{ijp} \\ Ratio2_{ii,i,j,p} &\leq Cap\_s.q\_r_{ii,j,p} \\ &\forall i \neq ii, \quad \forall j, \quad \forall p \end{aligned} \tag{2.9.7}$$

Step 4 and 5:

The models were defined and solved as described in section 2.9.1. All results will be discussed in section 2.10.

2.10. IMPROVEMENT RESULTS

To compare the performance of the SOS1 and NLP linearization improvements discussed in sections 2.8 and 2.9 above, the unit-specific event based model was used as reference point. The number of time points was kept constant to emphasise the effect of the improvement on the model performance.

The NLP linearization technique was first applied to the unit-specific model as described above. Then the SOS1 variable approach was applied to the original unit-specific model. Lastly, both these improvements were applied to the unit-specific model. The results are summarised in Table 2.9:

Table 2.9: Comparison of model performances after certain improvements:

Model performance criteria	**Unit-specific**	**NLP linearization ONLY**	**SOS1 ONLY**	**SOS1 & NLP linearization**
Number of points p	6	6	6	6
Binary variables	72	72	36	36
Continuous variables	451	451	469	469
Constraints	1085	1193	1079	1187
Objective$_{max}$ value	374.86	374.56	374.86	374.72
CPU time (sec)	**388.60**	**200.00**	**188.00**	**158.00**

a. Binary variables

As expected, the use of SOS1 variables reduced the total number of binary variables in the model formulation. The effect of this reduction can clearly be seen in the reduction of CPU solution time from 388 seconds to 188 seconds. This represents an improvement of 51.5% in

CPU solution time, while the objective function value remained the same. Thus, the objective function value was not compromised in order to achieve the solution time reduction.

b. CPU time

All the improvement efforts (SOS1 and NLP linearization) resulted in faster CPU times. All of the improvement models achieved the same objective value yielded by the unit-specific approach, thus optimality was not compromised to achieve the faster solution times. When the SOS1 and the NLP linearization improvements were simultaneously incorporated in the model, it resulted in the best overall CPU time.

c. Result Conclusion

Both the SOS1 variables and the NLP linearization improvements resulted in very good solutions with fast solution times. Both these approaches are implemented and evaluated during the development improvements of the SCS scheduling problem in Chapter 4.

2.11. TECHNIQUE EVALUATION AND IMPROVEMENT CONCLUSION

In this chapter, four scheduling time representation techniques were presented according to the classification given by Floudas and Lin (2004). These techniques include the discrete time formulation and three continuous time formulations, namely the time-slot formulation, the global event based formulation and the unit-specific event based formulation. Each technique's advantages and disadvantages were discussed.

An example problem was formulated to assist in the demonstration of the different techniques' application and detail mathematical formulation. The results from each technique's model solution were compared in terms of model size and model performance. Based on this particular case's solutions, the time-slot and unit-specific approaches yielded the best results.

The *unit-specific* event based continuous time formulation was chosen to apply to the SCS scheduling problem described in Chapter 1. Although the unit-specific and the time-slot methods yielded very similar results, the unit-specific approach was chosen because it is the latest contribution to the research field of continuous time formulation for scheduling. The application of the unit-specific formulation to the industrial-sized SCS scheduling problem is further discussed in Chapter 3.

Finally, two model performance improvement methods were discussed. The first method focussed on the reduction of binary variables in a model by using SOS1 variables. The second method explored the linearization and alternative solution method for non-linear equations in a model. The linearization equations substitute the non-linear equations in a preliminary mixed integer linear programming (MILP) model which is solved to establish upper and lower bounds for the real mixed integer non-linear programming (MINLP) model.

Both these improvement methods were applied to the example problem and resulted in very good solutions with fast solution times. These methods are also applied to the SCS scheduling model improvement phase which is discussed in Chapter 4.

CHAPTER 3: MODEL DEVELOPMENT (PHASE 1)

The SCS scheduling model is developed in two phases. The first development phase is described in this chapter. The basic mathematical formulation according to the unit-specific event based time representation is discussed in detail. The specific changes made to the formulation to customise it for the SCS environment are highlighted. Finally, the preliminary results from the first phase model are presented.

3.1. MODEL DEVELOPMENT STRATEGY

The development of the SCS scheduling model consists of two phases:

Phase 1: Basic Model
During phase 1 the basic coal volume movement is modelled, using the unit-specific event based time representation formulation presented in Chapter 2. The specific challenges posed by applying this formulation to the SCS situation are evaluated and discussed. The results are presented and evaluated to determine the performance of the unit-specific approach when applied to industrial-sized problems.

The basic problem formulation is presented in this chapter using the coal flow diagram introduced in Chapter 1 (Figure 1.2).

Phase 2: Improvement of model for operational use
During phase 2, the problem formulation of phase 1 is improved to ensure that the model solves in an acceptable time for operational use. This is discussed in Chapter 4.

3.2. DEVELOPMENT PRINCIPLES

The unit-specific event based time representation approach was discussed in Chapter 2. According to this approach, different tasks *i* in different units *j* can refer to one event point *p*, and yet take place at different times in the continuous time horizon (refer to Figure 2.5).

However, to emphasise the modelling approach followed during development, the following basic principles are highlighted again. All the principles aim at improving the model structure and minimising the solution time.

1. As far as possible, **non-linear equations** should be transformed to linear equations. This will ensure a minimum amount of non-linear equations, which will improve the model solution time. The following linearization techniques will be used:

   - *(Binary variable1).(Binary variable2) = 0*

     Using the Watters transformation, the non-linear equation above can be transformed to be linear:

     $$\begin{aligned} & BinaryVar1 + BinaryVar2 \leq 1 \\ & 0.5(BinaryVar1 + BinaryVar2) \geq 0 \end{aligned} \quad (3.1.1)$$

     Note that this linearization will force one or both of the binary variables to be 0. This is ideal for situations where either one or the other option can hold, but not both.

   - *(Binary variable).(Continuous variable1) = Continuous variable2*

     Using the linearization technique described by Glover (1975), the non-linear equation above can be transformed to be linear:

     $$\begin{aligned} & ContinuousVar2 \leq ContinuousVar1 + M(1 - BinaryVar) \\ & ContinuousVar2 \geq ContinuousVar1 - M(1 - BinaryVar) \\ & ContinuousVar2 \leq M.BinaryVar \end{aligned} \quad (3.1.2)$$

     Note that M is any big value and the continuous variables are positive. Thus, when the binary variable equals 0, continuous variable 2 will also equal 0 (third equation). When the binary variable equals 1, the relaxation term *M(1-BinaryVar)*, will disappear and continuous variable 2 will be equal to continuous variable 1 (first and second equations).

   - *(Continuous variable1).(Continuous variable2) = Continuous variable3*

     This type of non-linear equations should be restricted to the minimum. No linearization method is available to totally eliminate these non-linear equations. Therefore these equations will cause the model to be a Mixed Integer Non-Linear Programming (MINLP) model, which is more difficult to solve than Mixed Integer Linear Programming (MILP) models.

2. The amount of **event points** used should be minimised. Since every variable has p (the set of event points) as one of its indices, the reduction of event points would also reduce the total amount of variables in the model.

3. The number of **binary variables** should be restricted to the minimum. Binary variables slow down the solution time of a model.

4. The values used in the model must be **scaled** to ensure optimum model performance.

These principles, together with the unit-specific continuous time representation approach will form the basis for the model development in this chapter.

3.3. PHASE 1: MATHEMATICAL FORMULATION

3.3.1. Introduction

The basic coal flow diagram for one coal source was illustrated by Figure 1.2 in Chapter 1. Referring to this figure, the development of the basic model will be discussed under the following headings:

- Declaration of sets and variables.
- User input (parameters and constants).
- Source production profiles and supply.
- Coal outside a bunker, throw out and load back.
- Bunkers, extraction from bunkers and the bypass option.
- Strategic stockpiles, strategic throw out and load back.
- Stacking.
- Reclaiming and blend limits.
- Stockpiles.
- Factory demand and restrictions.
- Maintenance.
- Objective function.

3.3.2. Declaration of sets and variables

a. Sets:

p = set of event points $p \in \{ p_1, p_2, p_3, \dots, p_{12} \}$

r = set of time periods $r \in \{ period_1, period_2, \dots, period_6 \}$

per_{rp} = subset to assign event points to periods

$period_1 \in \{ p_1, p_2 \}$ $period_4 \in \{ p_7, p_8 \}$

$period_2 \in \{ p_3, p_4 \}$ $period_5 \in \{ p_9, p_{10} \}$

$period_3 \in \{ p_5, p_6 \}$ $period_6 \in \{ p_{11}, p_{12} \}$

The assignment of event points to periods will be discussed in 3.3.4.

i and *ii*	=	set of sources	*i* or *ii* $\in$ { *Br, Mb, Bo, Tw, Syf, Mdl*}
j and *jj*	=	set of stockpile yards	*j* or *jj* $\in$ { $y_1, y_2, \ldots, y_6$ }
k and *kk*	=	set of individual stockpiles per yard	*k* or *kk* $\in$ { sp_1, sp_2, sp_3, sp_4 }

b. Variables:

b.1. Binary Variables:

w_{ijp} = 1 when extracting coal from source i to stockpile j at point p
= 0 otherwise

$v1_{ip}$ = 1 when coal is thrown out at bunker i at point p
= 0 otherwise

$v2_{ip}$ = 1 when coal is loaded back into bunker i at point p
= 0 otherwise

ww_{jkp} = 1 when stacking coal on yard j, stockpile k at point p
= 0 otherwise

x_{jkp} = 1 when reclaiming coal from yard j, stockpile k at point p
= 0 otherwise

$z1_{jp}$ = 1 when coal is thrown out at strategic stockpile j at point p
= 0 otherwise

$z2_{jp}$ = 1 when coal is loaded back from strategic stockpile j at point p
= 0 otherwise

y_{jkp} = 1 when a new stockpile is created on position k, yard j, at point p
= 0 otherwise

$x2_{ijp}$ = 1 when bypassing coal from mine i to yard j, at point p
= 0 otherwise

b.2. Positive variables:

Quantity variables:

q_m_{ip}	Amount of coal produced by source i at point p (kt)
q_bo_{ip}	Amount of coal thrown out at bunker i at point p (kt)
q_bl_{ip}	Amount of coal loaded back at bunker i at point p (kt)
q_b_{ijp}	Amount of coal extracted from bunker i to yard j at point p (kt)
q_s_{ijkp}	Amount of coal from source i stacked on yard j, stockpile k at point p (kt)
q_r_{ijkp}	Amount of coal from source i reclaimed from yard j, stockpile k at point p (kt)

q_prop_{ijp}	Amount of coal from source i bypassed at yard j, at point p (kt)
q_so_{jp}	Amount of coal thrown out at strategic stockpile j at point p (kt)
q_sl_{jkp}	Amount of coal loaded back from strategic stockpile j to stockpile k at point p (kt)
q_rsl_{jkp}	Amount of coal from strategic stockpile j that were loaded back to stockpile k and reclaimed at point p (kt)
$Total\_e_p$	Total feed to the Eastern factory at point p (kt)
$Total\_w_p$	Total feed to the Western factory at point p (kt)

Storage variables:

ST_b_{ip}	Amount of coal stored in bunker i at point p (kt)
Out_b_{ip}	Amount of coal outside bunker i at point p (kt)
ST_sm_{ijkp}	Amount of coal from source i on stockpile k, yard j, at point p (kt)
ST_sl_{jkp}	Amount of strategic stockpile coal on stockpile k, yard j, at point p (kt)
$Strat_{jp}$	Amount of coal on strategic stockpile j at point p (kt)
$heapl\_s_{jkp}$	Heap length of stockpile k, yard j, at point p (mx10)

Time and duration variables:

Ts_m_{ip}	Starting time value if source i production *starts* at point p (h)
Tf_m_{ip}	Finishing time value if source i production *stops* at point p (h)
Dur_m_{ip}	Duration of source i production if production starts at point p (h)
Ts_bl_{ip}	Starting time value if loading back coal at bunker i *starts* at point p (h)
Tf_bl_{ip}	Finishing time value if loading back coal at bunker i *stops* at point p (h)
Dur_bl_{ip}	Duration of loading back coal at bunker i if it starts at point p (h)
Ts_b_{ijp}	Starting time value if extracting coal from bunker i to yard j *starts* at point p (h)
Tf_b_{ijp}	Finishing time value if extracting coal from bunker i to yard j *stops* at point p (h)
Dur_b_{ijp}	Duration of extracting coal from bunker i to yard j if it starts at point p (h)
Ts_s_{jkp}	Starting time value if stacking coal from source i on stockpile k, yard j *starts* at point p (h)
Tf_s_{jkp}	Finishing time value if stacking coal from source i on stockpile k, yard j *stops* at point p (h)
Dur_s_{jkp}	Duration of stacking coal on stockpile k, yard j if it starts at point p (h)

Ts_sl_{jp}	Starting time value if loading back coal from strategic stockpile j *starts* at point p (h)
Tf_sl_{jp}	Finishing time value if loading back coal from strategic stockpile j *stops* at point p (h)
Dur_sl_{jkp}	Duration of loading back coal from strategic stockpile j to specific stockpile k, if it starts at point p (h)
Ts_r_{jkp}	Starting time value if reclaiming coal from stockpile k, yard j *starts* at point p (h)
Tf_r_{jkp}	Finishing time value if reclaiming coal from stockpile k, yard j *stops* at point p (h)
Dur_r_{jkp}	Duration of reclaiming coal from stockpile k, yard j if it starts at point p (h)
Pos_r_{jkp}	Reclaimer position tracking variable for stockpile k, yard j,at point p

b.3. Variable:

z_{max}	Objective function variable to maximise profit

3.3.3. User input (parameters and constants)

The following parameters and constant values will be determined by the user and will be used as input to the model. The detail values may differ each time the model is solved, since the actual plant situation is dynamic. An example of the user input interface may be seen in Appendix B.

a. Parameters:

In the context of this document, a parameter is a constant value that has an index. Parameters with two or more indices result in tables.

$ST0\_b_i$	Starting level of bunker i (%)
$Out0\_b_i$	Starting level of coal outside bunker i (kt)
$Strat0_j$	Starting level of strategic stockpile j (kt)
$ST0\_sm_{ijk}$	Starting level of source i contribution to stockpile k on yard j (kt)
$ST0\_sl_{jk}$	Starting level of coal from strategic stockpile j on stockpile k (kt)
$heap0\_s_{jk}$	Starting heap length of stockpile k on yard j (m x 10)
Cap_b_i	Capacity of bunker i (kt)

Cap_strat_j	Capacity of strategic stockpile j (kt)
$Rate\_b_j$	Maximum rate for conveying coal from bunker i (kt/h)
$Rate\_m_{ir}$	Production rate of source i in period r (kt/h)
CL_i	Limits for source i contribution to the total feed to the factory at any point (%)
$Ts0\_m_r$	Starting time value for each period r (h)
H_m_r	End-of-period time value for each period r (h)
CT_b_{ij}	Change-over time lost when starting extraction from bunker i to stockpile j (h)
$CT\_s_{k,kk}$	Change-over time lost when changing the stacker position from stockpile k to kk (h)
$CT\_r_{k,kk}$	Change-over time lost when changing the reclaimer position from stockpile k to kk, given there are no stockpiles between k and kk (h)
$Pos0\_s_{jk}$	Starting stacker position on stockpile k on yard j
$Pos0\_r_{jk}$	Starting reclaiming position on stockpile k on yard j
$Max0\_s_{jk}$	The maximum capacity of a stockpile k on yard j (kt / m x10)

b. Constants:

In the context of this document, a constant is a value that remains the same throughout the scheduling model's time horizon, but does not have an index. In the GAMS context these constant values are called scalars.

H	Scheduling time horizon (chosen to be 24 hours)
HH	A chosen big value e.g. 100
Bunk_min	The minimum level of any bunker (%)
Dur_min	The minimum duration when coal is extracted from a bunker (h)
Rate_bl	Rate for loading back coal with one front-end loader at the bunkers or the strategic stockpiles (kt/h)
Rate_r	The maximum rate at which any reclaimer can reclaim (kt/h)
Rate_s	The maximum rate at which any stacker can stack (kt/h)
Rate_f	The maximum rate at which one side of the factory can receive coal

Tot_length	Total length of any yard j (m x10)
Min_length	The minimum length for any individual stockpile k (m x10)
Cost	The handling cost when throwing out and loading back coal at the bunkers or the strategic stockpiles (R/kt)
Income_s	Income per kt stacked (R/kt) (this is a fictional value which will be explained in 3.3.13)
Income_r	Income per kt reclaimed (R/kt)
Prop_penalty	Penalty when coal is bypassed directly to the factory(R/kt) (this is a fictional value which will be explained in section 3.3.13)
CL_sl	Limit for strategic stockpile coal contribution to the total amount of coal supplied to the factory at any point p (%)
Loaders	The number of front-end loaders available for loading back coal at the bunkers and the strategic stockpiles
Dem_w	Coal demand of Western factory during time horizon
Dem_e	Coal demand of Eastern factory during time horizon

3.3.4. Source production profiles and supply

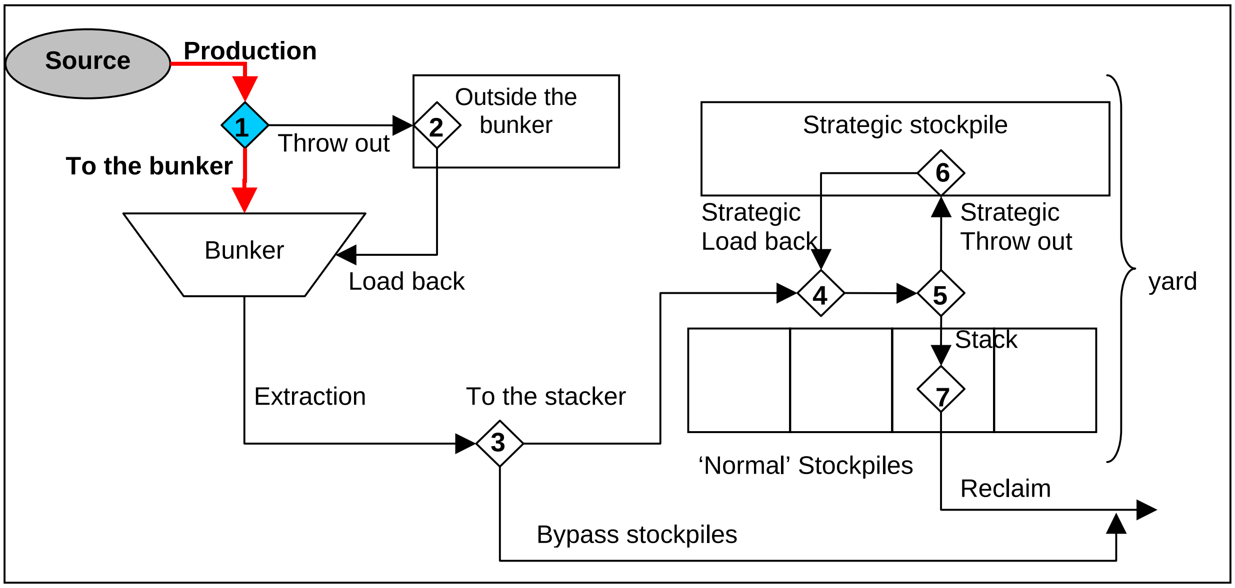
3.3.4.1. Problem statement

Referring to Figure 1.2, the boundaries for this section are indicated in Figure 3.1.

Only the one option of decision point 1, namely to throw the produced coal into the bunker, is considered in this section. The other option, to throw the produced coal outside the bunker will be discussed in section 3.3.5.

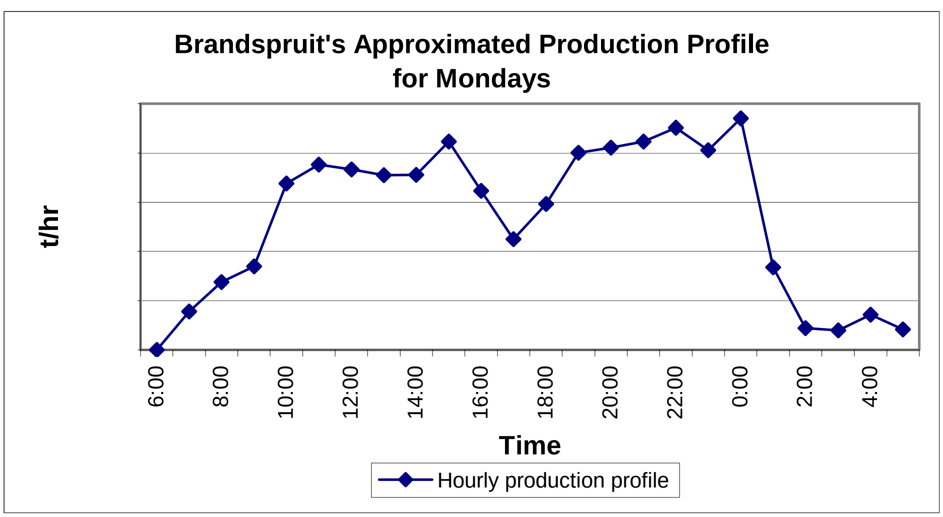
Each source produces coal at a specific rate (t/hr) which varies during a 24-hour period (Figure 3.2). Note that the detail production rates are hidden to ensure confidentiality. The variation illustrated in Figure 3.2 is a direct result of activities at the start and end of production shifts at the mines. Therefore, each mine has a unique production profile for every day of the week. These production profiles are very important, since the mines' production initiate the whole coal supply chain and its time sequence.

Figure 3.1: Coal source production and supply:



However, a source's production as per production profile is given in terms of hourly figures, which is in direct conflict with the unit-specific event point based continuous time representation described in Chapter 2. Because of this conflict, it was decided *to combine the global event based time representation technique with the unit-specific event point based approach.*

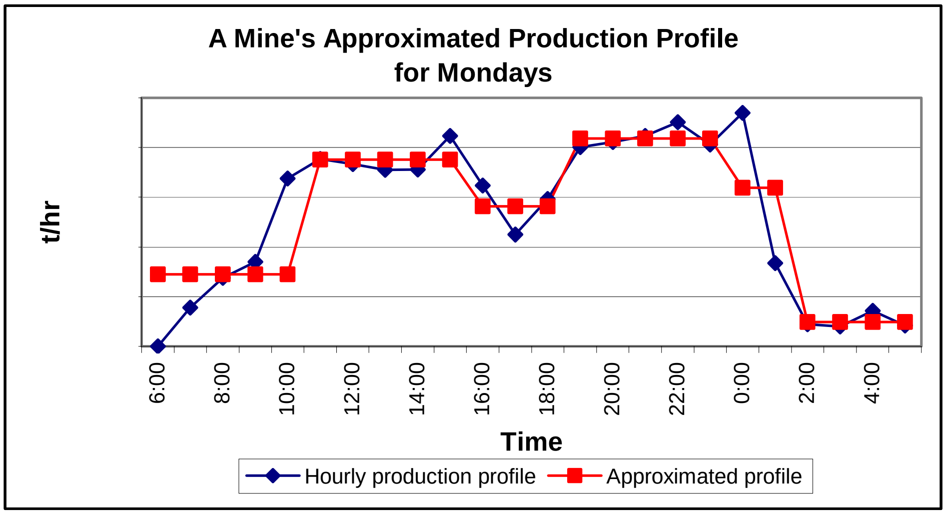
Figure 3.2: An example of a mine's production profile for a specific day:



3.3.4.2. Approximated production profiles

The daily production profiles for each source were calculated using the average of six weeks' production data. Profiles for every day of the week and for each source were created, including the Fridays when the mines only have one shift. After the profiles were created, it were analysed and divided into periods with fixed global starting and finishing times. Periods were chosen such to minimise both the number of periods per day and the variation in the production rate during the period. Thus, every production profile has exactly the same periods, with the same starting and finishing times.

Figure 3.3: Approximation of production profiles by using periods:



For each period, an average production rate was calculated to approximate the hourly production profile during that period (Figure 3.3). The periods with its fixed starting and finishing times were applied to all profiles.

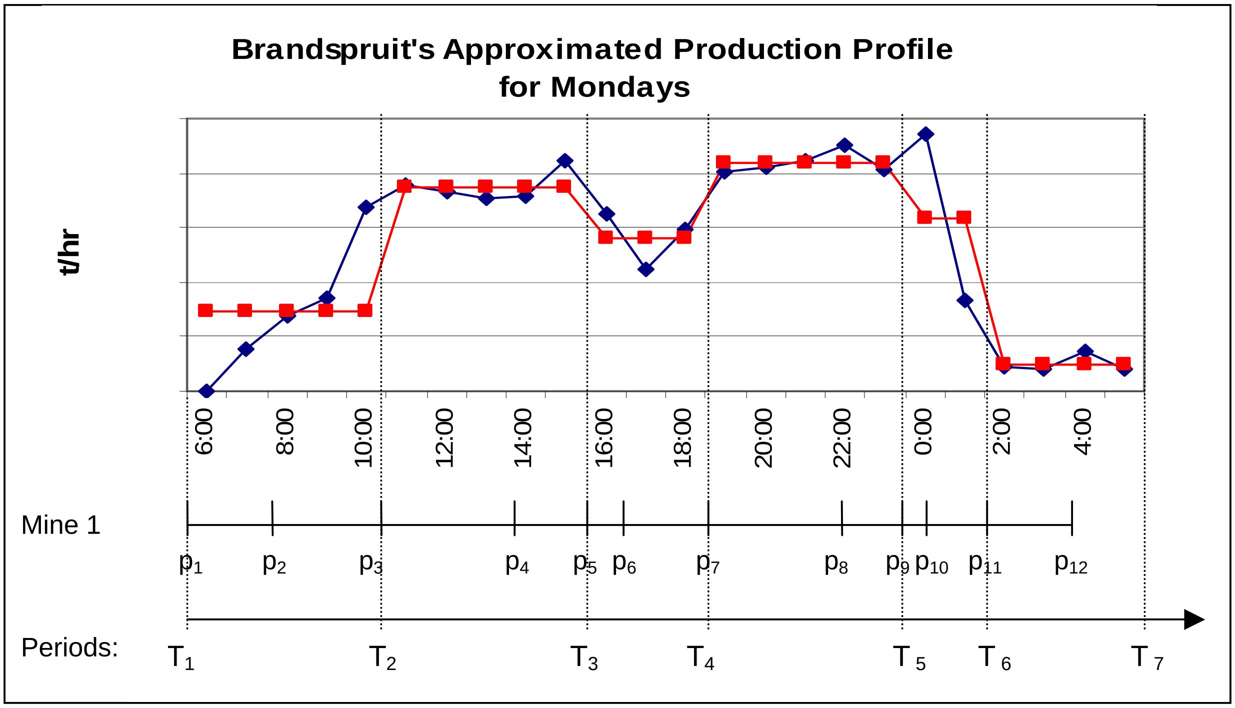
To determine the periods, the following was taken into account:

- The number of periods had to be minimised, while still giving a good approximation of the hourly production profile.
- The length of a period was determined by minimising the variation of the hourly production rates around the approximated average for that period.

To combine the global event based time representation technique with the unit-specific event point based approach, two event points were allocated to each period (refer to Figure 3.4). The first event point was given a fixed starting time which coincided with the starting time of the period. The second event point's finishing time had to be equal to or less than the period's fixed end time. Thus, the finish time of the first point and the starting time of the second point provided flexibility to the fixed start and finished times of a period.

This ensured time sequence with the sources' production profiles, but also gave the required flexibility of the unit-specific event based time representation.

Figure 3.4: Graphical representation of the technique combination:



Note that more event points may be allocated to a period to give even more flexibility, but an increase in the number of event points will increase the size of the model exponentially. Table 3.1 illustrates the event point allocation:

Table 3.1: Example of event point allocation:

Periods	**Period start time**	**Period end time**	**Event points**	**Event starting time**	**Previous event's finishing time**
Period 1	06:00	11:00	p1	= 06:00	N/A
			p2	≤ 11:00	≤ 11:00
Period 2	11:00	16:00	p3	= 11:00	≤ 11:00
			p4	≤ 16:00	≤ 16:00
Period 3	16:00	19:00	p5	= 16:00	≤ 16:00
			p6	≤ 19:00	≤ 19:00
Period 4	19:00	24:00	p7	= 19:00	≤ 19:00
			p8	≤ 24:00	≤ 24:00
Period 5	0:00	02:00	p9	= 0:00	≤ 24:00
			p10	≤ 02:00	≤ 02:00
Period 6	02:00	06:00	p11	= 02:00	≤ 02:00
			p12	N/A	≤ 06:00

3.3.4.3. Mathematical formulation

The sources' production was modelled as follows:

a. Time and sequencing constraints

For the mathematical formulation of the time and sequencing constraints, the modelling starting time value of the time horizon is always set to 0. For example, if the schedule is required for a 24 hour period starting from 11:00, the modelling starting time will be 0, and all output time values will have the value of (x + 11). This prevents a sequencing problem at midnight when the time changes from 24:00 to 0:00.

As explained in Chapter 2, the principle used to model the start and finish times of an event, states that an event starts at one point p and finishes at the next point (p+1). An event point may indicate the start of an event, the end of a previous event, or both. Thus, resulting in the following equation:

$$Tf\_m_{i,p+1} = Ts\_m_{ip} + Dur\_m_{ip} \qquad \forall i, \quad \forall p < p_n \tag{3.2.1}$$

Equation 3.2.1 ensures that all finish time values (at p+1) are equal to the associated starting time values (at p) plus the duration of the production event (during p). This will cause the starting and finishing time values to be equal when the duration of an event is 0, indicating that the event did not take place.

To ensure that the scheduled production matches the production profile, the event duration during a period has to equal the total length of that period. The constraint is valid because the production profile already accounts for the times when production is low or even 0.

$$\sum_{p \, \forall per_{rp}} Dur\_m_{ip} = H\_m_r - Ts0\_m_r \qquad \forall i, \quad \forall r \tag{3.2.2}$$

Note that only the points p allocated to the specific period r are added in equation 3.2.2. For example, when period 1 is under consideration, only points p_1 and p_2 are part of the summation.

As explained in section 3.3.4.2, the first event point in every period has a fixed starting time, due to the approximation of the production profiles:

$$Ts\_m_{ip} = Ts0\_m_r \qquad \forall i, \quad \forall p \in \{p_1, p_3, p_5, ..., p_{11}\}, \quad \forall per_{rp} \tag{3.2.3}$$

To ensure the correct sequencing of an event point in every period, the next event should start after the previous event has finished:

$$Ts\_m_{ip} \geq Tf\_m_{ip} \qquad \forall i, \quad \forall p \tag{3.2.4}$$

The finish time of every event has to be less than the associated period's end time:

$$Tf\_m_{i,p+1} \leq H\_m_r \qquad \forall i, \quad \forall p < p_n, \quad \forall per_{rp} \tag{3.2.5}$$

Note that the finish time value of an event starting at point p is allocated to the next point (p+1), even if the next point (p+1) is not associated with the same period as p (see the example in Table 3.1).

b. Production constraints

The amount of coal produced at a certain point p is a function of the allocated duration from equations 3.2.1 and 3.2.2, as well as the approximated production rate for the associated period:

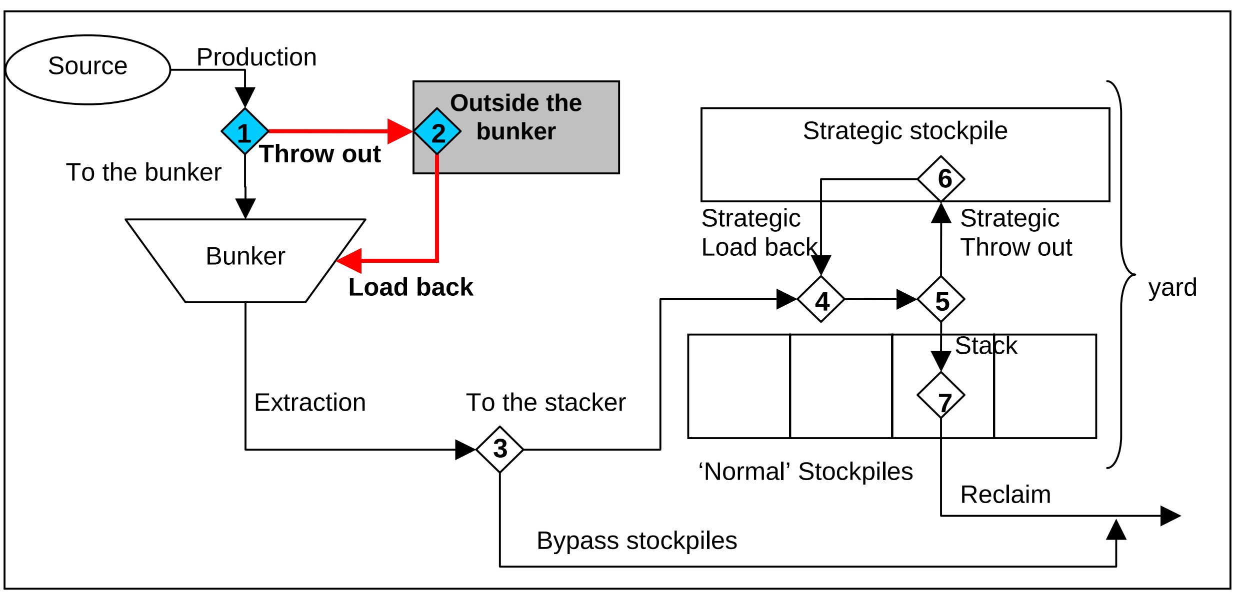
$$q\_m_{ip} = Dur\_m_{ip} \times Rate\_m_{ir} \qquad \forall i, \quad \forall per_{rp} \qquad (3.2.6)$$

3.3.5. Coal outside a bunker, Throw out and Load back

3.3.5.1. Problem statement

Referring to Figure 1.2, the boundaries for this section are illustrated by Figure 3.5. As explained in Chapter 1, throwing out coal outside the bunker is only an emergency measure when the bunker is full, to ensure that the mine's production is not stopped. Coal thrown out accumulates additional handling cost (cost penalty) and creates additional fine coal (quality penalty).

Figure 3.5: Coal outside a bunker, throw out and load back:



At decision point 1, there are two coal flow options: Either the mine's production is thrown into the bunker, or the total production for that event p is thrown out outside the bunker.

When coal is loaded back into the bunker, front-end loaders are used. There is a limited amount of front-end loaders available, which is shared between all the mine bunkers and the strategic stockpiles (refer to section 3.3.7). This places a constraint on the amount of coal that may be loaded back.

The following operating rules are applicable when throwing out or loading back coal at the bunkers:

- When starting to throw out coal outside the bunker, the coal should be thrown out until the bunker has reached the 75% level again. Note that throwing out coal does not reduce the bunker level. The bunker level can only be reduced by extracting coal to the stockpiles. This is only a guideline and may be challenged (see Chapter 4).
- When starting to load back, the bunker must not be more than 75% full by the end of the loading back activity. Should this operating rule not be adhered to, it may happen that the mine's production must be thrown out at point p+1, as a result of the amount of coal that was loaded back, and filling up the bunker at point p.
- As a result of the two operating rules above, throwing out and loading back may never happen simultaneously (decision point 2).

3.3.5.2. Mathematical formulation

a. Allocation constraints

Equation 3.3.1 uses the binary variables for throwing out ($v1_{ip}$) and loading back ($v2_{ip}$) coal to enforce the operating rule which states that throwing out and loading back may never happen simultaneously. Note that the equation forces either one or both of the variables to be 0.

$$v1_{ip} + v2_{ip} \leq 1 \qquad \forall i, \quad \forall p \tag{3.3.1}$$

b. Time and sequence constraints

No additional time variables and time constraints were used to formulate the amount of coal thrown out at a point p. The reason for the decision was to minimise the amount of variables and constraints. When an amount of coal is produced by the mine and thrown out directly to outside the bunker, the same starting and finishing times that hold for the production event will also hold for the throw out event. Therefore, only the load back event's time and sequence constraints had to be modelled.

As in equation 3.2.1, the relationship between the start, finish and duration of the loading back event can be stated as follows:

$$Tf\_bl_{i,p+1} = Ts\_bl_{ip} + Dur\_bl_{ip} \qquad \forall i, \quad \forall p < p_n \tag{3.3.2}$$

To ensure that the load back event's next starting time value is always after the finish time value of the previous event, the following equation holds:

$$Ts\_bl_{ip} \geq Tf\_bl_{ip} - H(1 - v2_{ip}) \qquad \forall i, \quad \forall p \tag{3.3.3}$$

Note that the $H(1 - v2_{ip})$ term acts as a relaxation term in the linearization of equation 3.3.3 (refer to Chapter 2). It only enforces the constraint whenever the binary variable equals one (that is the event at point p is activated). This term adds slack to an otherwise strict equation whenever throw out or load back events are not activated.

To ensure the correct sequencing of the production event's starting time value with the load back event's finish time value, the following equation holds:

$$Ts\_m_{ip} \geq Tf\_bl_{ip} - H(1 - v2_{i,p-1}) \qquad \forall i, \quad \forall p > p_1 \tag{3.3.4.a}$$

Note that this equation is necessary to prevent a production event (which possibly may result in a throw out event) starting before the previous load back event (at p -1) has finished.

The reverse side of equation 3.3.4.a also holds:

$$Ts\_bl_{ip} \geq Tf\_m_{ip} - H(1 - v2_{ip}) \qquad \forall i, \quad \forall p \tag{3.3.4.b}$$

This equation is necessary to prevent a load back event (at point p) starting before the previous production event (which possibly may result in a throw out event) has finished.

As in equation 3.2.5, the finish time of an event must be less than the period's end time:

$$Tf\_bl_{i,p+1} \leq H\_m_r + H(1 - v2_{ip}) \qquad \forall i, \quad \forall p < p_n, \quad \forall per_{rp} \tag{3.3.5}$$

Again, the relaxation term adds slack to the equation.

Since the loading back event is controlled by a binary variable (which is different from the source production duration), the duration of the event is given an upper limit as follows:

$$Dur\_bl_{ip} \leq H \times v2_{ip} \qquad \forall i, \quad \forall p \tag{3.3.6}$$

The binary variable ensures that the upper limit of the duration is enforced when the event is activated ($v2_{ip} = 1$). However, when the event does not take place ($v2_{ip} = 0$), the upper limit is also forced to be 0, and thereby the duration itself has to be 0.

The duration lower limit is given by the following:

$$Dur\_bl_{ip} \geq Dur\_min \times v2_{ip} \qquad \forall i, \quad \forall p \qquad (3.3.7)$$

The binary variable ensures that the lower limit of the duration is set to a minimum value, forcing the duration to have a value higher than the minimum, when the event does indeed take place ($v2_{ip} = 1$).

c. Throw out quantity constraints

The amount of coal thrown out at a certain point p is dependant on the activation of the throw out binary variable $v1_{ip}$. When the throw out option is chosen at decision point 1 (refer to Figure 3.5), the total mine production for that specific event point p is directed to the throw out option. Therefore, the following equation holds:

$$q\_bo_{ip} = q\_m_{ip} \times v1_{ip} \qquad \forall i, \quad \forall p \qquad (3.3.8.a)$$

Note that equation 3.3.8.a is a non-linear equation which is linearized as follows (refer to Chapter 2):

$$\begin{aligned} &q\_bo_{ip} \geq q\_m_{ip} - HH(1 - v1_{ip}) \\ &q\_bo_{ip} \leq q\_m_{ip} + HH(1 - v1_{ip}) \\ &q\_bo_{ip} \leq HH \times v1_{ip} \\ &\qquad \forall i, \quad \forall p \end{aligned} \qquad (3.3.8.b)$$

When the throw out event is activated ($v1_{ip} = 1$), the first two equations forces the amount thrown out to be equal to the amount produced. However, when the event is not activated ($v1_{ip} = 0$), the first two equations becomes redundant and the third equation becomes binding, forcing the quantity variable to be 0.

d. Load back quantity constraints

The lower limit for the amount of coal loaded back at a certain point p is a function of the allocated duration from equations 3.3.2, 3.3.6 and 3.3.7, as well as the rate per front-end loader. This ensures that the load back quantity will occupy at least one front-end loader, if the event is activated:

$$q\_bl_{ip} \geq Dur\_bl_{ip} \times Rate\_bl \qquad \forall i, \quad \forall p \qquad (3.3.9)$$

The upper limit is based on the maximum amount of front-end loaders available:

$$q\_bl_{ip} \leq Dur\_bl_{ip} \times Rate\_bl \times Loaders \qquad \forall i, \quad \forall p \qquad (3.3.10)$$

A second upper limit for the amount of coal loaded back is determined by the amount of coal outside the bunker at a certain point p:

$$q\_bl_{ip} \leq Out\_b_{ip} \qquad \forall i, \quad \forall p \tag{3.3.11}$$

e. Storage constraints

The amount of coal outside the bunker at the start of the scheduling horizon is given by the user:

$$Out\_b_{ip} = Out0\_b_i \qquad \forall i, \quad p = p_1 \tag{3.3.12}$$

The material balance for the amount of coal outside the bunker can be stated as follows:

$$Out\_b_{ip} = Out\_b_{i,p-1} - q\_bl_{i,p-1} + q\_bo_{i,p-1} \qquad \forall i, \quad \forall p > p_1 \tag{3.3.13}$$

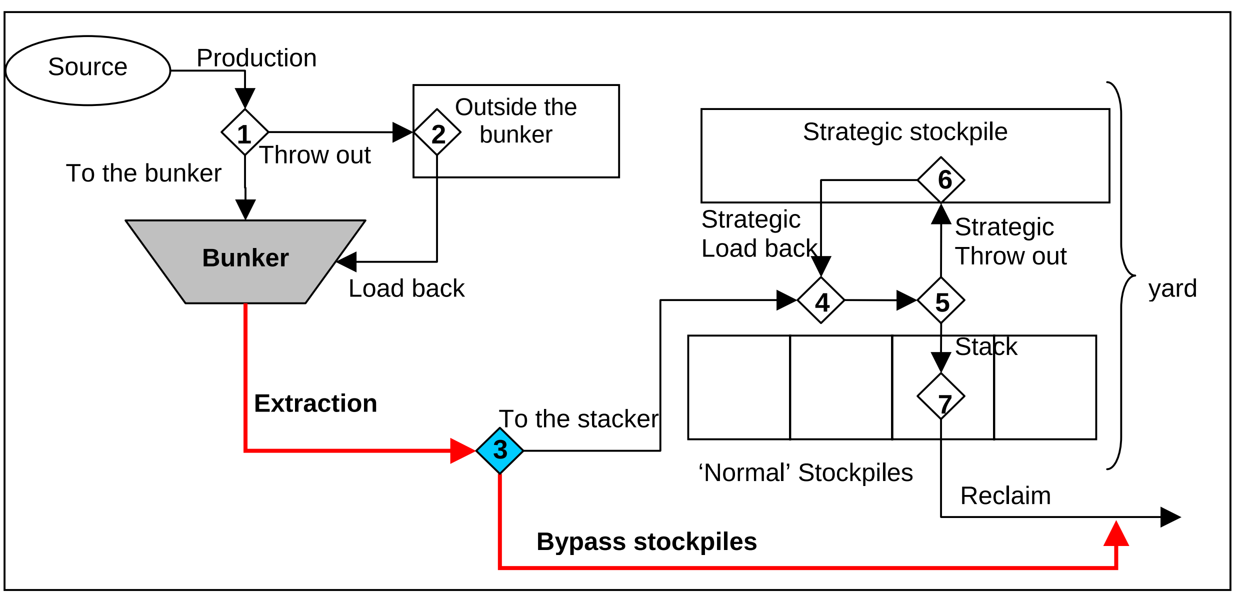
The equation states that the amount of coal outside the bunker at a point p equals the amount of coal outside the bunker at the previous point (p-1), less the amount of coal that was loaded back during the previous point (p-1), plus the amount of coal that was thrown out during the previous point (p-1).

3.3.6. Bunkers, extraction from bunkers and the bypass option

3.3.6.1. Problem statement

Referring to Figure 1.2, the boundaries for this section are illustrated in Figure 3.6.

Figure 3.6: Bunkers, extraction from bunkers and the bypass option:



Each source has its own, dedicated bunker where coal from the source is temporarily stored before it is conveyed to the stockpiles. The bunker may be seen as a continuous operation where coal can be extracted as it enters the bunker. Thus, the extraction process does not have to wait for production events to finish.

Each bunker has its own capacity. The bunkers have operating rules such as a minimum level at which extraction of coal is stopped and a maximum level at which source production is thrown outside the bunker.

Each bunker has a dedicated conveyor trajectory to the stockpile yard area. From there, the coal is directed to any of the stockpiles or it is bypassed directly to the factory.

The following infrastructure restrictions must be taken into account:

- Each conveyor trajectory has its own capacity (t/hr).
- A source can only supply to one yard j at a certain point p.
- There is only one conveyor conveying the Western mines' coal (Brandspruit and Middelbult) to the Eastern stockpile yards (yard 4, 5 and 6) (refer to Figure 1.1).
- There is only one conveyor conveying the Eastern mines' coal (Bosjespruit, Twistdraai, Syferfontein and Middlings) to the Western stockpile yards (yard 1, 2 and 3) (refer to Figure 1.1).
- Time is lost when changing a source's destination yard, because the whole conveyor trajectory is stopped and the correct route started again. Time is lost during the stop and start-up procedures.
- The bypass option at a specific yard j is available for only one mine i at a time p.

At the stockpile yards, the following decision must be taken (decision point 3): Either the coal must be conveyed to the stacker (where it can either be stacked on one of the individual stockpiles on that yard, or thrown out on the strategic stockpile) or the coal must be directed to bypass the stockpiles totally and be send to the factory directly. *Note that, if coal from one mine i is bypassed at a certain stockpile j, coal from another mine ii may still be conveyed to the stacker to be thrown out or stacked.*

The bypass option is not available for all sources due to physical infrastructure restrictions. Table 3.2 summarises the bypass infrastructure restrictions:

Note that the Western mines can only be bypassed at the Western stockpiles and the Eastern mines can only be bypassed at the Eastern stockpiles. Yard 3 and 4 and the whole Syferfontein trajectory has no bypassing infrastructure.

Table 3.2: Bypass restrictions

Source	Yards available to bypass
Brandspruit and Middelbult	Yard 1 and 2
Bosjespruit, Twistdraai and Middlings	Yard 5 and 6
Syferfontein	None

If coal is bypassed to the factory, the full stream of coal extracted from the bunkers must be bypassed, since there are no infrastructure facilities available to divide the stream into portions for stacking and bypassing.

3.3.6.2. Mathematical formulation

a. Allocation constraints - extraction

The extraction of coal from bunker i to yard j is indicated with the binary variable w_{ijp}. The following equation enforces the infrastructure restriction that a bunker's coal may only be extracted to one yard:

$$\sum_j w_{ijp} \leq 1 \qquad \forall i, \quad \forall p \tag{3.4.1}$$

The binary variable w_{ijp} is also used to enforce the restriction of one conveyor from West to East and one conveyor from East to West. With the Watters transformation, the following holds:

From West to East:

$$w_{ijp} + w_{ii,jj,p} \leq 1$$
$$\forall i \neq ii, \quad \forall j \neq jj, \quad \forall i, ii \in \{Br, Mb\}, \quad \forall j, jj \in \{y_4, y_5, y_6\}, \quad \forall p \tag{3.4.2.a}$$

$$0.5(w_{ijp} + w_{ii,jj,p}) \geq 0$$
$$\forall i \neq ii, \quad \forall j \neq jj, \quad \forall i, ii \in \{Br, Mb\}, \quad \forall j, jj \in \{y_4, y_5, y_6\}, \quad \forall p \tag{3.4.2.b}$$

From East to West:

$$w_{ijp} + w_{ii,jj,p} \leq 1$$
$$\forall i \neq ii, \quad \forall j \neq jj, \quad \forall i, ii \in \{Bo, Tw, Syf, Mdl\}, \quad \forall j, jj \in \{y_1, y_2, y_3\}, \quad \forall p \tag{3.4.3.a}$$

$$0.5(w_{ijp} + w_{ii,jj,p}) \geq 0$$
$$\forall i \neq ii, \quad \forall j \neq jj, \quad \forall i, ii \in \{Bo, Tw, Syf, Mdl\}, \quad \forall j, jj \in \{y_1, y_2, y_3\}, \quad \forall p \tag{3.4.3.b}$$

Note that the formulation will force one or both of the binary variables to be 0, ensuring that only one Western source can send coal east at a point p, and only one Eastern source can send coal west at a point p.

b. Allocation constraints – bypass option

To bypass coal from bunker i at yard j, the binary variable $x2_{ijp}$ has to be activated. The following equation enforces the infrastructure restriction that a stockpile yard's bypass option is only available for one mine i at a specific time point p:

$$\sum_i x2_{ijp} \leq 1 \qquad \forall j, \quad \forall p \tag{3.4.4}$$

However, to ensure that a bypass option is only activated if an extraction event is activated, the following equation holds:

$$x2_{ijp} \leq w_{ijp} \qquad \forall i, \quad \forall j, \quad \forall p \tag{3.4.5}$$

It is important to note that the bypassing of mine i's coal at yard j does not exclude the use of stacker j to stack or throw out another mine ii's coal simultaneously (also refer to section 3.3.8).

c. Time and sequence constraints

To sequence the extraction event to start after the start of the production event of the same point, the following equation holds:

$$Ts\_b_{ijp} \geq Ts\_m_{ip} \qquad \forall i, \quad \forall j, \quad \forall p \tag{3.4.6}$$

Equation 3.4.6 ensures that coal is not extracted before production from the source has started. The formulation also ensures a continuous operation, thus, when coal enters the bunker, it can immediately be extracted to the stockpiles.

Based on the same principles as loading coal back from outside the bunker (equations 3.3.2 to 3.3.7), the following equations hold for the start and finish times as well as the duration of the extraction events:

$$Tf\_b_{i,j,p+1} = Ts\_b_{ijp} + Dur\_b_{ijp} \qquad \forall i, \quad \forall j, \quad \forall p < p_n \tag{3.4.7}$$

$$Ts\_b_{ijp} \geq Tf\_b_{ijp} - H(1 - w_{ijp}) \qquad \forall i, \quad \forall j, \quad \forall p \tag{3.4.8}$$

$$Tf\_b_{i,j,p+1} \leq H\_m_r \qquad \forall i, \quad \forall j, \quad \forall p < p_n, \quad \forall per_{rp} \tag{3.4.9}$$

$$Dur\_b_{ijp} \leq H \times w_{ijp} \qquad \forall i, \quad \forall j, \quad \forall p \tag{3.4.10}$$

$$Dur\_b_{ijp} \geq Dur\_min \times w_{ijp} \qquad \forall i, \quad \forall j, \quad \forall p \tag{3.4.11}$$

Since the finish time value of an event starting at point p is allocated to the next point (p+1), equation 3.4.9 ensures that the finish time value does not exceed the end value of the period (refer to section 3.2).

d. Change-over time constraints

The following equations include the time lost when a source's destination yard is changed from jj to j:

$$Ts\_b_{ijp} \geq Tf\_b_{i,jj,p} + (CT\_b_{ij} \times w_{ijp}) - H(1 - w_{ijp}) \qquad \forall i, \quad \forall j \neq jj, \quad \forall p \tag{3.4.12}$$

Note that the change-over time penalty is only enforced when the binary variable is activated, otherwise it is relaxed with the $H(1 - w_{ijp})$ term, as explained previously.

e. Quantity constraints

Based on the same principles as loading back coal from outside the bunker, the following equations hold for the amount of coal conveyed from source i to yard j:

$$q\_b_{ijp} = Dur\_b_{ijp} \times Rate\_b_i \qquad \forall i, \quad \forall j, \quad \forall p \tag{3.4.13}$$

Upper limit based on the time horizon:

$$q\_b_{ijp} \leq Rate\_b_i \times H \qquad \forall i, \quad \forall j, \quad \forall p \tag{3.4.14}$$

The upper limit for the amount of coal extracted from a bunker i to any stockpile j is based on the amount of coal in the bunker:

$$q\_b_{ijp} \leq ST\_b_{ip} \qquad \forall i, \quad \forall j, \quad \forall p \tag{3.4.15}$$

f. Bypass constraints

No additional time variables and time constraints were used to formulate the amount of coal bypassed at point p. The reason for this decision was to minimise the amount of variables and constraints. When an amount of coal is extracted from a bunker and bypassed directly to the factory, the same starting and finishing times that hold for the extraction event will also hold for the bypass event. Therefore, only the following quantity constraints were added:

Brandspruit and Middelbult restrictions:

$$q\_prop_{ijp} = q\_b_{ijp} \times x2_{jp} \quad \forall i \in \{Br, Mb\}, \quad \forall j \in \{y_1, y_2\}, \quad \forall p \tag{3.4.16.a}$$

$$q\_prop_{ijp} = 0 \quad \forall i \in \{Br, Mb\}, \quad \forall j \in \{y_3, y_4, y_5, y_6\}, \quad \forall p \tag{3.4.16.b}$$

Note that equation 3.4.16.a is a non-linear equation which is linearized as follows (refer to Chapter 2):

$$\begin{aligned} & q\_prop_{ijp} \geq q\_b_{ijp} - HH(1 - x2_{ijp}) \\ & q\_prop_{ijp} \leq q\_b_{ijp} + HH(1 - x2_{ijp}) \\ & q\_prop_{ijp} \leq HH \times x2_{ijp} \\ & \forall i \in \{Br, Mb\}, \quad \forall j \in \{y_1, y_2\}, \quad \forall p \end{aligned} \tag{3.4.16.c}$$

Bosjespruit, Twistdraai and Middlings restrictions:

$$q\_prop_{ijp} = q\_b_{ijp} \times x2_{jp} \quad \forall i \in \{Bo, Tw, Mdl\}, \quad \forall j \in \{y_5, y_6\}, \quad \forall p \tag{3.4.17.a}$$

$$q\_prop_{ijp} = 0 \quad \forall i \in \{Bo, Tw, Mdl\}, \quad \forall j \in \{y_1, y_2, y_3, y_4\}, \quad \forall p \tag{3.4.17.b}$$

Similar to equation set 3.4.16.c, equation 3.4.17.a is linearized as follows (refer to Chapter 2):

$$\begin{aligned} & q\_prop_{ijp} \geq q\_b_{ijp} - HH(1 - x2_{ijp}) \\ & q\_prop_{ijp} \leq q\_b_{ijp} + HH(1 - x2_{ijp}) \\ & q\_prop_{ijp} \leq HH \times x2_{ijp} \\ & \forall i \in \{Bo, Tw, Mdl\}, \quad \forall j \in \{y_5, y_6\}, \quad \forall p \end{aligned} \tag{3.4.17.c}$$

Syferfontein restriction:

$$q\_prop_{ijp} = 0 \qquad \forall i \in \{Syf\}, \quad \forall j, \quad \forall p \tag{3.4.18}$$

Equations 3.4.16 and 3.4.17 are dependent on the activation of the bypass binary variable $x2_{ijp}$. If the bypass event is indeed activated ($x2_{ijp} = 1$), the bypass quantity variable equals the exact value of the extraction quantity variable, otherwise it is forced to be 0.

g. Storage constraints

The amount of coal in the bunker at the start of the scheduling horizon is given by the user as a percentage of the total bunker capacity:

$$ST\_b_{ip} = ST0\_b_i / 100 \times Cap\_b_i \qquad \forall i, \quad p = p_1 \tag{3.4.19}$$

The upper limit for the amount of coal in the bunker can be stated as follows:

$$ST\_b_{ip} \leq Cap\_b_i - (0.25 \times Cap\_b_i \times v1_{i,p-1}) - (0.25 \times Cap\_b_i \times v2_{i,p-1}) \qquad \forall i, \quad \forall p > p_1 \tag{3.4.20}$$

This equation forces the amount of coal in the bunker to always be less than its capacity. Note that this equation also captures the operating rules for throwing out and loading back coal (discussed in section 3.3.5.1). If coal is either thrown out or loaded back at the previous event point (p-1), the resulting amount of coal in the bunker at the next point (p) must be less than 75% of the bunker capacity.

The lower limit for the bunker level is given by the user as a fraction:

$$ST\_b_{ip} \geq Bunk\_min \times Cap\_b_i \qquad \forall i, \quad \forall p \tag{3.4.21}$$

As with the amount of coal outside the bunker, the material balance for the amount of coal in the bunker at point p can be stated as follows:

$$ST\_b_{ip} = ST\_b_{i,p-1} + q\_m_{i,p-1} - \sum_j q\_b_{i,j,p-1} + q\_bl_{i,p-1} - q\_bo_{i,p-1} \qquad \forall i, \quad \forall p > p_1 \tag{3.4.22}$$

The equation states that the amount of coal in the bunker at a point p equals the amount of coal in the bunker at the previous point (p-1); plus the amount of coal produced during the previous point (p-1); less the amount of coal extracted from the bunker to any of the stockpiles during the previous point (p-1); plus the amount of coal that was loaded back during the previous point (p-1); less the amount of coal that was thrown out during the previous point (p-1). Note that both the amount produced by the source and the amount thrown out are included, to account for decision point 1 discussed in sections 3.3.4 and 3.3.5.

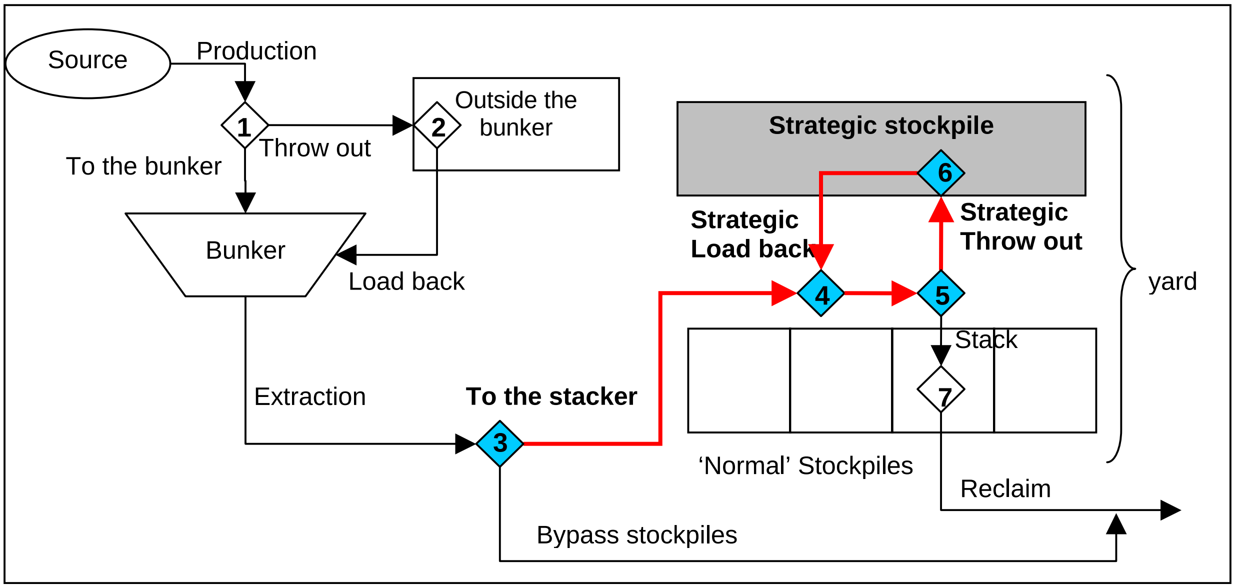
3.3.7. Strategic stockpiles, strategic throw out and load back

3.3.7.1. Problem statement

Referring to Figure 1.2, the boundaries for this section is illustrated in Figure 3.7.

As explained in Chapter 1, the purpose of the strategic stockpiles is to store coal for a longer term. The coal is then loaded back during times when the mines produce less coal, for example during Christmas and Easter. There are many actions that take place regarding throwing out and compacting the coal for longer term storage. These detail actions will not be considered for the purposes of this model.

Figure 3.7: Strategic stockpiles, strategic throw out and load back:



To throw out coal on the strategic stockpile, the route to the stacker is chosen at decision point 3 (refer to Figure 3.7). However, instead of stacking on the normal stockpiles, the stacker's boom is turned 180° to the strategic stockpile's side and the coal is thrown out there (decision point 5).

To load back coal from a strategic stockpile to one of the normal stockpiles on that yard, front-end loaders are used. The coal is loaded onto the conveyor leading to the stacker, from where it is stacked on the stockpile. Coal extracted from the bunker and coal loaded back from the strategic stockpiles may not be stacked simultaneously (decision point 4). The strategic stockpile coal now forms one of the layers in the stockpile from where it can be reclaimed as per normal reclaiming procedure (refer to Chapter 1).

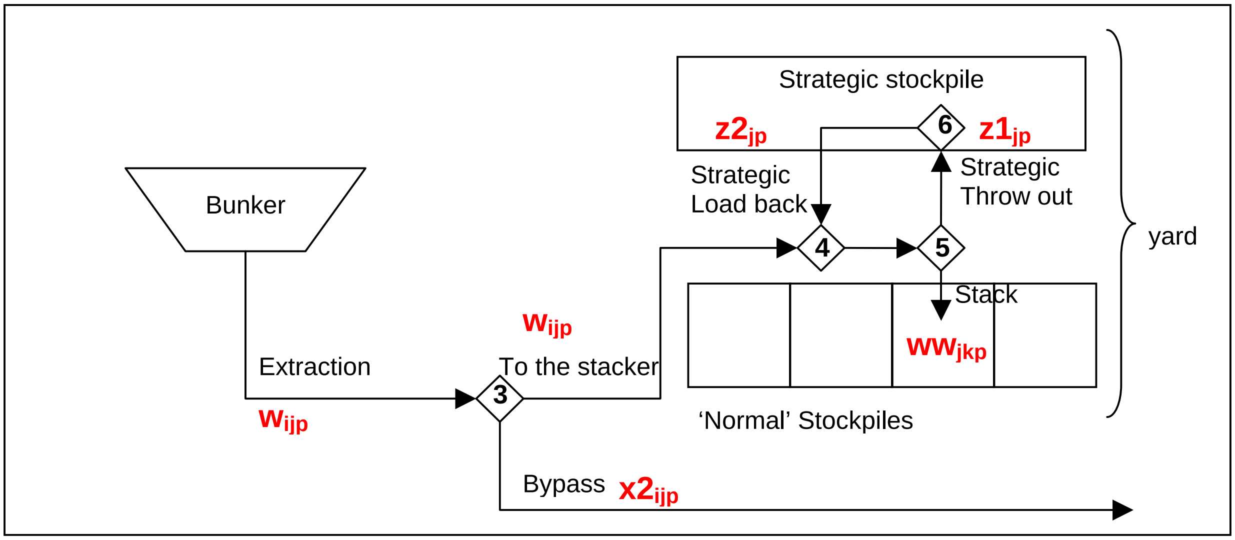
Note that it is not possible to throw out and load back strategic stockpile coal simultaneously, because the stacker is needed for both these operations, just in different directions (decision point 6).

For the purpose of this model, the different mines' contribution to the strategic stockpiles is not recorded. This approach is valid, because a specific blend limit is set for the total amount of strategic stockpile coal supplied to Synfuels at a point p. The limit is not stated in terms of the specific contributions of the different sources to the strategic stockpile. This decision simplified the strategic stockpile formulation.

3.3.7.2. Mathematical formulation

a. Allocation constraints

Figure 3.8: Relevant binary variables for decision points 4, 5 and 6:



To explain the formulation of the allocation constraints, the following binary variables are indicated on Figure 3.8:

- Binary variable for extracting coal from bunker i to stockpile yard j (w_{ijp}).
- Binary variable for bypassing coal from bunker i at stockpile yard j ($x2_{ijp}$).
- Binary variable for throwing out on the strategic stockpile ($z1_{jp}$).
- Binary variable for loading back from the strategic stockpile ($z2_{jp}$).
- Binary variable for stacking on individual stockpile k on yard j (ww_{jkp}) (also refer to section 3.3.8).

These binary variables are relevant in the formulation of decision points 4, 5 and 6. The allocation constraints will be discussed according to these decision points.

Decision point 4:

Coal may not be extracted from a bunker as well as loaded back from the strategic stockpile simultaneously. Therefore, the following equation holds:

$$z2_{jp} + \sum_i w_{ijp} \leq 1 \qquad \forall j, \quad \forall p \tag{3.5.1}$$

Equation 3.5.1 ensures that either the load back event or one of the extraction events to the specific stockpile j is activated, but not both.

Decision point 5:

Coal cannot be stacked and thrown out on the strategic stockpiles simultaneously:

$$z1_{jp} + \sum_{k} ww_{jkp} \leq 1 \qquad \forall j, \quad \forall p \tag{3.5.2}$$

Equation 3.5.1 ensures that either the throw out event or one of the stacking events on the individual stockpiles on that specific yard is activated, but not both.

Decision point 6:

Throwing out coal on the strategic stockpile and loading back coal from the strategic stockpile may not happen simultaneously:

$$z1_{jp} + z2_{jp} \leq 1 \qquad \forall j, \quad \forall p \tag{3.5.3}$$

Note that equation 3.5.3 forces either one or both of the terms to be 0 at a point p.

In addition to the three constraints stated above, the following two constraints are necessary to link the different coal flow options to each other:

To ensure that coal is only thrown out when coal is extracted from a bunker to that specific stockpile yard, the following equation holds:

$$z1_{jp} \leq \sum_{i} w_{ijp} \qquad \forall j, \quad \forall p \tag{3.5.4}$$

Equation 3.5.4 ensures that the upper limit for $z1_{jp}$ is 0 when no coal is extracted towards the specific yard j.

Similarly, to ensure that a stacking event is activated whenever coal is loaded back from the strategic stockpile:

$$z2_{jp} \leq \sum_{k} ww_{jkp} \qquad \forall j, \quad \forall p \tag{3.5.5}$$

Equation 3.5.5 ensures that the upper limit for $z2_{jp}$ is 0 when no stacking event on the specific yard j is activated.

Table 3.3 summarises the different coal flow options with the relevant binary variables.

Table 3.3: Summary of strategic stockpile allocation equation results

	Scenario at a specific yard j:	$\sum_i w_{ijp}$	$z1_{jp}$	$z2_{jp}$	$\sum_k ww_{jkp}$
1	Coal extracted from one of the bunkers stacked on one of the normal stockpiles	1	0	0	1
2	Coal extracted from one of the bunkers thrown out on the strategic stockpile	1	1	0	0
3	Coal loaded back from the strategic stockpile and stacked on one of the normal stockpiles (no extraction)	0	0	1	1

b. Time and sequence constraints – Strategic stockpile load back

Based on the same principles as loading coal back from outside the bunker (equations 3.3.2 to 3.3.7), the following equations hold for the start and finish times as well as the duration of the strategic stockpile coal load back events. Note that the duration of the strategic stockpile load back event is allocated to a specific individual stockpile k in equation 3.5.10.

$$Tf\_sl_{j,p+1} = Ts\_sl_{jp} + \sum_k Dur\_sl_{jkp} \qquad \forall j, \quad \forall p < p_n \qquad (3.5.6)$$

$$Ts\_sl_{jp} \geq Tf\_sl_{jp} - H(1 - z2_{jp}) \qquad \forall j, \quad \forall p \qquad (3.5.7)$$

$$Tf\_sl_{j,p+1} \leq H\_m_r \qquad \forall j, \quad \forall p < p_n, \quad \forall per_{rp} \qquad (3.5.8)$$

$$\sum_k Dur\_sl_{jkp} \leq H \times z2_{jp} \qquad \forall j, \quad \forall p \qquad (3.5.9)$$

$$Dur\_sl_{jkp} \leq H \times ww_{jkp} \qquad \forall j, \quad \forall k, \quad \forall p \qquad (3.5.10)$$

$$\sum_k Dur\_sl_{jkp} \geq Dur\_min \times z_{jp} \qquad \forall j, \quad \forall p \qquad (3.5.11)$$

To ensure the correct sequencing of an extraction event's starting time value with the strategic stockpile load back event's finish time value, the following equation holds:

$$Ts\_b_{ijp} \geq Tf\_sl_{jp} - H(1 - z2_{j,p-1}) \qquad \forall i, \quad \forall j, \quad \forall p > p_1 \qquad (3.5.12.a)$$

Note that this equation is necessary to prevent an extraction event starting before the previous strategic stockpile load back event (at p -1) has finished. The reverse side of equation 3.5.12.a also holds:

$$Ts\_sl_{jp} \geq Tf\_b_{ijp} - H(1 - z2_{ip}) \qquad \forall i, \quad \forall j, \quad \forall p \qquad (3.5.12.b)$$

This equation prevents a strategic stockpile load back event (at point p) starting before the previous extraction event has finished.

c. Time and sequence constraints – Strategic stockpile throw out

As with bunker throw out and bypassing events, no additional time variables and time constraints were used to formulate the amount of coal thrown out on the strategic stockpiles at any given point p. The reason for this decision was to minimise the amount of variables and constraints. When an amount of coal is extracted from a bunker and thrown out on a strategic stockpile, the same starting and finish times that hold for the extraction event would also hold for the throwing out event.

d. Quantity constraints – Strategic stockpile load back

Based on the same principles as loading back coal from outside the bunker (equations 3.3.9 to 3.3.11), the following equations hold for the amount of coal loaded back from the strategic stockpiles.

The lower limit for the amount of coal loaded back from the strategic stockpile at a certain point p is a function of the allocated duration from equations 3.5.9, 3.5.10 and 3.5.11, as well as the rate per front-end loader. This ensures that the load back quantity will occupy at least one front-end loader, if the event is activated.

$$q\_sl_{jkp} \geq Dur\_sl_{jkp} \times Rate\_bl \qquad \forall j, \quad \forall k, \quad \forall p \tag{3.5.13}$$

The upper limit is based on the maximum amount of front-end loaders available:

$$q\_sl_{jkp} \leq Dur\_sl_{jkp} \times Rate\_bl \times Loaders \qquad \forall j, \quad \forall k, \quad \forall p \tag{3.5.14}$$

A second upper limit for the amount of coal loaded back from the strategic stockpiles is determined by the amount of coal on the strategic stockpile at a certain point p:

$$q\_sl_{jkp} \leq Strat_{jp} \qquad \forall j, \quad \forall k, \quad \forall p \tag{3.5.15}$$

The upper limit for the total amount of coal handled with front-end loaders during a period is based on the number of front-end loaders available:

$$\sum_{i,\forall per_{rp}} q\_bl_{ip} + \sum_{j,k,\forall per_{rp}} q\_sl_{jkp} \leq rate\_bl \times Loaders \times (H\_m_r - Ts0\_m_r) \qquad \forall r \tag{3.5.16}$$

The following equation ensures that the capacity limit of the stacker is not exceeded:

$$q\_sl_{jkp} \leq rate\_s \times Dur\_sl_{jkp} \qquad \forall j, \quad \forall k, \quad \forall p \qquad (3.5.17)$$

e. Quantity constraints – Strategic stockpile throw out

The quantity of coal that can be thrown out on the strategic stockpile can be formulated as follows:

$$\begin{aligned} q\_so_{jp} &\leq \sum_i q\_b_{ijp} - \sum_i q\_prop_{ijp} + HH(1 - z1_{jp}) \qquad \forall j, \quad \forall p \\ q\_so_{jp} &\geq \sum_i q\_b_{ijp} - \sum_i q\_prop_{ijp} - HH(1 - z1_{jp}) \qquad \forall j, \quad \forall p \end{aligned} \qquad (3.5.18.a)$$

The set of equations above forces the quantity of coal thrown out at a strategic stockpile to be equal to the amount extracted from a mine's bunker to that yard, less the amount bypassed directly to the factory. However, this can only be true if the strategic stockpile throw out binary variable $z1_{jp}$ is activated.

If the set of equations in 3.5.17.a is relaxed (the binary variable is not activated), the following equation sets the upper limit for q_so_{jp} equal to 0:

$$q\_so_{jp} \leq HH \times z1_{jp} \qquad \forall j, \quad \forall p \qquad (3.5.18.b)$$

f. Storage constraints

The amount of coal on the strategic stockpile at the start of the scheduling horizon is given by the user:

$$Strat_{jp} = Strat0_j \qquad \forall j, \quad p = p_1 \qquad (3.5.19)$$

The material balance for the amount of coal on the strategic stockpile can be stated as follows:

$$Strat_{jp} = Strat_{j,p-1} - \sum_k q\_sl_{j,k,p-1} + q\_so_{j,p-1} \qquad \forall j, \quad \forall p > p_1 \qquad (3.5.20)$$

As before, the amount of coal on the strategic stockpile j at point p is determined by the amount of coal on the strategic stockpile at the previous point (p-1), less the total amount of coal loaded back to the individual stockpiles, plus the amount of coal thrown out on the strategic stockpile.

The upper limit for the amount of coal on a strategic stockpile is that specific stockpile's capacity:

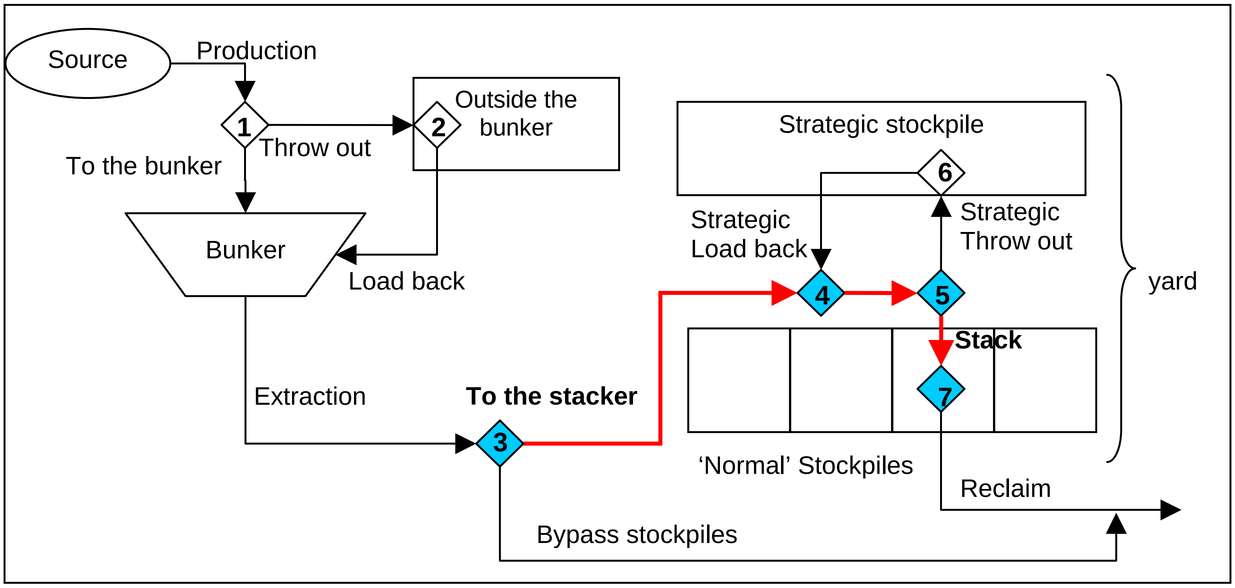
$$Strat_{jp} \leq Cap\_strat_j \qquad \forall j, \quad \forall p \qquad (3.5.21)$$

3.3.8. Stacking

3.3.8.1. Problem statement

Referring to Figure 1.2, the boundaries for this section is illustrated in Figure 3.9. Note that the stockpile philosophy is discussed separately in section 3.3.10.

Figure 3.9: Stacking:



Coal extracted from a mine's bunker, is conveyed to a stockpile yard. At decision point 3 the coal can either be conveyed to the stacker or be directed to bypass the stockpiles totally. If the coal is conveyed to the stacker, it may be stacked on one of the individual stockpiles or it may be thrown out (decision point 5). Note that coal extracted from the mine bunkers may not be stacked simultaneously with strategic stockpile coal that is loaded back (decision point 4). This section will only focus on the stacking action.

The following infrastructure restrictions must be taken into account:

- There is only one stacker per stockpile yard.
- Only one source's coal can be stacked at a point p. *Note that, if coal from one mine i is bypassed at a certain stockpile j, coal from another mine ii may still be conveyed to the stacker to be thrown out or stacked.*
- Time is lost when a stacker changes position from one stockpile to another on the same yard. The rails of the stacker are situated just outside the yard, which enable the stacker to move from one side to the other without hindrances caused by the reclaimer or other stockpiles that might be situated in between. (Compare to the reclaimer position discussed in section 3.3.10.1)

A very important fact when formulating the stacking constraints is that each source's coal has its own quality properties, which may differ from the properties of another source's coal (refer to Chapter 1). Also, the blend limits given by Synfuels, are stated in terms of the percentage contribution from each source in the final coal blend sent to the factory. Therefore, it is important to keep track of the amount of coal from each source that is stacked on a specific stockpile.

The stockpile philosophy will be discussed in section 3.3.10.

3.3.8.2. Binary variables

To allocate a specific source's coal to a specific stockpile, on a specific yard, at a specific time point, the following example binary variable will be needed:

$$XX_{ijkp} \qquad \forall i, \quad \forall j, \quad \forall k, \quad \forall p \tag{3.6.1}$$

This binary variable results in 1728 single binary variables.

Since one of the model development principles is to reduce binary variables, this variable is divided into two separate variables:

- The first binary variable w_{ijp} is used to activate extraction from a certain bunker i to a certain yard j, at a point p.
- The second binary variable ww_{jkp} allocates the coal that was sent to yard j, to a specific stockpile k, at a point p.

As a result of the formulation explained above, the number of single binary variables is reduced from 1728 to 720. The result is illustrated in Table 3.4:

Table 3.4: Comparison of binary variable options:

	Option A	**Option B**	
	XX_{ijkp}	w_{ijp}	ww_{jkp}
6 Sources i	√	√	
6 Yards j	√	√	√
4 Individual stockpiles k	√		√
12 Event points p	√	√	√
Total single variables	1728	432	288
Total per option	**1728**	**720**	

3.3.8.3. Mathematical formulation

a. Allocation constraints

As explained above, there is only one stacker per stockpile, thus only one point of stacking at a yard. To comply with this restriction, equation 3.6.2 holds:

$$\sum_{k} ww_{jkp} \leq 1 \qquad \forall j, \quad \forall p \tag{3.6.2}$$

To ensure that either the bypass event, or the stacking event, or the throw out event is activated when coal is extracted from the bunker, equation 3.6.3 is stated as follows:

$$\sum_{i} w_{ijp} = \sum_{i} x2_{ijp} + \sum_{k} ww_{jkp} + z1_{jp} \qquad \forall j, \quad \forall p \tag{3.6.3}$$

Note that equation 3.5.2 ensures that either the stacking event or the strategic stockpile throw out event is activated, but not both. However, it is possible to use the stacker for one of these events while another mine's coal is bypassed at the same yard. Thus, the maximum limit for equation 3.6.3 is two (a possible bypass event and one of the stacker events activated).

b. Quantity constraints

The quantity variable for stacking has four indices to track the amount from every source that was stacked on a specific stockpile k on yard j. Similar to the strategic stockpile throw out event, the quantity of coal stacked is a result of the following equation, and not a result of the allocated duration as with previous discussions:

$$\begin{gathered} q\_s_{ijkp} \geq q\_b_{ijp} - q\_prop_{ijp} - H(2 - ww_{jkp} - w_{ijp}) \\ q\_s_{ijkp} \leq q\_b_{ijp} - q\_prop_{ijp} + H(2 - ww_{jkp} - w_{ijp}) \\ \forall i, \quad \forall j, \quad \forall k, \quad \forall p \end{gathered} \tag{3.6.4}$$

In this set of equations a coal balance is calculated. Coal which is extracted from a bunker may be bypassed or conveyed to the stacker. Either way, the full amount of coal must be directed to the coal flow route that was chosen. Therefore, the amount of coal bypassed directly to the factory is subtracted from the amount of coal extracted from the bunker. Once this amount is established, the set of equations is only enforced if both the extraction event ($w_{ijp} = 1$) and the stacking event at that specific stockpile is activated ($ww_{jkp} = 1$). If these binary variables are not activated, the set of equations is relaxed and the following equation enforces the upper limit (which for this case will be 0):

$$q\_s_{ijkp} \leq HH \times ww_{jkp} \qquad \forall i, \quad \forall j, \quad \forall k, \quad \forall p \tag{3.6.5}$$

The following equation ensures that the specific mine's stacking quantity is set to 0 if no coal is extracted from that mine to the specific stockpile yard j:

$$q\_s_{ijkp} \leq HH \times w_{ijp} \qquad \forall i, \quad \forall j, \quad \forall k, \quad \forall p \tag{3.6.6}$$

The following equation ensures that the capacity limit of the stacker is not exceeded:

$$\sum_k q\_s_{ijkp} \leq rate\_s \times Dur\_b_{ijp} \qquad \forall i, \quad \forall j, \quad \forall p \tag{3.6.7}$$

c. Time and sequence constraints

Similar to the application of binary variables, a separate set of timing and duration equations are defined for stacking. This is done to keep track of the specific individual stockpile where the stacker is stacking for timing and change-over purposes. For this reason, the stacking time constraints are dependent on both the ‘extraction with stacking’ and the strategic stockpile load back events. Since both of these events play a role in the stacker’s exact position on the stockpile yard, the following equations hold:

Extraction synchronization:

$$\begin{gathered} Ts\_s_{jkp} \geq Ts\_b_{ijp} - H(2 - ww_{jkp} - w_{ijp} + x2_{ijp}) \\ Ts\_s_{jkp} \leq Ts\_b_{ijp} + H(2 - ww_{jkp} - w_{ijp} + x2_{ijp}) \\ \forall i, \quad \forall j, \quad \forall k, \quad \forall p \end{gathered} \tag{3.6.8}$$

The set of equations above ensures that the extraction and stacking starting times are equal if coal is indeed extracted to a specific stockpile k on yard j. The bypass binary variable $x2_{ijp}$ is added to the relaxation term to ensure that the constraint is not enforced when the extracted coal is bypassed directly to the factory.

Strategic stockpile load back synchronization:

$$\begin{gathered} Ts\_s_{jkp} \geq Ts\_sl_{jp} - H(2 - ww_{jkp} - z2_{jp}) \\ Ts\_s_{jkp} \leq Ts\_sl_{jp} + H(2 - ww_{jkp} - z2_{jp}) \\ \forall j, \quad \forall k, \quad \forall p \end{gathered} \tag{3.6.9}$$

The set of equations aboveforces the stacking starting time value of the specific individual stockpiles to be equal to the starting time value of the strategic stockpile load back event, if the event is indeed activated ($z2_{jp} = 1$). Otherwise, the equations are relaxed.

As before, the following duration equation holds:

$$Tf\_s_{j,k,p+1} = Ts\_s_{jkp} + Dur\_s_{jkp} \qquad \forall j, \quad \forall k, \quad \forall p < p_n \tag{3.6.10}$$

Since the binary variables w_{ijp} and $z2_{jp}$ ensures that extracting and stacking coal from the mines and the strategic stockpile load back events will not happen simultaneously, the durations of these events may be added as follows:

$$Dur\_s_{jkp} = \sum_i (q\_s_{ijkp} / rate\_b_i) + Dur\_sl_{jkp} \qquad \forall j, \quad \forall k, \quad \forall p \tag{3.6.11}$$

The following equation sets the upper bound for the finish time of a stacking event, according to the period end times, as above:

$$Tf\_s_{j,k,p+1} \le H\_m_r \qquad \forall j, \quad \forall k, \quad \forall p < p_n, \quad \forall per_{rp} \tag{3.6.12}$$

d. Change-over time constraints

The following equations include the time lost when a stacker's position is changed from one stockpile k to another stockpile kk:

$$Ts\_s_{j,kk,p} \ge Tf\_s_{jkp} + (CT\_s_{k,kk} \times ww_{j,kk,p}) - H(2 - ww_{j,kk,p} - ww_{j,k,p-1}) \qquad \forall j, \quad \forall k, kk, \quad \forall p > p_1 \tag{3.6.13.a}$$

Note that the change-over time penalty is only enforced when the stacking binary variable is activated for stockpile kk at point p. The relaxation term ensures that the stacker was stacking at a stockpile k at the previous point p-1, and that the stacker is activated to be stacking at another stockpile kk at the current point p.

The stacker's position at the start of the scheduling horizon is given by the user. To ensure that change-over time is taken into account also for the first event point, the following equation holds:

$$Ts\_s_{j,kk,p} \ge Tf\_s_{jkp} + (CT\_s_{k,kk} \times ww_{j,kk,p}) - H(2 - ww_{j,kk,p} - Pos0\_s_{jk}) \qquad \forall j, \quad \forall k, kk, \quad \forall p = p_1 \tag{3.6.13.b}$$

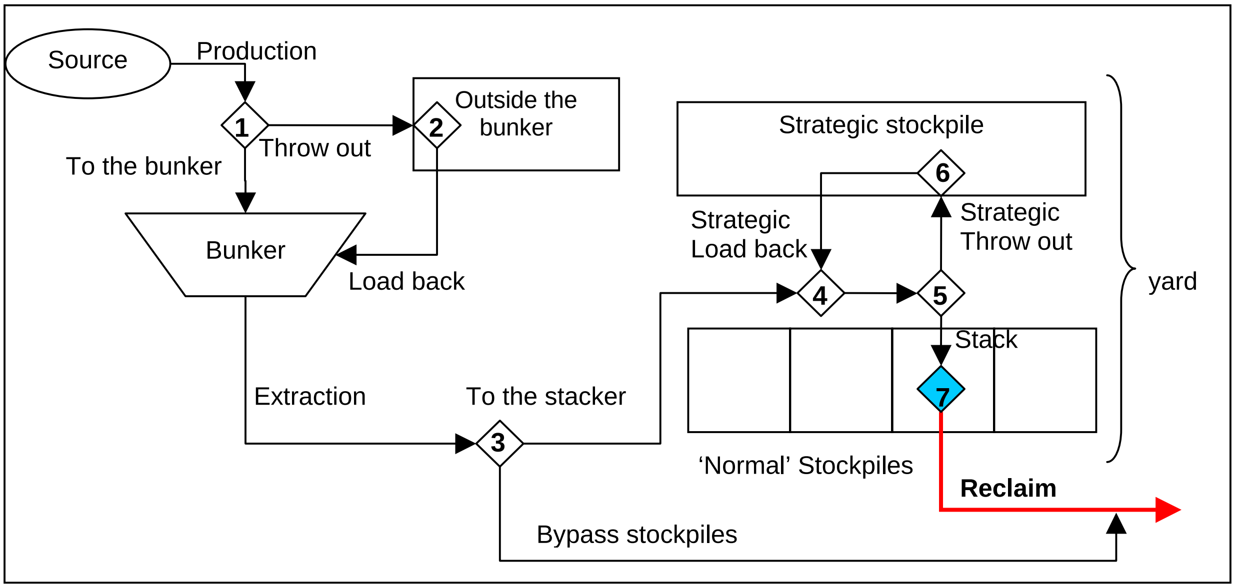
Note that the initial position of the stacker is used in the relaxation term, instead of the stacking event of the previous point (p-1).

3.3.9. Reclaiming

3.3.9.1. Problem statement

Referring to Figure 1.2, the boundaries for this section is illustrated in figure 3.10. Coal is reclaimed from the normal stockpiles and conveyed to Synfuels to supply the demand for coal, as explained in Chapter 1.

Figure 3.10: Reclaiming:



The following infrastructure restrictions have to be enforced:

- There is only one reclaimer per stockpile yard.
- Reclaiming speeds can be adjusted, and need not be at the maximum.
- A stockpile cannot simultaneously be stacked and reclaimed (see stockpile process discussion in Chapter 1).
- Bypassing and reclaiming may happen simultaneously. When bypassing coal directly from the mine bunkers to the factory, the reclaimed coal and the bypassed coal share the conveyor space. Therefore, the total amount should not exceed the capacity of the conveyor.
- Yards 1, 2 and 3 only supply the Western factory. Yards 4, 5 and 6 only supply the Eastern factory.
- Time is lost when the reclaimer changes position from one stockpile to another on the same yard. However, the reclaimer is positioned within the yard, which prevents it from passing stockpiles that may be situated between its current position and its destination position. For example, if the reclaimer finished reclaiming a stockpile on position 1 in the yard, it will not be able to move to stockpile 3 if there is another stockpile situated at position 2 (compare to stacker position discussion in section 3.3.8.1).

Note that coal may be reclaimed from more than one yard simultaneously at a point p to supply the demand of the factory. The total amount of coal sent to the Western or the Eastern factories at a time may not exceed the maximum capacity of the factory conveyors.

As explained in section 3.3.8.1, it is important to keep record of the amount of coal from each source in the total supply to Synfuels at each point p. This enables the scheduling model to optimise according to the blend limits and the blend plan as given by Synfuels. The amount of strategic stockpile coal reclaimed from the individual stockpiles is tracked separately.

Note that only the blend limits were considered for the first phase's development of the scheduling model. The blend plan is included in the improved model discussed in Chapter 4.

3.3.9.2. Mathematical formulation

a. Allocation constraints

Either stacking or reclaiming can be activated at a specific stockpile, but not both. Binary variables for stacking (ww_{jkp}) and reclaiming (x_{jkp}) are used to enforce the following restriction:

$$ww_{jkp} + x_{jkp} \leq 1 \qquad \forall j, \quad \forall k, \quad \forall p \qquad (3.7.1)$$

Note that either one or both the binary variables are forced to be 0.

As explained above, there is only one reclaimer per stockpile, thus only one point of reclaiming at a yard. To comply with this restriction, equation 3.7.2 states:

$$\sum_k x_{jkp} \leq 1 \qquad \forall j, \quad \forall p \qquad (3.7.2)$$

b. Time and sequence constraints

To ensure that the starting time value of the reclaiming event is sequenced with the periods, it has to be equal to or after the starting time values of the different periods:

$$Ts\_r_{jkp} \geq Ts0\_m_r \qquad \forall j, \quad \forall k, \quad \forall per_{rp} \qquad (3.7.3)$$

The starting time value of the reclaiming event at point p must be after the finish time value of the reclaiming event at the previous point:

$$Ts\_r_{jkp} \geq Tf\_r_{jkp} \qquad \forall j, \quad \forall k, \quad \forall p \qquad (3.7.4)$$

The bypassing event does not have its own time indicators, but uses the extraction event's time variables (refer to section 3.3.6.2). Bypassing and reclaiming must be sequenced per Western and Eastern sides, because the conveyor capacity is shared and may not be exceeded (refer to section 3.3.9.1). To enable sequencing, the following set of equations holds:

West:

$$Ts\_r_{jkp} \geq Ts\_b_{i,jj,p} - H(2 - x_{jkp} - x2_{i,jj,p})$$
$$Ts\_r_{jkp} \leq Ts\_b_{i,jj,p} + H(2 - x_{jkp} - x2_{i,jj,p}) \quad (3.7.5.a)$$
$$\forall i, \quad \forall j, jj \in \{y_1, y_2, y_3\}, \quad \forall k, \quad \forall p$$

East:

$$Ts\_r_{jkp} \geq Ts\_b_{i,jj,p} - H(2 - x_{jkp} - x2_{i,jj,p})$$
$$Ts\_r_{jkp} \leq Ts\_b_{i,jj,p} + H(2 - x_{jkp} - x2_{i,jj,p}) \quad (3.7.5.b)$$
$$\forall i, \quad \forall j, jj \in \{y_4, y_5, y_6\}, \quad \forall k, \quad \forall p$$

Equations 3.7.5.a and 3.7.5.b only become binding when both the reclaiming event and the extraction event are activated for point p. Note that the reclaiming event may be activated for a different stockpile yard j than the bypassing event's yard jj. When both events are activated simultaneously on one side (Eastern or Western side), the reclaiming starting time value is equal to the bypass event starting time value. Otherwise, the equations are relaxed.

The reclaiming event on a specific stockpile (point p) must also be sequenced with the stacking event (previous point p-1) on the same stockpile to ensure that coal is not reclaimed before the stacking process is finished:

$$Ts\_r_{jkp} \geq Tf\_s_{jkp} - H(2 - x_{jkp} - ww_{j,k,p-1}) \qquad \forall j, \quad \forall k, \quad \forall p > p_1 \quad (3.7.6.a)$$

The reverse of equation 3.7.6.a also holds. The stacking event on a specific stockpile (point p) must be sequenced with the reclaiming event (previous point p-1) on the same stockpile to ensure that coal is not stacked before the reclaiming process is finished:

$$Ts\_s_{jkp} \geq Tf\_r_{jkp} - H(2 - x_{j,k,p-1} - ww_{jkp}) \qquad \forall j, \quad \forall k, \quad \forall p > p_1 \quad (3.7.6.b)$$

Coal may be reclaimed from more than one reclaimer at a time to supply the factory's demand. Therefore, the reclaiming timing variables between the yards supplying to the factory (Western and Eastern sides separately) must be sequenced. The following set of equations forces the starting time values for the reclaiming events on different yards to coincide:

West:

$$Ts\_r_{jkp} \geq Ts\_r_{jj,kk,p} - H(2 - x_{jkp} - x_{jj,kk,p})$$
$$Ts\_r_{jkp} \leq Ts\_r_{jj,kk,p} + H(2 - x_{jkp} - x_{jj,kk,p}) \quad (3.7.7.a)$$
$$\forall j, jj \in \{y_1, y_2, y_3\}, \quad \forall j \neq jj, \quad \forall k, kk, \quad \forall p$$

East:

$$Ts\_r_{jkp} \geq Ts\_r_{jj,kk,p} - H(2 - x_{jkp} - x_{jj,kk,p})$$
$$Ts\_r_{jkp} \leq Ts\_r_{jj,kk,p} + H(2 - x_{jkp} - x_{jj,kk,p}) \qquad (3.7.7.b)$$
$$\forall j,\ jj \in \{y_4, y_5, y_6\}, \quad \forall j \neq jj, \quad \forall k, kk, \quad \forall p$$

Similar to the stacking constraints (refer to section 3.3.8), the following constraints hold:

$$Tf\_r_{j,k,p+1} \leq H\_m_r \qquad \forall j, \quad \forall k, \quad \forall p < p_n, \quad \forall per_{rp} \qquad (3.7.8)$$

$$Tf\_r_{j,k,p+1} = Ts\_r_{jkp} + Dur\_r_{jkp} \qquad \forall j, \quad \forall k, \quad \forall p < p_n \qquad (3.7.9)$$

$$Dur\_r_{jkp} \leq H \times x_{jkp} \qquad \forall j, \quad \forall k, \quad \forall p \qquad (3.7.10)$$

$$Dur\_r_{jkp} \geq Dur\_min \times x_{jkp} \qquad \forall j, \quad \forall k, \quad \forall p \qquad (3.7.11)$$

c. Reclaimer position and Change-over constraints

As explained in section 3.3.9.1 above, the reclaimer can physically not change position if there is another stockpile between its current position and its destination position. Even if a portion of the stockpile at the reclaimer's current position is still remaining, it will prevent the reclaimer from changing to another stockpile.

c.1. Reclaimer Position:

Before the change-over constraints can be formulated, the position of the reclaimer needs to be tracked at every event point, even if no reclaiming event took place. Note that the reclaimer position variable that is used (Pos_r_{jkp}) is not a binary variable, but with the proposed formulation, it can only take a value of 1 or 0. The following equations represent the different reclaiming tracking activities on a yard j:

1. When the reclaiming event at position k is activated ($x_{jkp} = 1$), the tracking variable is set to one:

$$Pos\_r_{jkp} \geq 1 - H(1 - x_{jkp})$$
$$Pos\_r_{jkp} \leq 1 + H(1 - x_{jkp}) \qquad (3.7.12)$$
$$\forall j, \quad \forall k, \quad \forall p$$

2. When no reclaiming event takes place on the yard, the position variable keeps its previous value. In this instance, the position variable acts as *a 'memory' variable*. These equations do not hold for the first event point.

$$Pos\_r_{jkp} \geq Pos\_r_{j,k,p-1} - H\sum_{kk} x_{j,kk,p}$$
$$Pos\_r_{jkp} \leq Pos\_r_{j,k,p-1} + H\sum_{kk} x_{j,kk,p} \qquad (3.7.13)$$
$$\forall j, \quad \forall k, \quad \forall p > p_1$$

3. To account for the first event point which is excluded in equation 3.7.13 above, the following set of equations take into account the original position of the reclaimer as given by the user:

$$Pos\_r_{jkp} \geq Pos0\_r_{jk} - H\sum_{kk} x_{j,kk,p}$$
$$Pos\_r_{jkp} \leq Pos0\_r_{jk} + H\sum_{kk} x_{j,kk,p} \qquad (3.7.14)$$
$$\forall j, \quad \forall k, \quad \forall p = p_1$$

4. To set the position value at a certain position k to 0 if reclaiming takes place at any other position kk:

$$Pos\_r_{jkp} \geq -H(1-\sum_{kk \neq k} x_{j,kk,p})$$
$$Pos\_r_{jkp} \leq H(1-\sum_{kk \neq k} x_{j,kk,p}) \qquad (3.7.15)$$
$$\forall j, \quad \forall k, \quad \forall p > p_1$$

5. Lastly, only one position variable on a stockpile yard j may have a value:

$$\sum_{k} Pos\_r_{jkp} \leq 1 \qquad \forall j, \quad \forall p \qquad (3.7.16)$$

c.2. Change possibility:

In the following equations the heap lengths of the stockpiles that might be in the reclaimer's way are used to determine whether the reclaimer is allowed to change to the assessed destination position. If the heap lengths equal 0, no stockpile is in the way and the change is possible.

In the following equations, the following sets are used:

- Set k: The current position of the reclaimer
- Set kk: The destination position of the reclaimer
- Set kkk: All possible positions between k and kk, including k.

The following scenarios are formulated:

1. k ≤ kkk < kk : The current position value k of the reclaimer is smaller than the destination position value kk:

$$x_{jkp} \leq 1 - 0.01\left(\sum_{kkk<kk \ AND \ kkk \geq k} heapl\_s_{j,kkk,p}\right) + HH(1 - Pos\_r_{j,k,p\text{-}1}) \qquad \forall j, \quad \forall k < kk \quad \forall p > p_1 \tag{3.7.17.a}$$

For the first event point, the reclaimer's original position is used:

$$x_{jkp} \leq 1 - 0.01\left(\sum_{kkk<kk \ AND \ kkk \geq k} heapl\_s_{j,kkk,p}\right) + HH(1 - Pos0\_r_{jk}) \qquad \forall j, \quad \forall k < kk \quad \forall p = p_1 \tag{3.7.17.b}$$

2. k ≥ kkk > kk : The current position value k of the reclaimer is bigger than the destination position value kk:

$$x_{jkp} \leq 1 - 0.01\left(\sum_{kkk>kk \ AND \ kkk \leq k} heapl\_s_{j,kkk,p}\right) + HH(1 - Pos\_r_{j,k,p\text{-}1}) \qquad \forall j, \quad \forall k > kk \quad \forall p > p_1 \tag{3.7.18.a}$$

For the first event point, the reclaimer's original position is used:

$$x_{jkp} \leq 1 - 0.01\left(\sum_{kkk>kk \ AND \ kkk \leq k} heapl\_s_{j,kkk,p}\right) + HH(1 - Pos0\_r_{jk}) \qquad \forall j, \quad \forall k > kk \quad \forall p = p_1 \tag{3.7.18.b}$$

It is important to note that the principle used in formulating these equations is that x_{jkp} is a binary variable. Therefore, the sum of the heap lengths between k and kk is multiplied with an arbitrary small number (0.01 in this case). If any heap length exists between k and kk, it is reduced to a number smaller than 1 and subtracted from 1, to ensure that the x_{jkp} binary variable cannot have the value of 1, thus ensuring that no reclaiming can take place at the assessed destination position kk.

c.3. Change-over time lost

Similar to the stacking change-over time (refer to equation 3.6.13), the following equations accommodate the time lost when the reclaimer changes from position k to position kk:

$$Ts\_r_{j,kk,p} \geq Tf\_r_{jkp} + (CT\_r_{k,kk} \times x_{j,kk,p}) - H(2 - x_{j,kk,p} - Pos\_r_{j,k,p\text{-}1}) \qquad \forall j, \quad \forall k \neq kk, \quad \forall p > p_1 \tag{3.7.19.a}$$

$$Ts\_r_{j,kk,p} \geq Tf\_r_{jkp} + (CT\_r_{k,kk} \times x_{j,kk,p}) - H(2 - x_{j,kk,p} - Pos0\_r_{jk}) \qquad \forall j, \quad \forall k \neq kk, \quad \forall p = p_1 \tag{3.7.19.b}$$

d. Quantity constraints

The amount of coal reclaimed from a stockpile consists of the layers of coal from the mines and other layers of coal loaded back from the strategic stockpile. To set the upper limits for these variables, the following equations are used:

Upper limit based on the amount of coal on the stockpile from each individual source:

$$q\_r_{ijkp} \leq ST\_sm_{ijkp} \qquad \forall i, \quad \forall j, \quad \forall k, \quad \forall p \tag{3.7.20}$$

Upper limit based on the amount of coal on the stockpile loaded back from the strategic stockpile:

$$q\_rsl_{jkp} \leq ST\_sl_{jkp} \qquad \forall j, \quad \forall k, \quad \forall p \tag{3.7.21}$$

As explained in section 3.3.9.1, coal may be reclaimed from different reclaimers simultaneously to supply the factory demand, and reclaiming rates need not be at the maximum. It may therefore happen that one reclaimer reclaims at 2000 t/hr and the other at 1500 t/hr to supply the factory of 3500 t/hr. It is therefore necessary to set upper and lower limits to the reclaiming amounts:

Lower limit for reclaiming amount:

$$\sum_i q\_r_{ijkp} + q\_rsl_{jkp} \geq Rate\_r \times Dur\_min \times x_{jkp} \qquad \forall j, \quad \forall k, \quad \forall p \tag{3.7.22}$$

Upper limit for reclaiming amount:

$$\sum_i q\_r_{ijkp} + q\_rsl_{jkp} \leq Rate\_r \times Dur\_r_{jkp} \qquad \forall j, \quad \forall k, \quad \forall p \tag{3.7.23}$$

When coal from the mine is bypassed at yard j, it shares conveyor space with the reclaimed coal. Therefore, the bypassed coal is also taken into account when setting the upper limit for the conveyor capacity.

In equation 3.7.5, the starting time values of the bypassing event and the reclaiming event were set equal if both the events are indeed activated. However, the durations of the events are not synchronised. Therefore, the following constraints ensure that the conveyor capacity is not exceeded irrespective of which duration is considered.

Bypassing duration:

$$\sum_{i,k} q\_r_{ijkp} + \sum_k q\_rsl_{jkp} + q\_prop_{ii,j,p} \leq (Rate\_r \times Dur\_b_{ii,j,p}) + HH(1 - x2_{ii,j,p}) \qquad \forall ii, \quad \forall j, \quad \forall p \tag{3.7.24}$$

Reclaiming duration:

$$\sum_{i,k} q\_r_{ijkp} + \sum_{k} q\_rsl_{jkp} + \sum_{i,p} q\_prop_{ijp} \leq (Rate\_r \times \sum_{k} Dur\_r_{jkp}) + HH(1 - \sum_{k} x_{jkp}) \quad \forall j, \quad \forall p \qquad (3.7.25)$$

e. Blend limit constraints

The total amount of coal supplied to each factory at a point p is given by the following equations:

West:

$$Total\_w_p = \sum_{\substack{i,k, \\ \forall j \in \{y_1, y_2, y_3\}}} q\_r_{ijkp} + \sum_{\substack{k, \\ \forall j \in \{y_1, y_2, y_3\}}} q\_rsl_{jkp} + \sum_{\substack{i, \\ \forall j \in \{y_1, y_2, y_3\}}} q\_prop_{ijp} \qquad \forall p \qquad (3.7.26)$$

East:

$$Total\_e_p = \sum_{\substack{i,k \\ \forall j \in \{y_4, y_5, y_6\}}} q\_r_{ijkp} + \sum_{\substack{k \\ \forall j \in \{y_4, y_5, y_6\}}} q\_rsl_{jkp} + \sum_{\substack{i \\ \forall j \in \{y_4, y_5, y_6\}}} q\_prop_{ijp} \qquad \forall p \qquad (3.7.27)$$

To set the upper limit for the amount of coal from a certain source i to be supplied to the factory at a point p:

West:

$$\sum_{\substack{k, \\ \forall j \in \{y_1, y_2, y_3\}}} q\_r_{ijkp} + \sum_{\forall j \in \{y_1, y_2, y_3\}} q\_prop_{ijp} \leq CL_i \times Total\_w_p \qquad \forall i, \quad \forall p \qquad (3.7.28.a)$$

East:

$$\sum_{\substack{k, \\ \forall j \in \{y_4, y_5, y_6\}}} q\_r_{ijkp} + \sum_{\forall j \in \{y_4, y_5, y_6\}} q\_prop_{ijp} \leq CL_i \times Total\_e_p \qquad \forall i, \quad \forall p \qquad (3.7.28.b)$$

Note that the CL_i parameter is a user input and indicates the maximum limit for coal from source i as a percentage of the total amount supplied at a point p.

The same principle holds for the amount of coal from the strategic stockpile that is supplied to the factory at a point p:

West:

$$\sum_{\substack{k, \\ \forall j \in \{y_1, y_2, y_3\}}} q\_rsl_{jkp} \leq CL\_bleedin \times Total\_w_p \qquad \forall p \qquad (3.7.29.a)$$

East:

$$\sum_{\substack{k, \\ \forall j \in \{y_4, y_5, y_6\}}} q\_rsl_{jkp} \leq CL\_bleedin \times Total\_e_p \qquad \forall p \qquad (3.7.29.b)$$

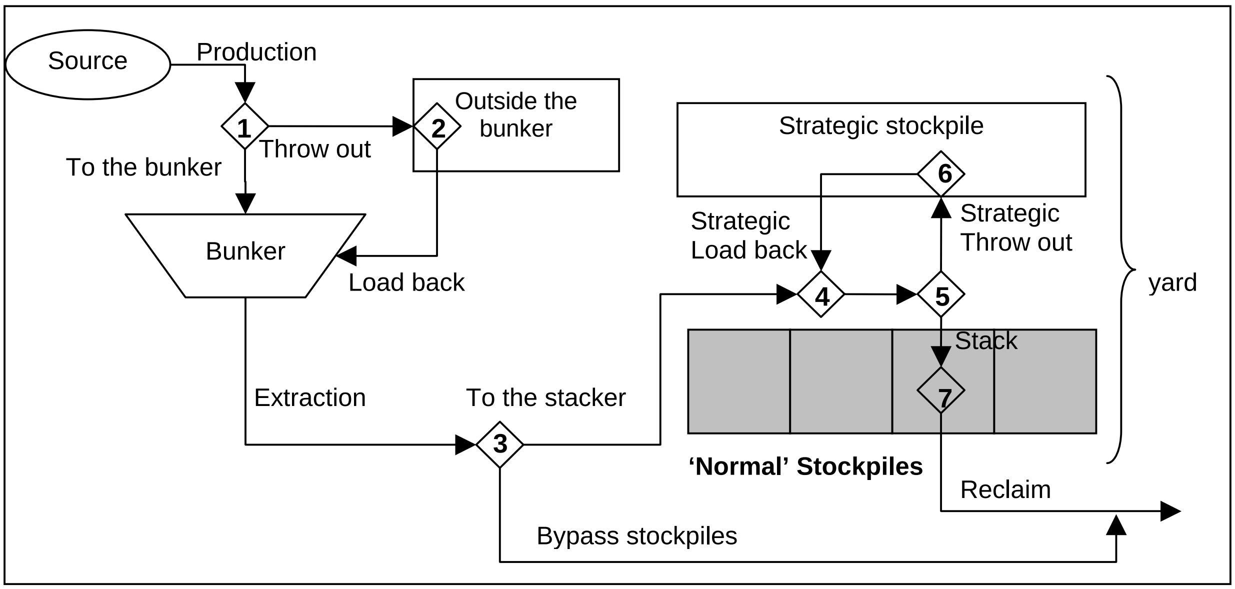
Equations 3.7.28 and 3.7.29 ensure that the maximum limit for a specific source's contribution to the total amount of coal conveyed to the factory is adhered to.

3.3.10. Stockpiles

3.3.10.1. Problem statement

Referring to Figure 1.2, the boundaries for this section is illustrated in Figure 3.11.

Figure 3.11: Normal stockpiles:



a. Individual stockpiles

The stockpiling philosophy was briefly discussed in Chapter 1, but will be explained in more detail here.

The lifecycle of a stockpile can be described as follows:

1. If a new stockpile is to be started on a position on a stockpile yard, the position must be empty (no coal occupying the position).
2. A heap length is chosen for the new stockpile (within the given limits).
3. The stacking process starts. Coal is stacked in horizontal layers across the length of the stockpile (that was set in point 2). Therefore, the length of the stockpile is a fixed value during the stacking process.
4. A specific stockpile's stacking process may be stopped and started as required, but the stockpile may not be reclaimed as long as the stockpile is not stacked to capacity.
5. For a stockpile to reach capacity, the stockpile must contain a certain amount of coal per meter. This value is specified by the user.

6. As soon as the stockpile is full, the reclaiming process can start.
7. The stockpile is steadily reclaimed from one side to the other. The reclaimer reclaims vertical ‘slices’ of the stockpile, removing the total amount of coal that was stacked per meter reclaimed.
8. Thus, the heap length of the stockpile also steadily decreases as the reclaiming process progresses.
9. Because the reclaimer reclaims the stockpile in vertical ‘slices’, each slice contains a small part of each layer that was stacked on the stockpile.
10. The reclaiming process may also be interrupted as required, but a new stockpile may not be started on the same position until all the coal from the previous stockpile has been reclaimed (heap length = 0).

b. Stockpile positions

As mentioned before, it is essential to keep track of source amounts in each stockpile. To enable the tracking, stockpile positions k were assigned to each yard j.

The following factors influenced the choice of the number of positions to be included for each yard:

- By examining the operational reports at SCS, it was determined that the number of stockpiles on a yard seldom exceeds four.
- The number of stockpile positions on a yard influences the size of the model and is restricted to the minimum realistic figure.

Therefore, four stockpile positions k were assigned to each yard j. Note that SCS does not use official stockpile positions, but to enable separate stockpiles on a yard in the scheduling model, the creation of positions were necessary.

The length of a stockpile position is not fixed. For example, one position may be empty (length = 0), while another position may contain a long stockpile of 320m. The total length of the stockpile yard is the upper limit for the combined length of all the stockpiles. The user may also specify a minimum length for any individual stockpile.

3.3.10.2. Mathematical formulation

a. Heap length constraints

The combined length of all the individual stockpiles on a yard j may not exceed the total length of the yard:

$$\sum_{k} heapl\_s_{jkp} \leq Tot\_l \qquad \forall j, \quad \forall p \tag{3.8.1}$$

The heap length for each stockpile at the start of the scheduling time horizon is given by the user:

$$heapl\_s_{jkp} = heapl0\_s_{jk} \qquad \forall j, \quad \forall k, \quad \forall p = p_1 \tag{3.8.2}$$

The stockpile's heap length life cycle is controlled by the following set of equations:

$$\begin{aligned} heapl\_s_{jkp} &\leq heapl\_s_{j,k,p-1} - ((\sum_{i} q\_r_{i,j,k,p-1} + q\_rsl_{j,k,p-1}) / Max0\_s_{jk}) + (Tot\_l \times y_{jkp}) \\ heapl\_s_{jkp} &\geq heapl\_s_{j,k,p-1} - ((\sum_{i} q\_r_{i,j,k,p-1} + q\_rsl_{j,k,p-1}) / Max0\_s_{jk}) + (Min\_l \times y_{jkp}) \\ &\forall j, \quad \forall k, \quad \forall p > p_1 \end{aligned} \tag{3.8.3}$$

Equations 3.8.3 function as follows throughout the life cycle of a stockpile, where the binary variable y_{jkp} indicates the start of a new stockpile:

- Stacking:
  While stacking, the reclaiming term will be 0, and the binary variable will not be activated, therefore the heap length of each point p will be equal to the heap length of the previous point (p-1).
- Reclaiming:
  When reclaiming, the reclaiming term (which consists of the amount reclaimed divided by the capacity per meter as given by the user) subtracts the length of the amount of coal that was reclaimed during the previous point (p-1). The binary variable will still not be activated. This process will repeat itself until the value of $heapl\_s_{jkp}$ equals 0.
- New stockpile:
  When a new stockpile is started, an appropriate new heap length must be chosen, within the given upper and lower limits. Since the heap length of the existing stockpile will be 0, and y_{jkp} =1, the upper and lower limits will be enforced (refer to equations 3.8.8 to 3.8.10 for more detail regarding the new stockpile indicator).

b. Stockpile constraints

The amount of coal from each source on every individual stockpile at the start of the scheduling horizon is given by the user:

$$ST\_sm_{ijkp} = ST0\_sm_{ijk} \qquad \forall i, \quad \forall j, \quad \forall k, \quad \forall p = p_1 \tag{3.8.4}$$

The amount of coal from the strategic stockpiles loaded back on every individual stockpile at the start of the scheduling horizon is given by the user:

$$ST\_sl_{jkp} = ST0\_sl_{jk} \qquad \forall j, \quad \forall k, \quad \forall p = p_1 \qquad (3.8.5)$$

The material balance for the amount of coal from each source on every individual stockpile is formulated as before:

$$ST\_sm_{ijkp} = ST\_sm_{i,jk,p\text{-}1} - q\_r_{i,j,k,p\text{-}1} + q\_s_{i,j,k,p\text{-}1} \quad \forall i, \quad \forall j, \quad \forall k, \quad \forall p > p_1 \qquad (3.8.6)$$

The material balance for the amount of coal from the strategic stockpiles on every individual stockpile can be stated as follows:

$$ST\_sl_{jkp} = ST\_sl_{jk,p\text{-}1} - q\_rsl_{j,k,p\text{-}1} + q\_sl_{j,k,p\text{-}1} \quad \forall j, \quad \forall k, \quad \forall p > p_1 \qquad (3.8.7)$$

Each material balance includes the amount of that specific type of coal on the stockpile at the previous point p (p-1), less the amount of coal that has been reclaimed since the previous point (p-1), plus the amount of coal that has been stacked since the previous point (p-1).

The upper limit for the total amount of coal on an individual stockpile is determined by the amount of coal that must be stacked per meter (as specified by the user) for that specific stockpile:

$$\sum_i ST\_sm_{ijkp} + ST\_sl_{jkp} \le heapl\_s_{jkp} \times Max0\_s_{jk} \qquad \forall j, \quad \forall k, \quad \forall p \qquad (3.8.8)$$

Equations 3.8.9 and 3.8.10 facilitate the stockpile lifecycle:

Reclaiming:

$$\sum_i ST\_sm_{ijkp} + ST\_sl_{jkp} \ge (heapl\_s_{jkp} \times Max0\_s_{jk}) - HH(1 - x_{jkp}) \quad \forall j, \quad \forall k, \quad \forall p \qquad (3.8.9)$$

Equation 3.8.9, together with equation 3.8.8, force the stockpile to be full (equal to capacity) before the reclaiming process can be started ($x_{jkp}=1$).

New stockpile:

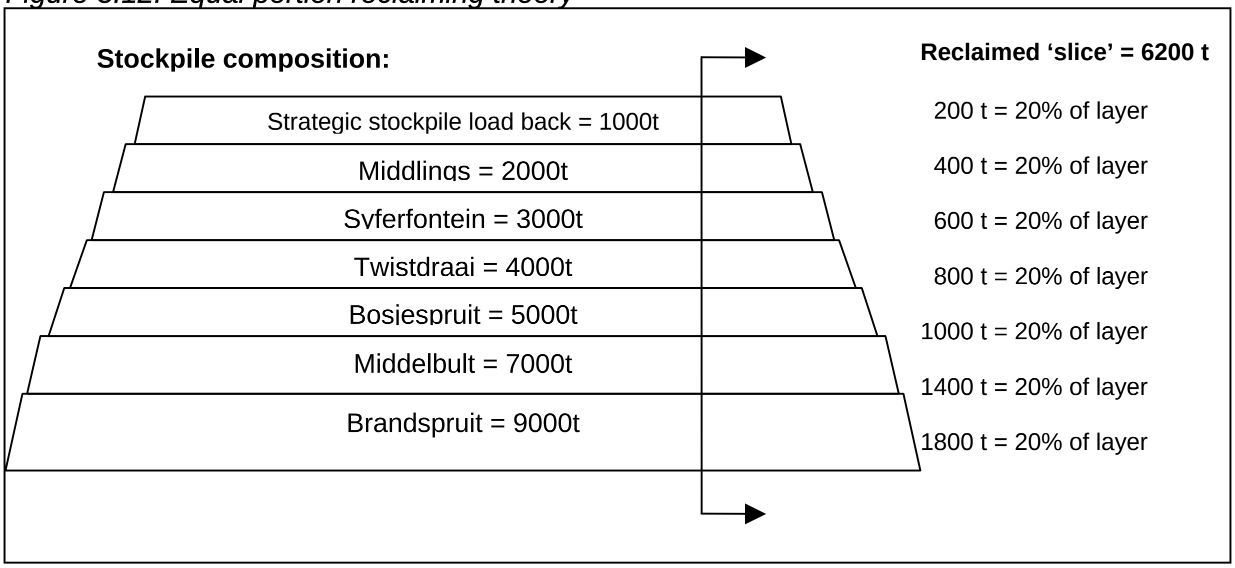
$$\sum_i ST\_sm_{ijkp} + ST\_sl_{jkp} \le HH(1 - y_{jkp}) \qquad \forall j, \quad \forall k, \quad \forall p \qquad (3.8.10)$$

Equation 3.8.10 forces the stockpile to be empty before a new stockpile can be started ($y_{jkp}=1$).

c. Reclaiming process constraints

It was mentioned in section 3.3.10.1 that the reclaiming process removes vertical 'slices' of coal from the stockpile. Thereby, each 'slice' contains a part of each layer that was stacked on the stockpile. This means that with every 'slice' that was reclaimed, an equal portion (%) of all the layers was removed. This theory can be illustrated with the following diagram:

Figure 3.12: Equal portion reclaiming theory



To ensure that the same portion of each layer is reclaimed, the following ***non-linear*** equations are used:

$$\frac{q\_r_{ijkp}}{ST\_sm_{ijkp}} = \frac{q\_r_{ii,j,k,p}}{ST\_sm_{ii,j,k,p}} \qquad \forall i, ii, \quad i \neq ii, \quad \forall j, \quad \forall k, \quad \forall p \qquad (3.8.11)$$

This equation ensures that the portions reclaimed from the different sources' layers on the stockpile, are equal.

Similarly, the portion for the reclaimed portion of strategic stockpile coal must be equal to the other sources' portions:

$$\frac{q\_r_{ijkp}}{ST\_sm_{ijkp}} = \frac{q\_rsl_{jkp}}{ST\_sl_{jkp}} \qquad \forall i, \quad \forall j, \quad \forall k, \quad \forall p \qquad (3.8.12)$$

Note that these two equations are the first ***non-linear*** *equations in the mathematical formulation of the scheduling problem described in this chapter.*

3.3.11. Factory demand and restrictions

3.3.11.1. Problem statement

The following restrictions must be considered:

- The daily demand for coal by the Western and Eastern factories must be met.
- The conveyor towards each factory has a certain capacity (different from the reclaiming conveyor capacity). The total amount of coal sent to the factory may not exceed this capacity.

3.3.11.2. Mathematical formulation

Synfuels has a constant demand for coal. To ensure that this demand is met, the following lower limits hold:

Western factory:

$$\sum_{p} Total\_w_p \geq Dem\_w \qquad \forall p \tag{3.9.1}$$

Eastern factory:

$$\sum_{p} Total\_e_p \geq Dem\_e \qquad \forall p \tag{3.9.2}$$

In equation 3.7.5, the starting time values of the bypassing event and the reclaiming event were set equal if both the events are indeed activated. In equation 3.7.7, the starting time values of different reclaiming activities per Western and Eastern side are set equal. However, the durations of these events are not synchronised. Therefore, the following constraints ensure that the factory conveyor capacity is not exceeded irrespective of which duration is considered.

Bypassing duration:

$$\begin{gathered} Total\_w_p \leq (Rate\_f \times Dur\_b_{ijp}) + HH(1 - x2_{ijp}) \\ \forall i, \quad \forall j \in \{y_1, y_2, y_3\} \quad \forall p \end{gathered} \tag{3.9.3}$$

Reclaiming duration:

$$\begin{gathered} Total\_w_p \leq (Rate\_f \times \sum_{k} Dur\_r_{jkp}) + HH(1 - \sum_{k} x_{jkp}) \\ \forall j \in \{y_1, y_2, y_3\}, \quad \forall p \end{gathered} \tag{3.9.4}$$

Similar equations also hold for the Eastern side.

3.3.12. Maintenance

All maintenance activities were excluded from the first phase model development. However, the maintenance schedule's integration with the operational activities' schedule is discussed in Chapter 4.

3.3.13. Objective function

The objective function formulated in the first phase of the development differs from the objective function of the final scheduling model (refer to Chapter 4). For the first phase, the objective function was chosen to maximise profit, thereby maximising income (by supplying coal to the factory) and minimising cost (such as throwing out coal at the bunkers).

The objective function gives direction to the whole model. It is therefore possible to emphasise some of the elements in the objective functions, by giving it larger or lesser coefficients ('penalties' and 'bonuses'). The role and the influence of the objective function and its structure is discussed at length in Chapter 4.

The elements of the objective function are as follows:

- *Income:* The amount of coal (source and strategic stockpile coal) that is reclaimed and supplied to Synfuels. This is the most viable option to pursue.
- *Income:* The amount of coal bypassed directly to the factory. This option will be penalised because of the risk of supplying 'unblended' coal, with high variability in the quality.
- *Cost:* The amount of coal thrown out at the bunkers and at the strategic stockpiles. This option carries a penalty due to additional handling cost, but also the degradation of coal quality because of the additional handling.
- *Income:* The amount of coal stacked on the stockpiles. This is not a true income (although coal on the stockpiles is viewed as a company asset), but a small encouragement to ensure coal flow through the model.

The objective function is therefore formulated as follows:

$$
\begin{aligned}
z_{\max} = \; & (\sum_{i,j,k,p} q\_r_{ijkp} + \sum_{j,k,p} q\_rsl_{jkp}).Income\_r \\
& + (\sum_{i,p} q\_prop_{ijp}).(Income\_r - Prop\_penalty) \\
& - (\sum_{i,p} q\_bo_{ip} + \sum_{j,p} q\_so_{jp}).Cost \\
& + (\sum_{i,j,k,p} q\_s_{ijkp}).Income\_s
\end{aligned} \qquad (3.11.1)
$$

3.4. RESULTS

3.4.1. Hardware

The mathematical formulation as discussed in section 3.3 results in a Mixed Integer Non-Linear Programming problem (MINLP). GAMS is used as the modelling interface, with Cplex as the Mixed Integer Linear Programming (MILP) solver and CONOPT as the Non-Linear Programming (NLP) solver. The computer used to get the results has a 2.0GHz CPU and 768MB RAM.

3.4.2. Results

Two versions of the model formulated in section 3.3 were tested:

- *Single period model*: The model as formulated, but with only one production period. The results reported in Table 3.5 were achieved with the first period's production rates. Three event points were allocated to the single period to ensure enough event points to accommodate all events.
- *Multi-period model*: The model as formulated, with all six production periods and two event points allocated to each (refer to section 3.3.4).

The reason for testing both these versions of the formulated model, is to test the effect of the number of periods and event points on the solution time of the model. The size and results of both versions of the model are summarised in Table 3.5.

The single period model did not solve within 10 hours, which is very long for a model with only three event points. When the multi-period model with all its periods and twelve event points was tested, the *model did not solve within three days (72 hours)*. Both these models were solved in the Relaxed Mixed Integer Non-Linear Programming (RMINLP) mode, proving that solutions to both these models do indeed exist.

Table 3.5: Size and results of the single and multi-period models

Model type	Reporting criteria	Single period model	Multi-period model
RMINLP	Objective value$_{max}$	1173.57	5422.97
RMINLP	CPU time	25.2 sec	1226.5 sec
MINLP	Number of points p	3	12
MINLP	Binary variables	504	2016
MINLP	Continuous variables	3481	13993
MINLP	Constraints	17203	61134
MINLP	Objective value$_{max}$	-	-
MINLP	CPU time	> 10 hours	> 72 hours
MINLP	Optcr	50%	50%
MINLP	DICOPT Cycles	3	3

This result was critical, since the scheduling model has to be used operationally on a daily basis. It is therefore not acceptable for the model to take longer than three days (or ten hours) to solve. Ideally, the model solution time must be less than an hour.

SCS has a very dynamic system. Events such as equipment breakdowns, longer than expected maintenance downtimes and higher (or lower) than expected mine production rates all have an impact on the operations at SCS. Therefore, it is critical for the scheduling model to be able to adapt to these types of system changes quickly and efficiently.

As explained in Chapter 2, the structure of the model formulation and the amount of binary variables play a big role in the solving performance of a model. Both these elements are addressed in Chapter 4 in order to improve the solution time of the model. The model formulation improvements are discussed in detail.

These model improvements are aimed at reducing the model solution time from more than 10 hours to less than one hour.

3.5. PHASE 1 CONCLUSION

In this chapter, the detail mathematical formulation of the SCS scheduling problem described in Chapter 1 was discussed. The unit-specific event based continuous time representation technique was used to develop the presented formulation.

Some of the most important formulation aspects that received attention in this chapter include the following:

- The approximation of the mines' production profiles by creating production periods. The result is a combination between a global event based time representation and the unit-specific event based time representation in the model (refer to section 3.3.4).
- The use of various binary variables at each decision point in the coal flow of a mine's coal (refer to section 3.3.7), and the reduction of binary variables when stacking coal on a specific stockpile (refer to section 3.3.8).
- The sequencing of time values to prevent certain events to start before other events have finished. Especially at the mine bunkers (section 3.3.5), the stackers (sections 3.3.7 and 3.3.8) and the reclaimers (section 3.3.9), these constraints play an important role. Reclaiming and bypass events that happen simultaneously are forced to coincide to ensure that conveyor capacities are not exceeded.
- The use of a continuous variable (in stead of a binary variable) to keep track of the reclaimer's position. This variable is formulated to act as a 'memory' variable that keeps track of the reclaimer's position even if no reclaiming event is scheduled for the reclaimer (equations 3.7.12 to 3.7.16).
- The restriction of a reclaimer's movement if there are stockpiles situated between its current and its destination position (equations 3.7.17 and 3.7.18).
- The only non-linear equation in the model resulted from the formulation of the reclaiming process. When reclaiming a stockpile, equal portions of each of the layers on the stockpile are reclaimed (section 3.3.10).

The model was tested with a single period and a multi-period approach. The single period model with three event points solved within 24 hours, but the multi-period model did not solve within 72 hours (although the relaxed MINLP model solved within seconds).

As a result of the model not solving within an acceptable time for operational use, the model structure and the use of binary variables are revised in Chapter 4.

CHAPTER 4: MODEL IMPROVEMENT (PHASE 2)

This chapter presents the second phase of the SCS scheduling model development. The improvement techniques discussed in Chapter 2 are applied to the basic formulation presented in Chapter 3 and discussed in detail. Other solution time reduction options are also investigated. Finally, the results from the improved model are presented and compared to the results in Chapter 3.

4.1. INTRODUCTION

The basic mathematical formulation for the SCS scheduling problem was developed in the first phase (Chapter 3). In the second phase, the problem formulation of phase one was improved to ensure that the model solves in an acceptable time for operational use. The improvement efforts was aimed at reducing the model solution time to less than one hour.

The following improvement approaches are followed or explored to achieve a faster solution time:

1. The use of ***SOS1 variables*** to reduce the amount of binary variables in the model. The Special Ordered Sets of type 1 (SOS1) variables are applied in the following ways in the model:
    a. To model infrastructure restrictions (such as the restriction that only one Western mine can supply to the Eastern stockpiles at a time).
    b. To model the decision options presented in Figure 1.2.
    c. To account for the maintenance activities which were not formulated in the basic model in Chapter 3.
2. The revision of the ***time and duration variables' application***. The reduction in time and duration variables will reduce the amount of sequencing constraints and therefore also simplify the model structure.
3. The application of the ***alternative NLP linearization and solution technique*** discussed in Chapter 2, to improve solution time with regards to the non-linear equations in the model. As a result of this application, the blend plan non-linear equations (which were excluded from the basic model in Chapter 3) are included in the improved model.
4. In Chapter 3, two versions of the basic model were tested, namely a ***single-period model*** and a multi-period model. This approach of dividing the 24 hour scheduling model into six smaller scheduling models, each representing a single production period,

is explored further to reduce the number of event points per model.

5. Finally, the option of dividing the model into two parts, a 'Reclaiming and Bypass' part and an 'Extraction and Stacking' part, is explored.

The principles followed for each of these approaches will be discussed separately. The specific changes made to the basic model in Chapter 3 will be presented in detail. Finally, the model results will be presented and compared to the results of the basic model in Chapter 3.

4.2. SPECIAL ORDERED SETS TYPE 1 (SOS1) VARIABLES

There are a number of ways to improve a model's performance, one of which is the reduction of the number of binary variables in the model. The number of binary variables is a very important factor, which slow down the solution time of the model dramatically. In Chapter 2, the concept of SOS1 variables was introduced. SOS1 variables were applied to the basic model to reduce the number of binary variables in the model.

The definition for SOS1 variables from the GAMS user manual is as follows (Brooke et al, 1998): "*At most one variable within an SOS1 set can have a non-zero value. This variable can take any positive value... The members of the innermost index belong to the same set*."

Thus, if QQ_{jpi} is an SOS1 quantity variable for event i in unit j at point p, the following constraint illustrates the principle of using SOS1 variables:

$$\sum_i QQ_{jpi} \leq rate\_b \qquad \forall j, \quad \forall p \qquad (4.2.1)$$

Note that the position of the i-index is situated last to be the set under control, according to the definition. Equation 4.2.1 ensures that at most one quantity variable of all the events i in a certain unit j has a value. If it indeed has a value, the upper limit is the rate for that specific event. The use of an SOS1 variable in equation 4.2.1 eliminates the need for a binary variable to ensure that a stockpile yard j receives coal from only one mine.

The three application areas of the SOS1 variables (infrastructure restrictions, decision options and maintenance activities) are discussed in detail, with the additional sets and variables needed for the SOS1 formulation. The mathematical formulation for all three application areas is discussed in section 4.2.4.

4.2.1. SOS1 applied to infrastructure restrictions

4.2.1.1. Problem statement

The first application area for the SOS1 variables is the infrastructure restrictions at SCS. These restrictions include the following:

- A mine i may supply coal to only one stockpile yard j at a time.
- A stacker j may receive coal from only one mine i at a time.
- Only one mine's coal may be bypassed at a stockpile yard j at a time.
- Only one Western mine's coal (Brandspruit or Middelbult) may be conveyed to the Eastern stockpile yards (yard 4, 5 and 6) at a time.
- Only one Eastern mine's coal (Bosjespruit, Twistdraai, Syferfontein or Middlings) may be conveyed to the Western stockpile yards (yard 1, 2 and 3) at a time.

The two infrastructure restrictions not mentioned in the list above, is that there is only one stacker and one reclaimer per stockpile yard. However, these restrictions are still formulated with stacking and reclaiming binary variables as before, to facilitate change-over restriction and time allocation.

4.2.1.2. Sets and variables

a. Sets:

The following sets are declared in addition to the sets in section 3.3.2.a:

iw_i = Western sources (subset of sources i) $iw \in \{ Br, Mb\}$

ie_i = Eastern sources (subset of sources i) $ie \in \{ Bo, Tw, Syf, Mdl\}$

b. SOS1 Variables:

q_b_{ipj} The amount of coal extracted from bunker i to stockpile yard j at point p. Ensures that the mine i supplies coal to only one stockpile j at a point p.

q_supply_{jpi} The amount of coal supplied to stacker j from bunker i to at point p. Ensures that stacker j receives coal from only one mine i at a point p.

q_prop_{jpi} The amount of coal bypassed at stockpile yard j from bunker i to at point p. Ensures that only one mine i's coal is bypassed at stockpile yard j at a point p.

q_w2e$_{p\ iw}$ The total amount of coal conveyed from the Western mines iw to the Eastern stockpiles at point p. Ensures that only one Western mine iw's coal is conveyed to the Eastern stockpiles at a point p.

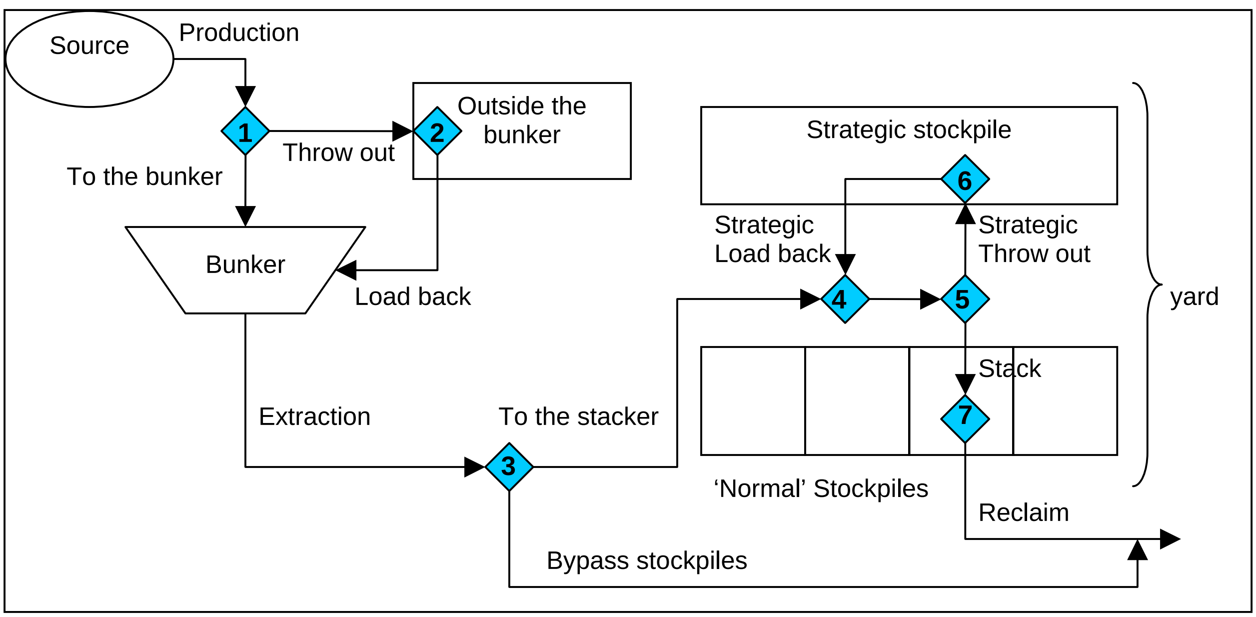
q_e2w$_{p\ ie}$ The total amount of coal conveyed from the Eastern mines ie to the Western stockpiles at point p. Ensures that only one Eastern mine ie's coal is conveyed to the Western stockpiles at a point p.

4.2.2. SOS1 applied to the decision points

4.2.2.1. Problem statement

The second application area for the SOS1 variables is the decision points in the following coal flow diagram:

Figure 4.1: Coal flow options and decision points for one source



The decision points where SOS1 variables are applied are the following:

1. The mine's production may either be thrown into the bunker, or be thrown out outside the bunker.
2. Coal may only be thrown out outside the bunker or be loaded back into the bunker from outside the bunker. These events may not happen simultaneously.
3. Coal extracted from the mine's bunker may either be conveyed to the stacker or be bypassed directly to the factory.

4. Either extracted coal from the mine's bunker or coal loaded back from the strategic stockpile may be conveyed to the stacker, but not simultaneously.
5. The stacker may either stack coal on the individual stockpiles or throw out coal on the strategic stockpile.
6. Coal may either be thrown out at a strategic stockpile, or loaded back from the strategic stockpile to be stacked on one of the individual stockpiles.

Decision point 7 states that an individual stockpile may either be stacked or reclaimed, but not both simultaneously. As explained in section 4.2.1.1 above, this decision point is not reformulated with an SOS1 variable, because the stacking and reclaiming binary variables are still used for the stacking and reclaiming processes.

4.2.2.2. Sets and variables

a. Sets:

To enable the SOS1 formulation of the decision points explained above, the options at the different decision points are declared as sets:

a = options to the stream of extracted coal (decision points 3 and 5)

$a \in \{ Bypass, Stack, Out\}$

b = options for coal conveyed to the stacker (decision point 4)

$b \in \{ Load\_in, ROM\}$

c = options for produced coal at the bunker (decision point 1)

$c \in \{ Mine\_out, Bunker\}$

d = options for coal handling outside a bunker or a strategic stockpile (decision points 2 and 6)

$d \in \{ Fel\_out, Fel\_in\}$

b. SOS1 Variables:

q_mopt_{ipc} Ensures that only one option c is activated for mine i's production at point p (decision point 1).

q_bfel_{ipd} Ensures that the front-end loaders (fel) are used for only one of the handling options d at bunker i, point p (decision point 2).

q_bopt_{ijpa} Ensures that only one of the options a is chosen for the stream of coal extracted from bunker i to stockpile yard j at point p (decision points 3 and 5).

q_sopt_{jpb} Ensures that only one of the options b is chosen for the type of coal conveyed to stacker j at point p (decision point 4).

Note that no SOS1 variable is declared for decision point 6, since the variables for decision points 3, 4 and 5 already implies that throw out and load back on the strategic stockpile may not occur simultaneously.

The application of the SOS1 variables declared above enables the reduction of a large quantity of the binary variables presented for the basic model formulated in Chapter 3.

4.2.3. SOS1 applied to maintenance actions

4.2.3.1. Problem statement

The third application area for the SOS1 variables is the maintenance actions for stackers and reclaimers. Since a piece of equipment can only be available for operation or on maintenance downtime, but not both, it is an ideal opportunity to apply another SOS1 variable.

The following maintenance options are included in the model:

- Stacker maintenance.
- Reclaimer maintenance.

Other equipment's maintenance options may later be included in the scheduling model, but for the purposes of this document only the stacker and reclaimer maintenance options are considered.

4.2.3.2. Sets and variables

a. Sets:

To enable the SOS1 formulation of the maintenance options, these options are declared as a set:

f = set of equipment availability options $f \in \{ Maint, Go\}$

b. SOS1 Variables:

The following two duration variables are redefined to include set f as explained above:

Dur_s_{jpf} Indicates the duration of either an operational event or a maintenance event at stacker j, point p.

Dur_r$_{jpf}$ Indicates the duration of either an operational event or a maintenance event at reclaimer j, point p.

The declaration of these duration SOS1 variables prevent the use of additional binary variables to model maintenance actions on specific equipment.

c. Continuous Variables:

The user will provide the required maintenance duration per period for both stackers and reclaimers:

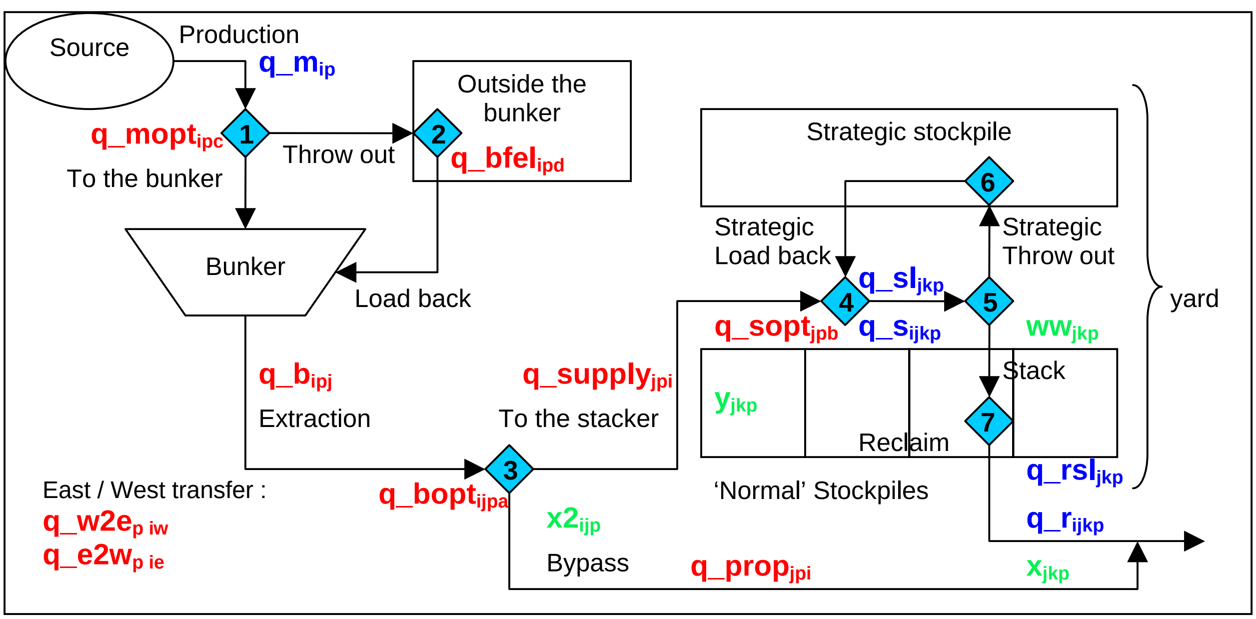
Dur_smaint$_{jr}$ Required stacker j maintenance during period r.

Dur_rmaint$_{jr}$ Required reclaimer j maintenance during period r.

4.2.4. Mathematical formulation of SOS1 variables

In sections 4.2.1 to 4.2.3 above, 11 new SOS1 variables and five new sets have been defined. The detail mathematical formulation of the SOS1 approach follows. Figure 4.2 indicates the position of the applicable quantity variables which includes SOS1 (red), positive continuous variables (blue) and remaining binary variables (green), as stated in Chapter 3.

Figure 4.2: Coal flow options, decision points and applicable quantity variables



The mathematical formulation for the following areas of concern will be discussed:

- Decision points 1 and 2.
- Extraction and the East / West restriction.
- Decision point 3.
- Decision point 4.
- Stacking with maintenance.
- Material balances.
- Reclaiming with maintenance.

4.2.4.1. Decision points 1 and 2

a. Quantity constraints

The amount produced by mine i at point p is determined as follows (refer to equation 3.2.6):

$$q\_m_{ip} = Dur\_m_{ip} \times Rate\_m_{ir} \qquad \forall i, \quad \forall per_{rp} \tag{4.2.2}$$

To ensure the full transfer of the mine's production to the chosen option at decision point 1, the following equation holds:

$$\sum_c q\_mopt_{ipc} = q\_m_{ip} \qquad \forall i, \quad \forall p \tag{4.2.3}$$

Equation 4.2.3 acts as a mass balance and ensures that the full amount of mine i's production at point p is directed to whichever option is chosen (throw out or throw into the bunker).

When the throw out option is chosen, the amount of coal is transferred to the throw out option of the decision point 2 SOS1 variable:

$$\begin{gathered} q\_mopt_{ipc} = q\_bfel_{ipd} \\ \forall i, \quad \forall p, \quad \forall c = Mine\_out, \quad \forall d = Fel\_out \end{gathered} \tag{4.2.4}$$

At decision point 2, the following upper limits are set:

1. A general upper limit for the SOS1 variable options:

   $$\sum_d q\_bfel_{ipd} \leq HH \qquad \forall i, \quad \forall p \tag{4.2.5}$$

   This constraint is similar to equation 3.3.6 where the binary variable was used to set an upper limit with the term ($H.v_{ip}$).

2. The upper limit for the amount of coal loaded back, determined by the amount of front-end loaders available:

$$\sum_{i} q\_bfel_{ipd} \leq rate\_bl \times H \times Loaders \quad \forall p, \quad \forall d = Fel\_in \tag{4.2.6}$$

This constraint is similar to equation 3.3.10.

3. The upper limit for the amount of coal loaded back, determined by the amount of coal outside the bunker:

$$q\_bfel_{ipd} \leq Out\_b_{ip} \qquad \forall i, \quad \forall p, \quad \forall d = Fel\_in \tag{4.2.7}$$

Equation 4.2.7 is similar to equation 3.3.11.

b. Duration constraints

The timing and duration constraints for the mine's production and the coal loaded back to the bunker remains the same as stated in sections 3.3.4 and 3.3.5, except for the following equation which replaces equations 3.3.6 and 3.3.7.

$$Dur\_bl_{ip} = \frac{q\_bfel_{ipd}}{rate\_bl} \qquad \forall i, \quad \forall p, \quad \forall d = Fel\_out \tag{4.2.8}$$

It is important to note the following:

- Since an exact number of front-end loaders is not allocated to the load back event, the duration of the load back event is calculated based on the loading rate of *one* front-end loader. This ensures that the 'worst case' scenario where only one front-end loader is available, is taken into account.

- The minimum duration limits for all durations linked to SOS1 variables are discarded. As a result of the SOS1 formulation and the absence of the binary variables, a lower limit will force the SOS1 variable to have a minimum value at *every point p*. In such a case no inactive events will occur. Thus, equation 3.3.7 becomes redundant.

4.2.4.2. Extraction from the bunkers and the East/West restriction

a. Quantity constraints

To ensure that a source i can only supply one yard j at a point p (similar to equation 3.4.1), and to include the quantity upper limit (similar to equation 3.4.15), the following equation holds:

$$\sum_{j} q\_b_{ipj} \leq ST\_b_{ip} \qquad \forall i, \quad \forall p \tag{4.2.9}$$

The upper limit for the amount of coal extracted from bunker i depends on the extraction rate (refer to equation 3.4.14):

$$q\_b_{ipj} \leq rate\_b_i \times H \qquad \forall i, \quad \forall j, \quad \forall p \tag{4.2.10}$$

The East / West transfer infrastructure restrictions are captured as follows:

West to East:

$$\begin{gathered} \sum_{iw} q\_w2e_{p,iw} \leq rate\_s \times H \\ q\_w2e_{p,iw} = \sum_{\forall j \in \{y_4, y_5, y_6\}} q\_b_{ii,p,j} \\ \forall ii = iw_i, \quad \forall p \end{gathered} \tag{4.2.11.a}$$

East to West:

$$\begin{gathered} \sum_{ie} q\_e2w_{p,ie} \leq rate\_s \times H \\ q\_e2w_{p,ie} = \sum_{\forall j \in \{y_1, y_2, y_3\}} q\_b_{ii,p,j} \\ \forall ii = ie_i, \quad \forall p \end{gathered} \tag{4.2.11.b}$$

Note the following:

- The first equation in each set enforces the SOS1 restriction that only one iw or ie source can be activated.
- The second equation in each set links the q_b_{ipj} extraction SOS1 variable with the East or West transfer SOS1 variables.
- The upper limit for the first equation in each set is set to the maximum stacking rate and not the maximum extraction rate, because the extraction rates of the different sources differ, but the stacking rate for all stackers are the same.

b. Time constraints

The time and duration constraints for the extraction events are discussed in section 4.3.

4.2.4.3. Decision point 3

a. Quantity constraints

At decision point 3, the coal extracted from bunker i to yard j must be directed either to the stacker or to the bypass conveyor in which case the coal is directly conveyed to the factory. If the coal is directed to the stacker, the coal can either be stacked on one of the individual stockpiles or thrown out on the strategic stockpile.

To ensure that the total amount of coal extracted from a bunker i is redirected to one of the options a, the following equation holds:

$$\sum_{a} q\_bopt_{ijpa} = q\_b_{ipj} \qquad \forall i, \quad \forall j, \quad \forall p \tag{4.2.12}$$

Equation 4.2.12 also ensures that the SOS1 variable always chooses an option a, if coal is indeed extracted. The reverse is also true: the SOS1 variable is forced to be 0 (that is, no option is chosen), if no coal is extracted from the mine's bunker.

Bypass option:

If the Bypass option is chosen, the following transfer equation holds:

$$q\_prop_{jpi} = q\_bopt_{ijpa} \qquad \forall i, \quad \forall j, \quad \forall p, \quad \forall a = Bypass \tag{4.2.13}$$

Equation 4.2.13 ensures that the full amount of coal extracted from the bunker and redirected to the bypass option, is indeed transferred to the bypass quantity variable.

The bypass SOS1 variable ensures that only one mine i's coal is bypassed at stockpile yard j at a time:

$$\sum_{i} q\_prop_{jpi} \leq rate\_r \times H \times \sum_{i} x2_{ijp} \qquad \forall j, \quad \forall p \tag{4.2.14.a}$$

The bypass binary variable is used to set the upper limit for the bypass SOS1 variable:

$$q\_prop_{jpi} \leq rate\_r \times H \times x2_{ijp} \qquad \forall i, \quad \forall j, \quad \forall p \tag{4.2.14.b}$$

Stacker options:

If one of the Stacker options is chosen, namely to stack or to throw out the coal on the strategic stockpile, the following transfer equation holds:

$$q\_supply_{jpi} = \sum_{a \in \{Stack, Out\}} q\_bopt_{ijpa} \qquad \forall i, \quad \forall j, \quad \forall p \tag{4.2.15}$$

Equation 4.2.15 ensures that the full amount of coal extracted from the bunker and redirected to the stacker options, is transferred to the stacker supply quantity variable.

Similar to the bypass SOS1 variable, the stacker supply SOS1 variable ensures that only one mine i's coal is supplied to stacker j at a time:

$$\sum_{i} q\_supply_{jpi} \leq rate\_s \times H \qquad \forall j, \quad \forall p \tag{4.2.16}$$

b. Time constraints

The time and duration constraints for the bypass events are discussed in section 4.3.

4.2.4.4. Decision point 4

a. Quantity constraints

At decision point 4, a decision must be taken which type of coal is to be conveyed to the stacker, coal extracted from the mine (at SCS the term Run Of Mine coal, that is ROM coal, is used) or coal loaded back from the strategic stockpile.

The stacker option SOS1 variable ensures that only one of the options above for a specific stacker j is chosen at a point p:

$$\sum_{b} q\_sopt_{jpb} \leq rate\_s \times Dur\_s_{jpf} \qquad \forall j, \quad \forall p, \quad \forall f = Go \tag{4.2.17}$$

Note that the upper limit for the SOS1 variable is dependent on the stacker's 'availability duration'. This concept is discussed in section 4.2.4.5.

Extraction (ROM) coal option:

If the Extraction coal option is chosen, the following transfer equation holds:

$$q\_sopt_{jpb} = \sum_{i} q\_supply_{jpi} \qquad \forall j, \quad \forall p, \quad \forall b = ROM \tag{4.2.18}$$

Equation 4.2.18 ensures that the full amount of coal extracted from the bunker and redirected to the stacker option, is transferred to the stacker option quantity variable.

Strategic stockpile load back option:

If the strategic stockpile load back option is chosen, the following upper limits are set (similar to equations 3.5.14 and 3.5.15):

$$q\_sopt_{jpb} \leq Strat_{jp} \qquad \forall j, \quad \forall p, \quad \forall b = Load\_in \tag{4.2.19}$$

$$q\_sopt_{jpb} \leq rate\_bl \times H \times Loaders \qquad \forall j, \quad \forall p, \quad \forall b = Load\_in \tag{4.2.20}$$

The following constraint is similar to equation 3.5.16 and ensures that the total amount of coal loaded back at the bunkers and from the strategic stockpiles does not exceed the capacity of the available number of front-end loaders:

$$\sum_{i} q\_bfel_{ipd} + \sum_{j} q\_sopt_{jpb} \leq rate\_r \times H \times Loaders$$
$$\forall p, \quad \forall b = Load\_in, \quad \forall d = Fel\_in \tag{4.2.21}$$

4.2.4.5. Stacking

The stacking binary variable ww_{jkp}, and the two stacking quantity variables q_s_{ijkp} and q_sl_{jkp} from the basic model in Chapter 3 remain unchanged.

a. Quantity constraints

The amount of coal extracted from bunker i to be stacked on stockpile k, yard j has to be linked to the variable q_s_{ijkp}. The stacking variable q_s_{ijkp} is controlled by the stacking binary variable (refer to equation 3.6.5), which ensures that stacking only takes place at one stockpile k on a yard j at a point p.

$$q\_bopt_{ijpa} = \sum_k q\_s_{ijkp} \quad \forall i, \quad \forall j, \quad \forall p, \quad \forall a = Stack \tag{4.2.22}$$

The same principle holds for the amount of coal loaded back from the strategic stockpile:

$$q\_sopt_{jpb} = \sum_k q\_sl_{jkp} \qquad \forall j, \quad \forall p, \quad \forall b = Load\_in \tag{4.2.23}$$

In Chapter 3 the stacking variable q_s_{ijkp} is linked to the stacking binary variable by equation 3.6.5. For the same reason, the following equation holds (similar to the duration constraint 3.5.10):

$$q\_sl_{jkp} \le HH \times ww_{jkp} \qquad \forall j, \quad \forall k, \quad \forall p \tag{4.2.24}$$

b. Duration constraints

The stacker duration variable Dur_s_{jpf} was redefined in 4.2.3.2 to include the set of availability options f. This SOS1 variable ensures that either the operational event or the maintenance event is activated, but not both.

The upper limit for any event allocated by the SOS1 variable is the scheduling time horizon:

$$\sum_f Dur\_s_{jpf} \le H \qquad \forall j, \quad \forall p \tag{4.2.25}$$

The SOS1 variable has to allocate the required maintenance per period. Note that the required maintenance is an input from the user.

$$\sum_{p \neq p_n} Dur\_s_{jpf} = Dur\_smaint_{jr} \qquad \forall j, \quad \forall per_{rp}, \quad \forall f = Maint \tag{4.2.26}$$

Note that the maintenance event may not be allocated to the last point p_n, to ensure that the finish time of the event can be allocated to a point p (refer to Chapter 2).

The operational stacking duration is determined as follows (similar to equation 3.6.11):

$$Dur\_s_{jpf} = \sum_{ik}(q\_s_{ijkp} / rate\_b_i) \quad + \sum_{k}(q\_sl_{jkp} / rate\_bl) \qquad \forall j, \quad \forall p, \quad \forall f = Go \tag{4.2.27}$$

The upper limit for the operational stacker event duration is set as follows (similar to equation 3.4.11):

$$Dur\_s_{jpf} \geq Dur\_min \times \sum_{k} ww_{jkp} \qquad \forall j, \quad \forall p, \quad \forall f = Go \tag{4.2.28}$$

Note that this SOS1 duration variable is allowed to have a lower limit due to the stacker binary variable ww_{jkp} which is activated when an operational stacker event is scheduled.

4.2.4.6. Reclaiming

a. Duration constraints

Similar to the stacking duration constraints, the following constraints hold:

$$\sum_{f} Dur\_r_{jpf} \leq \quad H \qquad \forall j, \quad \forall p \tag{4.2.29}$$

$$\sum_{p \neq p_n} Dur\_r_{jpf} = Dur\_rmaint_{jr} \qquad \forall j, \quad \forall per_{rp}, \quad \forall f = Maint \tag{4.2.30}$$

The operational reclaiming duration is determined as follows:

$$Dur\_r_{jpf} \geq \sum_{ik}(q\_r_{ijkp} / rate\_r) \quad + \sum_{k}(q\_sl_{jkp} / rate\_r) \qquad \forall j, \quad \forall p, \quad \forall f = Go \tag{4.2.31}$$

Note that this equation represents a minimum duration limit where *rate_r* equals the maximum reclaiming rate. Since a reclaimer's speed can be slower than the maximum reclaiming rate, the reclaiming duration may be longer than the minimum limit. This equation replaces equation 3.7.25.

The upper limit for the operational reclaimer event duration is set as follows (similar to equation 3.7.11):

$$Dur\_r_{jpf} \geq Dur\_min \times \sum_{k} x_{jkp} \qquad \forall j, \quad \forall p, \quad \forall f = Go \tag{4.2.32}$$

Note that this SOS1 duration variable is allowed to have a lower limit due to the reclaimer binary variable x_{jkp} which is activated when an operational reclaimer event is allocated.

Similar to constraint 3.7.24, the following equation holds. Note that the $Dur\_b_{ii,j,p}$ variable is replaced by a bypass duration variable Dur_prop_{ijp} which will be discussed in section 4.3.

$$\sum_{ik} q\_r_{ijkp} + \sum_{k} q\_rsl_{jkp} + \sum_{i} q\_prop_{jpi} \leq (Rate\_r \times \sum_{i} Dur\_prop_{ijp}) + HH(1 - \sum_{i} x2_{ijp}) \qquad \forall j, \quad \forall p \tag{4.2.33}$$

Similar to equations 3.9.3 and 3.9.4, the following constraints ensure that the capacity of the conveyor to the Western factory is not exceeded:

Bypassing duration:

$$Total\_w_p \leq (Rate\_f \times \sum_{i} Dur\_prop_{ijp}) + HH(1 - \sum_{i} x2_{ijp}) \qquad \forall j \in \{y_1, y_2, y_3\}, \quad \forall p \tag{4.2.34}$$

Reclaiming duration:

$$Total\_w_p \leq (Rate\_f \times Dur\_r_{jpf}) + HH(1 - \sum_{k} x_{jkp}) \qquad \forall j \in \{y_1, y_2, y_3\}, \quad \forall p, \quad \forall f = Go \tag{4.2.35}$$

Equations 4.2.34 and 4.2.35 also hold for the Eastern side. Note that the operational reclaiming duration is used in equation 4.2.25, thereby excluding the maintenance durations from the capacity calculation.

4.2.5. Previous constraints

The following quantity and allocation constraints discussed in Chapter 3 are still applicable: 3.4.16b, 3.4.17b, 3.4.18, 3.6.2, 3.6.5, 3.7.1, 3.7.2, 3.7.20, 3.7.21 and 3.7.22.

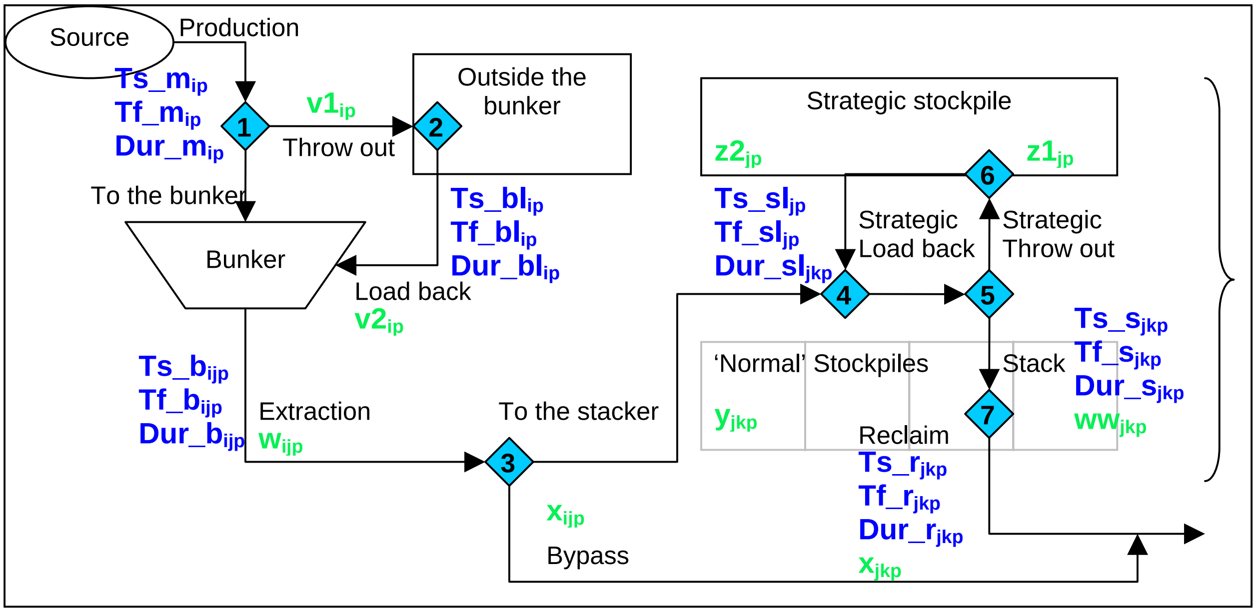
4.3. REVISION OF TIME AND DURATION VARIABLES AND SEQUENCING

The specific use and formulation of time and duration variables with its associated sequencing constraints have a significant impact on the size and structure of the model. The formulation of these variables presented in Chapter 3 is revised and reformulated. The aim is twofold:

- To minimise the amount of time and duration variables used.
- To simplify the model structure.

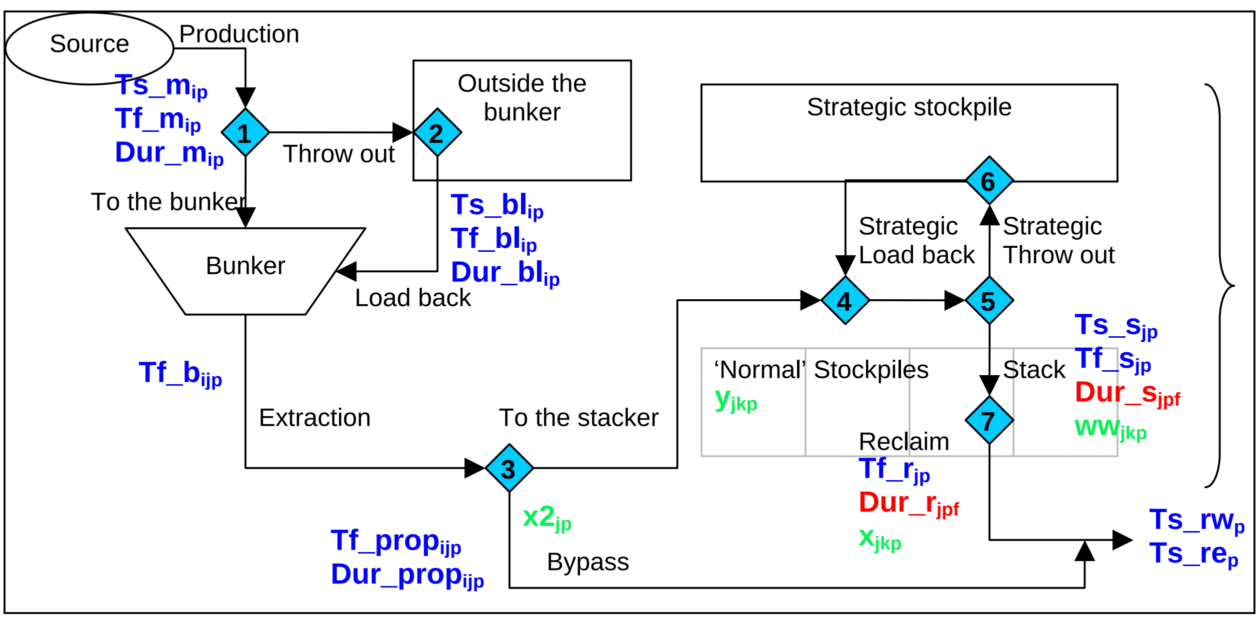
In the formulation presented in Chapter 3, most of the major events had starting, finishing and duration variables as illustrated in Figure 4.3 (binary variables indicated in green). This formulation resulted in 3888 single continuous variables. With this number of variables, the amount of sequencing constraints also escalated.

Figure 4.3: Coal flow options, decision points and Chapter 3's time and duration variables



The time and duration representation were changed as indicated in Figure 4.4. The normal time and duration variables (blue), the SOS1 duration variables (red) and the remaining binary variables (green) are indicated. Note that Figure 4.2 indicates all the SOS1 variables and should be read together with Figure 4.4.

Figure 4.4: Coal flow options, decision points and relevant variables



The basic strategy behind the changes can be summarised as follows:

1. To have *global starting and finishing time values for the stacker at each stockpile yard j*. Since there is only one stacker per yard, a global starting time for the stacker will be adequate. This will also ensure time sequencing between extraction, stacking, strategic throw out and load back events.
2. To have a *global starting time for reclaiming per Western and Eastern side*. Since all reclaimed or bypassed coal must share one conveyor to the factory per side, all reclaiming and bypassing activities need to be synchronised. A global starting time will achieve this.
3. To have a *global finishing time value for reclaiming at each stockpile yard j*. Similar to point 1 above, there is only one reclaimer per stockpile yard j. The global starting time for the reclaiming event is captured by point 2 above. Therefore, a global finishing time is defined to eliminate unnecessary variables.

4.3.1. Defining variables

The following positive continuous variables are defined as part of the revision process described above. Note that the SOS1 duration variables (indicated in Figure 4.4 in red) was defined in section 4.2 above.

Ts_s_{jp}	Starting time value for any event involving the stacker on yard j at point p.
Tf_s_{jp}	Finishing time value for any event involving the stacker on yard j at the previous point (p-1).
Ts_rw_{p}	Starting time value for any reclaiming or bypassing event on the Western side at point p.
Ts_re_{p}	Starting time value for any reclaiming or bypassing event on the Eastern side at point p.
Tf_r_{jp}	Finishing time value for any reclaiming event on yard j at the previous point (p-1).
Tf_prop_{ijp}	Finish time value if coal is bypassed to the factory from mine i at yard j, point p.
Dur_prop_{ijp}	Duration of the bypass event from mine i to yard j, starting at point p.

The two additional bypass variables are defined to ensure correct sequencing with the reclaiming variables when the bypass event does take place.

The variables that were discarded are:

- *The starting time (Ts_b_{ijp}) and duration (Dur_b_{ijp}) variables for bunker extractions.* Extracted coal can only be stacked, thrown out on the strategic stockpile, or bypassed. The global starting times and durations for all these events were defined above, making separate extraction starting time and duration variables redundant.
- *The starting time (Ts_sl_{jp}), finishing time (Tf_sl_{jp}) and duration (Dur_sl_{jp}) variables for coal loaded back from the strategic stockpiles.* The strategic load back event implies that the stacker must be used to stack the coal that has been loaded back. The global stacker starting, finishing and duration variables were defined above, making separate strategic load back variables redundant.

The formulation as presented in Figure 4.4 results in the number of time and duration variables to be reduced from 3888 to 2256, a 42% reduction.

4.3.2. Mathematical formulation

Since the basic time and duration formulation principles were illustrated in Chapter 3, the formulation of the new time variables will not be discussed in detail. However, the sequencing constraints will be discussed in more detail.

Table 4.1: Sequencing constraint summary with relevant equation numbers

Finish time →	1. Mine	2. Bunker Load	3. Extraction	4. Bypass	5. Stacker	6. Reclaimer
Start time ↓						
1. Mine	≥ 4.3.1	≥ 4.3.2	≥ 4.3.3	≥ 4.3.4	/	/
2. Bunker load	≥ 4.3.5	≥ 4.3.6	≥ 4.3.7	≥ 4.3.8	/	/
3. Stacker	≥ 4.3.9	≥ 4.3.10	≥ 4.3.11	/	≥ 4.3.12	≥ 4.3.13
4. Global supply	≥ 4.3.14	≥ 4.3.15	≥ 4.3.16	/	≥ 4.3.17	≥ 4.3.18

The sequencing constraints are structured around the four starting time variables, namely mine production, bunker load in, stacker and reclaiming/bypass starting times. Table 4.1 provides a summary guide for the sequencing constraints, with each constraint's relevant equation number.

Table 4.1 must be read as follows: Start with the start time column and read from left to right to the relevant finishing time column. For example: the stacker starting time must be greater than or equal to the extraction finishing time at a certain point p, the relevant equation is 4.3.11.

Each of the start time categories is discussed separately.

4.3.2.1. Mine production starting time

As illustrated in Table 4.4, the following sequencing constraints are necessary:

$$Ts\_m_{ip} \geq Tf\_m_{ip} \qquad \forall i, \quad \forall p \tag{4.3.1}$$

$$Ts\_m_{ip} \geq Tf\_bl_{ip} \qquad \forall i, \quad \forall p \tag{4.3.2}$$

$$Ts\_m_{ip} \geq Tf\_b_{ijp} \qquad \forall i, \quad \forall j, \quad \forall p \tag{4.3.3}$$

$$Ts\_m_{ip} \geq Tf\_prop_{ijp} - H(1 - x2_{i,j,p-1}) \qquad \forall i, \quad \forall j, \quad \forall p > p_1 \tag{4.3.4}$$

Note that equation 4.3.4 is the only equation that can be relaxed, since it is only the bypass event (in this set of equations) that is controlled with a binary variable. Equations 3.2.4 and 3.3.4.a are replaced by equations 4.3.1 and 4.3.2 respectively.

4.3.2.2. Bunker load in starting time

Very similar to the constraints above, the following sequencing constraints regarding the coal loaded in from outside the bunker, are necessary:

$$Ts\_bl_{ip} \geq Tf\_m_{ip} \qquad \forall i, \quad \forall p \tag{4.3.5}$$

$$Ts\_bl_{ip} \geq Tf\_bl_{ip} \qquad \forall i, \quad \forall p \tag{4.3.6}$$

$$Ts\_bl_{ip} \geq Tf\_b_{ijp} \qquad \forall i, \quad \forall j, \quad \forall p \tag{4.3.7}$$

$$Ts\_bl_{ip} \geq Tf\_prop_{ijp} - H(1 - x2_{i,j,p-1}) \qquad \forall i, \quad \forall j, \quad \forall p > p_1 \tag{4.3.8}$$

Equations 3.3.4.b and 3.3.3 are replaced by equations 4.3.5 and 4.3.6 respectively.

4.3.2.3. Stacker starting time

As illustrated in Table 4.4, the following stacker sequencing constraints are defined:

$$Ts\_s_{jp} \geq Tf\_m_{ip} - H[(\sum_{ii} q\_b_{ii,p,j}) - q\_b_{ipj}] \qquad \forall i, \quad \forall j, \quad \forall p \qquad (4.3.9)$$

$$Ts\_s_{jp} \geq Tf\_bl_{ip} - H[(\sum_{ii} q\_b_{ii,p,j}) - q\_b_{ipj}] \qquad \forall i, \quad \forall j, \quad \forall p \qquad (4.3.10)$$

$$Ts\_s_{jp} \geq Tf\_b_{i,jj,p} - H[(\sum_{ii} q\_b_{ii,p,j}) - q\_b_{ipj}] \qquad \forall i, \quad \forall j, jj \quad \forall p \qquad (4.3.11)$$

$$Ts\_s_{jp} \geq Tf\_s_{jp} \qquad \forall j, \quad \forall p \qquad (4.3.12)$$

$$Ts\_s_{jp} \geq Tf\_r_{jp} - H(2 - ww_{jkp} - x_{j,k,p-1}) \qquad \forall j, \quad \forall k, \quad \forall p > p_1 \qquad (4.3.13)$$

In an attempt to relax the sequencing constraints between the stacking, extraction, mine production and bunker load in events, the extraction SOS1 variable is used (equations 4.3.9, 4.3.10 and 4.3.11). If the amount of coal extracted from bunker i to yard j equals the total amount of coal extracted to that specific yard j, the equation is enforced. This will also be the case if no coal is extracted to that specific yard j. Otherwise it is relaxed. In equation 4.3.13, the stacking and reclaiming binary variables are used in the relaxation term.

4.3.2.4. Global supply starting time per side

The following supply sequencing constraints are defined for the Western side (similar equations hold for the Eastern side):

$$Ts\_rw_p \geq Tf\_m_{ip} - H(1 - x2_{ijp}) \qquad \forall i, \quad \forall j \in \{y_1, y_2, y_3\}, \quad \forall p \qquad (4.3.14)$$

$$Ts\_rw_p \geq Tf\_bl_{ip} - H(1 - x2_{ijp}) \qquad \forall i, \quad \forall j \in \{y_1, y_2, y_3\}, \quad \forall p \qquad (4.3.15)$$

$$Tf\_b_{ijp} \leq Ts\_rw_p + H(1 - x2_{ijp}) \qquad \forall i, \quad \forall j \in \{y_1, y_2, y_3\}, \quad \forall p \qquad (4.3.16)$$

$$Ts\_rw_p \geq Tf\_s_{jp} - H(2 - ww_{j,k,p-1} - x_{jkp}) \qquad \forall j \in \{y_1, y_2, y_3\}, \quad \forall k, \quad \forall p > p_1 \qquad (4.3.17)$$

$$Ts\_rw_p \geq Tf\_r_{jp} \qquad \forall j \in \{y_1, y_2, y_3\}, \quad \forall p \qquad (4.3.18)$$

The only time when the global supply variables have to be synchronised with the mine production, bunker load in and stacking event, is when the coal from that specific bunker is bypassed directly to the factory. Therefore, the bypass binary variable $x2_{ijp}$ is used in the relaxation terms of equations 4.3.14, 4.3.15 and 4.3.16.

4.3.2.5. Change-over time constraints

Conveyor change-over time

In equation 3.4.12, the applicable change-over time per conveyor is determined by the binary variable w_{ijp}. Since this binary variable is replaced by the SOS1 variables, the equation needs to be reformulated:

$$Tf\_b_{i,jj,p} \leq Tf\_b_{i,j,p+1} - \sum_{k}(q\_sm_{ijkp} / rate\_b_i) - CT\_b_{ij} \qquad \forall i, \quad \forall j, jj \quad \forall p < p_n \qquad (4.3.19)$$

In equation 4.3.19, the duration of the stacking event at point p is subtracted from the finish time of that same event (point p+1). The result is the starting time of the extraction/stacking event. The amount of change-over time to change from destination yard jj to yard j is subtracted from the starting time. The finishing time value of the previous event must be less than or equal to this calculated time value that takes the change-over time into consideration.

Stacker change-over time

The stacker change-over time constraints in equation 3.6.13 remain unchanged.

Reclaimer change-over time

The reclaimer starting time variable in the reclaimer change-over time constraints 3.7.19 is replaced by the global supply starting time variables defined in this section:

$$Ts\_rw_p \geq Tf\_r_{jp} + (CT\_r_{k,kk} \times x_{j,kk,p}) - H(2 - x_{j,kk,p} - Pos\_r_{j,k,p-1})$$
$$\forall j \in \{y_1, y_2, y_3\}, \quad \forall k, kk, \quad k \neq kk, \quad \forall p > p_1 \qquad (4.3.20.a)$$

$$Ts\_rw_p \geq Tf\_r_{jp} + (CT\_r_{k,kk} \times x_{j,kk,p}) - H(2 - x_{j,kk,p} - Pos0\_r_{j,k})$$
$$\forall j \in \{y_1, y_2, y_3\}, \quad \forall k, kk, \quad k \neq kk, \quad \forall p = p_1 \qquad (4.3.20.b)$$

Similar equations also apply to the Eastern side.

4.4. NLP LINEARIZATION AND ALTERNATIVE SOLUTION TECHNIQUE

Apart from the amount of binary variables in the model, the amount of non-linear equations in the formulation is also an important factor determining the model solution time. In section 2.9 a reformulation and alternative solution method for non-linear equations were discussed. As part of the model improvement efforts, the reformulation and alternative solution method is applied to the following two sets of non-linear equations:

- The *reclaiming process non-linear equations* 3.8.11 and 3.8.12 described in Chapter 3.
- The *blend plan non-linear equations* which were omitted from the model described in Chapter 3.

The linearization of these non-linear equations is discussed according to the steps provided in section 2.9 in Chapter 2.

4.4.1. Variables

The following additional positive continuous variables are declared in order to be used in the linearization equations:

Reclaiming process:

$Ratio_{jkp}$	The ratio in which the layers on the stockpile is reclaimed during the reclaiming process (refer to Figure 3.12).
$Portion1_{ijkp}$	The variable to substitute the non-linear part in the reclaiming process equation during linearization. This variable represents the layers of a specific mine i's coal.
$Portion2_{jkp}$	The variable to substitute the non-linear part in the reclaiming process equation during linearization. This variable represents the layers of coal loaded back from the strategic stockpile.

Blend plan – Western side:

$Blend\_w_{ip}$	The percentage contribution of mine i's coal in the total amount of coal supplied to the Western factory at point p.
$Bleedin\_w_p$	The percentage contribution of coal loaded back from the strategic stockpile in the total amount of coal supplied to the Western factory at point p.
$BBlend\_w_{ip}$	The variable to substitute the non-linear part in the Western blend equation during linearization. This variable represents the blend of a specific mine i's coal at a specific point p.
$BBleedin\_w_p$	The variable to substitute the non-linear part in the Western blend equation during linearization. This variable represents the blend of the coal loaded back from the strategic stockpiles at a specific point p.

Blend plan – Eastern side:

$Blend\_e_{ip}$	The percentage contribution of mine i's coal in the total amount of coal supplied to the Eastern factory at point p.
$Bleedin\_e_p$	The percentage contribution of coal loaded back from the strategic stockpile in the total amount of coal supplied to the Eastern factory at point p.
$BBlend\_e_{ip}$	The variable to substitute the non-linear part in the Eastern blend equation during linearization. This variable represents the blend of a specific mine i's

coal at a specific point p.

BBleedin_e$_p$ The variable to substitute the non-linear part in the Eastern blend equation during linearization. This variable represents the blend of the coal loaded back from the strategic stockpiles at a specific point p.

4.4.2. Linearization of non-linear equations

4.4.2.1. Reclaiming process

The steps described in section 2.9 of Chapter 2 are followed:

Step1:

Define the following non-linear equations to describe the reclaiming process (refer to equations 3.8.11 and 3.8.12):

$$q\_r_{ijkp} = ST\_m_{ijkp} \times Ratio_{jkp} \qquad \forall i, \quad \forall j, \quad \forall k, \quad \forall p \tag{4.4.1}$$

$$q\_rsl_{jkp} = ST\_sl_{jkp} \times Ratio_{jkp} \qquad \forall j, \quad \forall k, \quad \forall p \tag{4.4.2}$$

Note that the same $Ratio_{jkp}$ variable is used in both equations, ensuring that the reclaiming process remove equal portions of all layers on the stockpile per point p.

The non-linear parts in the equations above are replaced by the replacement variables defined in section 4.4.1:

$$\begin{aligned} &Portion1_{ijkp} = ST\_m_{ijkp} \times Ratio_{jkp} \\ &\therefore \quad q\_r_{ijkp} = Portion1_{ijkp} \\ &\qquad \forall i, \quad \forall j, \quad \forall k, \quad \forall p \end{aligned} \tag{4.4.3}$$

$$\begin{aligned} &Portion2_{jkp} = ST\_sl_{jkp} \times Ratio_{jkp} \\ &\therefore \quad q\_rsl_{jkp} = Portion2_{jkp} \\ &\qquad \forall j, \quad \forall k, \quad \forall p \end{aligned} \tag{4.4.4}$$

As a result of the substitution, the second equation in each set of equations above is now linear instead of non-linear.

Step 2:

Define the limits of the non-linear parts in the equations defined in Step 1. In this case, the limits for the ratio variable and the two storage variables ST_m_{ijkp} and ST_sl_{jkp} are set as follows:

$$Ratio_{jkp} \leq 1 \qquad \forall j, \quad \forall k, \quad \forall p \tag{4.4.5}$$

$$ST\_m_{ijkp} \leq Tot\_length \times Max0\_s_{jk} \qquad \forall i, \quad \forall j, \quad \forall k, \quad \forall p \tag{4.4.6}$$

$$ST\_sl_{jkp} \leq Tot\_length \times Max0\_s_{jk} \qquad \forall j, \quad \forall k, \quad \forall p \tag{4.4.7}$$

The ratio variable may not exceed 1 (therefore 100%), while the amount of coal from a certain source may not exceed the maximum capacity per stockpile yard. No minimum limits are applicable in these instances, therefore only the maximum limits will be applied to the linearization equations in Step 3.

It is important to note that the purpose of these limits is to introduce constants into the non-linear equation. The constants are used in the linearization equations described in Step 3. If the right hand side of any of the limits includes a variable, the linearization in Step 3 will not realise.

Step 3:

Apply McCormick's linearization technique (1976) to the layers of mine i's coal on the stockpile:

$$\begin{aligned} Portion1_{ijkp} \geq\ & (Tot\_length \times Max0\_s_{jk} \times Ratio_{jkp}) \\ & + (1 \times ST\_sm_{ijkp}) - (Tot\_length \times Max0\_s_{jk} \times 1) \\ Portion1_{ijkp} \leq\ & Tot\_length \times Max0\_s_{jk} \times Ratio_{jkp} \\ Portion1_{ijkp} \leq\ & 1 \times ST\_sm_{ijkp} \\ & \forall i, \quad \forall j, \quad \forall k, \quad \forall p \end{aligned} \tag{4.4.8}$$

Apply McCormick's linearization technique (1976) to the layers of strategic stockpile coal loaded back on the stockpile:

$$\begin{aligned} Portion2_{jkp} \geq\ & (Tot\_length \times Max0\_s_{jk} \times Ratio_{jkp}) \\ & + (1 \times ST\_sl_{jkp}) - (Tot\_length \times Max0\_s_{jk} \times 1) \\ Portion2_{jkp} \leq\ & Tot\_length \times Max0\_s_{jk} \times Ratio_{jkp} \\ Portion2_{jkp} \leq\ & 1 \times ST\_sl_{jkp} \\ & \forall j, \quad \forall k, \quad \forall p \end{aligned} \tag{4.4.9}$$

Steps 4 and 5 are discussed separately.

4.4.2.2. Blend plan constraints

The same principles as described above are followed to formulate the blend linearization equations. Only the Western blend linearization will be discussed in detail, since the Eastern blend linearization equations yields exactly the same format.

Step1:

Define the blending non-linear equations:

$$\sum_{\substack{\forall j \in \{y_1, y_2, y_3\} \\ \forall k}} q\_r_{ijkp} + \sum_{\forall j \in \{y_1, y_2, y_3\}} q\_prop_{jpi} = Blend\_w_{ip} \times Total\_w_p \qquad \forall i, \quad \forall p \tag{4.4.10}$$

$$\sum_{\substack{\forall j \in \{y_1, y_2, y_3\} \\ \forall k}} q\_rsl_{jkp} = Bleedin\_w_p \times Total\_w_p \qquad \forall p \tag{4.4.11}$$

Note that the *Blend_w*$_{ip}$ and *Bleedin_w*$_p$ variables represent the contribution of a specific source's coal to the total amount of coal supplied to the factory at a point p.

The non-linear parts in the equations above are replaced by the replacement variables defined in section 4.4.1:

$$BBlend\_w_{ip} = Blend\_w_{ip} \times Total\_w_p$$

$$\therefore \sum_{\substack{\forall j \in \{y_1, y_2, y_3\} \\ \forall k}} q\_r_{ijkp} + \sum_{\forall j \in \{y_1, y_2, y_3\}} q\_prop_{jpi} = BBlend\_w_{ip} \tag{4.4.12}$$

$$\forall i, \quad \forall p$$

$$BBleedin\_w_p = Bleedin\_w_p \times Total\_w_p$$

$$\therefore \sum_{\substack{\forall j \in \{y_1, y_2, y_3\} \\ \forall k}} q\_rsl_{jkp} = BBleedin\_w_p \tag{4.4.13}$$

$$\forall p$$

Step 2:

Define the limits of the non-linear parts in the equations defined in Step 1. In this case, the limits for the blend variables *Blend_w*$_{ip}$ and *Bleedin_w*$_p$ and the total amount of coal supplied to the factory are set as follows:

$$Blend\_w_{ip} \le CL_i \qquad \forall i, \quad \forall p \tag{4.4.14}$$

$$Bleedin\_w_p \le CL\_bleedin \qquad \forall p \tag{4.4.15}$$

$$Total\_w_p \le rate\_f \times H \qquad \forall p \tag{4.4.16}$$

A specific source's contribution to the total amount of coal supplied to the factory at a point p may not exceed the maximum limits set by the user (refer to equations 3.7.28 and 3.7.29). The total amount of coal supplied to the factory may not exceed the conveyor capacity. No minimum limits are applicable in these instances, therefore only the maximum limits will be applied to the linearization equations in Step 3.

Step 3:

Apply McCormick's linearization technique (1976) to the mines' contribution to the supply:

$$\begin{aligned}
BBlend_w_{ip} &\geq (CL_i \times Total_w_p) \\
&\quad + (rate_f \times H \times Blend_w_{ip}) - (CL_i \times rate_f \times H) \\
BBlend_w_{ip} &\leq CL_i \times Total_w_p \\
BBlend_w_{ip} &\leq rate_f \times H \times Blend_w_{ip} \\
&\forall i, \quad \forall p
\end{aligned} \tag{4.4.17}$$

Apply McCormick's linearization technique (1976) to the contribution of strategic stockpile coal to the total supply:

$$\begin{aligned}
BBleedin_w_p &\geq (CL_bleedin \times Total_w_p) \\
&\quad + (rate_f \times H \times Bleedin_w_p) - (CL_bleedin \times rate_f \times H) \\
BBleedin_w_p &\leq CL_bleedin \times Total_w_p \\
BBleedin_w_p &\leq rate_f \times H \times Bleedin_w_p \\
&\forall p
\end{aligned} \tag{4.4.18}$$

4.4.3. Solution method

In Chapter 2 the solution method was described in Steps 4 and 5. It is applied as follows:

Step 4:

Two models are defined, an MILP and an MINLP model. All the normal constraints described in sections 4.2 and 4.3 is included in both models. The difference between the models is summarised as follows:

- *MILP:*
  Included: Equations 4.4.3.b, 4.4.4.b, 4.4.12.b, 4.4.13.b, the limits 4.4.5 to 4.4.7 and 4.4.14 to 4.4.16, as well as the linearization equations 4.4.8, 4.4.9, 4.4.17 and 4.4.18.
  Excluded: The non-linear equations in 4.4.1, 4.4.2, 4.4.10 and 4.4.11.
- *MINLP:*
  Included: The non-linear equations in 4.4.1, 4.4.2, 4.4.10, 4.4.11 and the limits 4.4.5 to 4.4.7 and 4.4.14 to 4.4.16.
  Excluded: Equations 4.4.3.b, 4.4.4.b, 4.4.12.b, 4.4.13.b, as well as the linearization equations 4.4.8, 4.4.9, 4.4.17 and 4.4.18.

Step 5:

The MILP model is solved first. The solution found with this model provides upper and lower bounds to the MINLP problem. The MILP solution is used as input to the MINLP model which is solved after the inputs have been received.

This process provides a much more stable model. The Chapter 3 formulation would often result in NLP-infeasible answers. However, the Chapter 4 formulation, with the solution technique described above, provided frequent NLP-feasible answers.

Note that the tightness of the limits defined in Step 2 plays an important role in the effective application of this technique. The purpose of the linearization solution technique is to set boundaries for the MINLP solution. If the limits in Step 2 are defined too wide, the boundaries for the MINLP problem will also be set too wide, and solution problems may still occur.

4.5. SINGLE PERIOD vs MULTI-PERIOD MODELS

The index which contributes the most to the size of a model is the set of time points p. All binary variables and all continuous variables have the set of time points p as one of their indices. Therefore, by reducing the number of time points p, the number of variables in the model reduces dramatically.

In an effort to improve the scheduling model's solution time, the number of periods is reduced from six to one. A minimum of three event points are allocated to the single period to allow at most two consecutive events to be scheduled. The same periods described in Chapter 3, section 3.3.4, are used.

As a result of the period reduction above, the model will only provide a schedule for the one specific period for which it was solved. To get a 24 hour schedule, the scheduling model must solve six times, one period at a time. The results from one model solution must provide the input to the next model.

The advantages and disadvantages of solving six smaller models, each for only one period, compared to solving one large model which includes all six periods can be summarised as follows:

Advantages:

- The size of the model decreased with 75% (from twelve to three points).
- The model solution time reduced exponentially.
- In a dynamic system, it might be better to schedule more frequently for shorter periods, than for the total 24 hours. Possible changes in the system status (for example breakdowns) or deviation in mine production can then be captured and incorporated into the schedule.

Disadvantages:

- The sum of all the single periods' optimal solutions may not equal the global optimum when solving for 24 hours.
- The result of the point above is that it may happen that an event is scheduled in one period which causes a sub optimal answer in the next period.
- To ensure that the problem above is addressed, a planning model is needed to provide optimal daily targets for the scheduling model. This is a disadvantage because another model is needed instead of only one comprehensive scheduling model.

The advantages and disadvantages were considered and the decision was taken to implement the single period model rather than the multi-period model. The fast solution time (refer to section 4.8) was the determining factor. In addition, it was decided to develop a planning model to assist in providing global optimum targets for the scheduling model. This planning model's development and implementation are outside the scope of this project.

4.6. DIVIDING THE MODEL

The next attempt to reduce the model solution time was to divide the model into two smaller models according to the nature of the SCS scheduling problem. The operations are divided as follows:

1. Reclaiming and bypassing activities supplying coal to the factory.
2. Bunker, extraction, stacking and strategic stockpile activities conveying coal from the sources to the stockpiles at SCS.

A separate model for each of the two divisions above was formulated, each with its own objective function. The 'reclaiming' model and the 'stacking' model are discussed separately.

Reclaiming model

The reclaiming model includes all reclaiming activities as well as the bypass activities. The following points of interest should be noted:

- To ensure there will be coal available when a bypass activity is scheduled, the mine production events are also included in this model. However, mine bunker levels are not included in the model.
- The stockpiles' status remain the same for the time horizon of this model. No stacking activities are included in the model. Therefore, a stockpile cannot become reclaimable during the scheduling horizon, if it has not been in the 'reclaimable' status at the start of the scheduling horizon.
- Maintenance times on the reclaimers are included in the model.

The results from the reclaiming model includes detail schedules for all reclaiming and bypassing activities, taking into account mine production and reclaimer maintenance times.

Stacking model

The schedule from the 'reclaiming' model becomes the fixed input for the 'stacking' model. Thus, the stacking model cannot change any reclaiming or bypassing event, but has to schedule all other activities around these fixed inputs. All mine production activities, bunker throw out and load back activities, extraction, stacking and strategic stockpile activities are included in the stacking model. Take note of the following:

- Stockpiles' status may change during the scheduling horizon of the stacking model, since new stockpiles may be started or stockpiles may be stacked to capacity, changing its status from 'stacking' to 'reclaimable'. The last instance will result in the stockpile to be reclaimed only in the next scheduling cycle for the next period.
- Stacker maintenance activities are included in the model.

Result

The advantages and disadvantages when dividing the original model into two smaller parts are very similar to those of single period vs multi-period models discussed in section 4.5 above. The major advantage is the reduction in model size and therefore model solution time. The major disadvantage, again, is the question of global optimality when solving the two models separately. This optimality question will also be addressed with the use of a planning optimisation model as described in section 4.5.

4.7. OBJECTIVE FUNCTION

The last improvement to the Chapter 3 model was the revision of the objective function described in equation 3.11.1. Since a lot of structural changes were made to the model, as described in sections 4.2 to 4.6, the objective function had to change accordingly. Different stacking and reclaiming objective functions were defined to be used by the two different models.

Apart from the structural changes, the coefficients of the objective were revised as well. In Chapter 3, the real income and cost values were used in the objective function. This approach was changed to rather make use of scaled 'penalties' and 'bonuses' in the objective function. These bonus and penalty coefficients have a very big impact on the performance of the model. The SCS management must agree on the coefficients (or weights) used for the items in the objective function.

Stacking objective function

The main objective of the stacking objective function is to convey coal from the mine bunkers to the stockpiles, irrespective of the action taken at decision point 3 and 5 (refer to Figure 4.1). Penalties are incurred if coal is thrown out at the mine bunkers and when the bunker is not at its minimum level at the end of the scheduling horizon.

$$\begin{aligned} z_{\max} = \; & 2\times\sum_{i,j,p} q\_b_{ipj} \\ & -1.5\times\sum_{\substack{i,p \\ c=Mine\_out}} q\_mopt_{ipc} \\ & -1\times\sum_{i,p_1}(ST\_b_{ip}-(Bunk\_min\times Cap\_b_i)) \end{aligned} \qquad (4.7.1)$$

Note that the coefficients provide the priorities in the objective function. For example, it will be better to leave the bunker 100% full than to throw out coal at that bunker unnecessarily, but the most advantageous option will be to extract the coal from the bunker.

Reclaiming objective function

The main objective of the reclaiming objective function is to supply the factory with coal. Therefore, bonuses are achieved for the amount of coal reclaimed and the amount of coal bypassed from the mines to the factory. The following list of penalties may be incurred:

- A penalty on reclaimer runtime. This penalty ensures that the minimum amount of reclaimers are used in order to achieve the optimum blend to the factory.
- A penalty on bypass conveying duration. This penalty ensures that the conveyor from

the mine bunker is used optimally whenever coal must be bypassed to the factory. It prevents the model to schedule very small amounts of coal to be bypassed just to achieve a better blend.

- A penalty on the deviation from the blend plan per side. This penalty ensures that the nearest possible blend to the blend plan will be achieved. The following blend plan variables are applicable:

CL_w_i	Contribution percentage of mine i coal supplied to the Western factory
CL_e_i	Contribution percentage of mine i coal supplied to the Eastern factory
CLw_bleedin	Contribution percentage of strategic stockpile coal supplied to the Western factory
Cle_bleedin	Contribution percentage of strategic stockpile coal supplied to the Eastern factory

- A penalty for step changes in the blend between event points per side. This penalty ensures that the most stable blend is supplied to the factory.

The reclaiming objective function is formulated as follows:

$$\begin{aligned} z_{\max} = \; & 4\times\sum_{ijkp}(q\_r_{ijkp} + q\_rsl_{jkp}) \quad +1\times\sum_{ijp}(q\_prop_{jpi}) \\ & -0.5\times\sum_{i,p,f=Go}(Dur\_r_{jpf}) \quad -0.5\times\sum_{ijp}(Dur\_prop_{ijp}) \\ & -0.5\times\sum_{ip}(Blend\_w_{ip} - CL\_w_i) \quad -0.5\times\sum_{p}(Bleedin\_w_p - CLw\_bleedin) \\ & -0.5\times\sum_{ip}(Blend\_e_{ip} - CL\_e_i) \quad -0.5\times\sum_{p}(Bleedin\_e_p - CLe\_bleedin) \\ & -0.5\times\sum_{i,p>p_1}(Blend\_w_{i,p-1} - Blend\_w_{ip}) \quad -0.5\times\sum_{p>p_1}(Bleedin\_w_{p-1} - Bleedin\_w_p) \\ & -0.5\times\sum_{i,p>p_1}(Blend\_e_{i,p-1} - Blend\_e_{ip}) \quad -0.5\times\sum_{p>p_1}(Bleedin\_e_{p-1} - Bleedin\_e_p) \end{aligned} \qquad (4.7.2)$$

4.8. RESULTS

The results of the following models are compared:

1. The full single period model from Chapter 3, including stacker and reclaimer actions.
2. The full single period model formulated in Chapter 4, including stacker and reclaimer actions. The NLP linearization solution technique is applied to this model.
3. The divided single period model described in section 4.6, including a stacking and a reclaiming model. The NLP linearization solution technique is applied to this model.

In the case where more than one model was solved in order to achieve the final answer, the results from all the sub-models are reported. The results are discussed and compared in terms of model size, model solution time and the objective value.

Table 4.2: Size and result of models

Reporting criteria	**Full single period model (Chapter 3)**	**Full single period model (Chapter 4)**		
		MILP model	**MINLP model**	**Final result**
Number of points p	3	3	3	3
Binary variables	504	300	300	-
Continuous variables	3499	4749	4203	-
Constraints	15115	11892	10254	-
Objective value$_{max}$	-	180.1857	177.1756	**177.1756**
CPU time	**> 10 hours**	122.416 sec	233.366 sec	**355.788 sec**
Optcr	50%	10%	10%	-
DICOPT Cycles	3	-	3	-

Table 4.3: Size and result of the two parts of the divided model for one period

Reporting criteria	**Divided single period model (Chapter 4)**			
	Reclaim MILP	**Reclaim MINLP**	**Stack MILP**	**Final result**
Number of points p	3	3	3	3
Binary variables	180	180	120	-
Continuous variables	3339	2793	2581	-
Constraints	7487	5849	5630	-
Objective value$_{max}$	115.5401	116.8625	67.31	**184.1725**
CPU time	0.65 sec	3.345 sec	101.245 sec	**105.25 sec**
Optcr	10%	10%	10%	-
DICOPT Cycles	-	3	-	-

Chapter 3 and Chapter 4 model comparison (Table 4.2):

- Model size:

  The number of binary variables reduced with 40% as a result of the SOS1 application in Chapter 4. This however also increased the number of continuous variables with 20%. The number of constraints reduced as a result of the revision of the timing and duration constraints discussed in section 4.3.
- Solution time:

  The solution time of the model improved drastically from longer than ten hours to only 356 seconds (almost six minutes). The solution time target set in Chapter 3 was less than one hour. This was achieved.
- Objective value:

  No objective value is reported for the Chapter 3 model. No comparison can be made.

Overall, the improvements in Chapter 4 resulted in a significant improvement in both the model size and the model solution time.

Full model and divided model comparison – Chapter 4 (Table 4.3):

- Model size:

  The number of binary variables is reduced with 40% as a result of the separation of the reclaiming and stacking models. Logically, the number of variables and equations also reduced.
- Solution time:

  The solution time of the divided model was reduced by 70% to only 105 seconds (less than two minutes), compared to the full model. This dramatic reduction in solution time caused the model to be fit for operational use. With such a fast solution time, reaction to breakdowns or deviation in the mine supply can be incorporated and optimised in the model almost immediately.
- Objective value:

  The total of the divided models' objective functions is more than the full model's objective function. This result counters the global optimum concern raised in section 4.6.

The divided single period model's size and performance are substantially better than the original Chapter 3 models. Based on the results reported above, the divided single period model was chosen to be implemented at SCS (refer to Chapter 5 for implementation remarks).

Since the single period model was chosen, an optimisation planning model has to be developed to provide global optimum targets for the scheduling model. The development phase of this model is already underway and will be used together with the scheduling model as one optimisation tool for SCS.

4.9. PHASE 2 CONCLUSION

This chapter presented the actions taken to improve the basic model formulation of Chapter 3. Various techniques and alternative formulation approaches were applied to reduce the number of binary variables, to improve the model structure and to reduce the solution time of the model.

The following important improvement actions are highlighted:

- SOS1 variables were applied to the infrastructure restrictions, the decision points and the maintenance activities in the model. This formulation eliminated a large amount of binary variables.
- Semi-global timing and duration variables were implemented for each stockpile yard and for the Eastern and Western reclaiming sides. This reduced the number of variables and constraints. It also simplified the structure of the model.
- The NLP linearization and solution method was applied. This resulted in faster solution times for the MINLP models, since the pre-solved MILP model determines a feasible upper bound for the MINLP solution space.
- By solving for only one period at a time, the total number of event points (the biggest contributor to model size) per model is drastically reduced, and the solution time of a model reduced exponentially. Global optimality may however be compromised (refer to section 4.5).
- By dividing the model into two parts for stacking and reclaiming, each part's model size and therefore the solution time is still reduced. Again, global optimality may be compromised by such a division.

The improved models solved with very positive results. The divided single period model's total solution time is less than two minutes. In comparison with the original 72 hours in Chapter 3, this is a major improvement in the model's performance. The aim of the improvement actions was to reduce the model solution time to less than one hour. This was achieved, and it is concluded that the improvements implemented in Chapter 4 were successful.

CHAPTER 5: CONCLUSION

The final chapter of this document aims to conclude the development of the SCS scheduling model. The implementation phase and the value added to the SCS operations by using the model are discussed. The scheduling and time representation techniques applied in the SCS scheduling model are evaluated and future research opportunities highlighted.

5.1. THE END OF THE BEGINNING…

The development and improvement of the SCS scheduling model has been discussed at length in Chapter 3 and 4. The model development was thereby concluded. The implementation of the model in the SCS environment followed.

The rest of this chapter includes the following discussion points, with regards to the completed scheduling model:

- A brief overview of the practical implications when implementing the scheduling model.
- An example of the typical output from the operational scheduling model.
- A discussion on the value added to the SCS operations by using the scheduling model.

The scheduling formulation and time representation techniques as discussed in Chapter 2 and applied in Chapter 3 and 4 are evaluated. The applicability and the robustness of the available techniques are critically evaluated and commented upon.

Finally, opportunities for future research work are briefly discussed.

5.2. IMPLEMENTATION

The completed and fully operational SCS scheduling model is currently being implemented at SCS. The most important implementation aspects can be summarised as follows:

- Technical aspects: The input interface from the model to the input data, and the output interface from the model to the control room operator.
- Human related aspects: The change management and the training given to control room operators.

5.2.1. Technical implementation aspects

The technical aspects in preparation for the implementation revolved around two main factors:

- The programming of interfaces between GAMS, the available input data and the user.
- The format in which scheduling results are presented to the user.

a. Interfaces

The input data for the scheduling model is extracted from the real-time information system at SCS. The extracted data is available in MS Excel format. However, the required input and the output given by GAMS are in text-file (.txt) format.

An input interface was programmed in Visual Basic to translate input data from an MS Excel format to the correct GAMS format with the correct syntax. Similarly, an output interface was designed in MS Excel to translate the GAMS input into a user-friendly schedule.

The ease of use and the stability of the input and output interfaces are critical success factors for the implementation of the SCS scheduling model. These factors determine the usability of the model, and by implication contributes significantly to the value added by this model.

b. Output

The advantage of using MS Excel as an output user interface is the flexibility with which data is presented. The input and output sheets were customised to fit the exact needs of the control room operators and the SCS management team.

The schedule resulting from the model solution is presented to the control room operators in a Gantt chart format. Examples of the stacking and reclaiming schedules are presented in Tables 5.1 and 5.2.

An example of the detail status of the mine bunkers as well as a summary of the coal supplied to the factory are presented in Tables 5.3 and 5.4.

More detailed output statistics such as the strategic stockpile throw out and load back, are not included in this document. However, an example of the input sheets is presented in Appendix B.

Table 5.1: Gantt chart presenting the schedule for stacking

Sasol Coal Supply Scheduling Model Output

Output Sheet 1: Stacking action list and Gantt chart

	Event	Destination	Total tons Conveyed	Speed (t/hr)	Start time	Finish time	Duration
y1							
Tw	p3	sp4	4010	1798	8:46	11:00	2:13
y2							
y3							
Br	p1	sp4	3140	2000	6:00	7:34	1:34
y4							
Syf	p1	sp1	2200	1606	6:00	7:22	1:22
Syf	p3	sp4	3720	1597	8:40	11:00	2:19
y5							
Mdl	p1	sp1	1770	1204	6:00	7:28	1:28
Bo	p3	sp4	4660	2000	8:40	11:00	2:19
y6							
Tw	p1	sp4	1920	1794	6:00	7:04	1:04
Br	p3	sp4	4660	2000	8:40	11:00	2:19

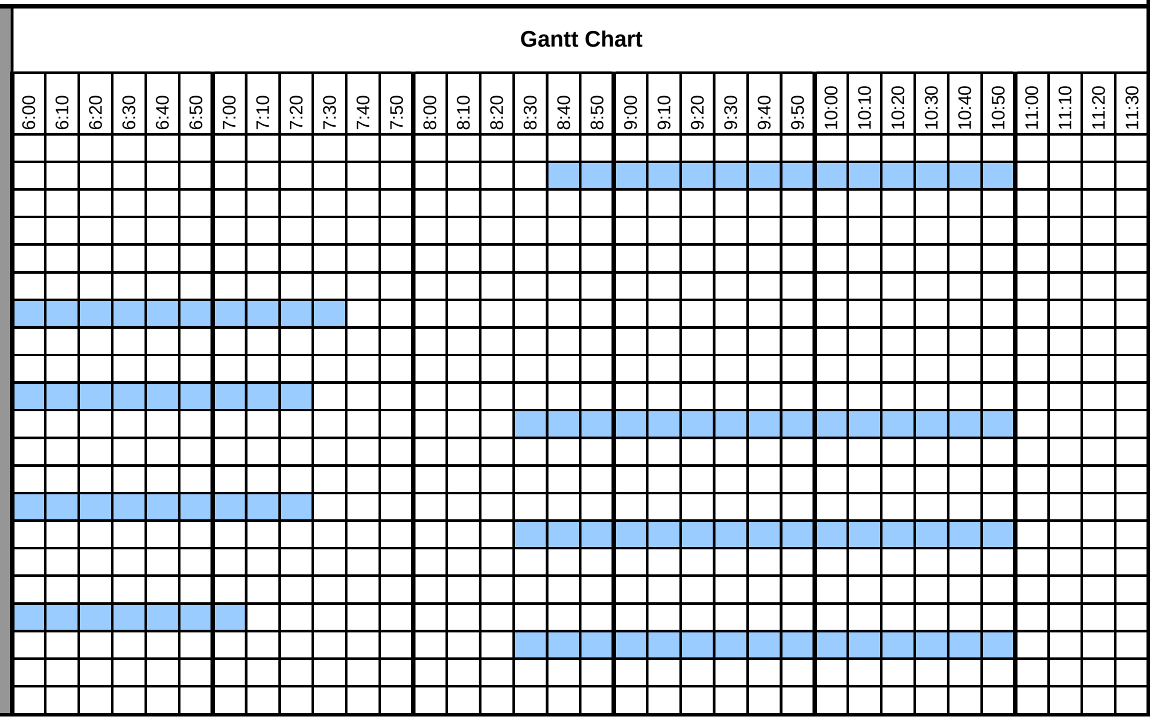


Table 5.2: Gantt chart presenting the schedule for reclaiming and bypassing

Sasol Coal Supply Scheduling Model Output

sasol reaching new frontiers

Output Sheet 2: Reclaiming action list and Gantt chart

	Event	Source	Total tons Conveyed	Speed (t/hr)	Start time	Finish time	Duration	6:00	6:10	6:20	6:30	6:40	6:50	7:00	7:10	7:20	7:30	7:40	7:50	8:00	8:10	8:20	8:30	8:40	8:50	9:00	9:10	9:20	9:30	9:40	9:50	10:00	10:10	10:20	10:30	10:40	10:50	11:00	11:10	11:20	11:30
WEST:																																											
y1																																											
y2	p2	sp3	433	866	6:00	6:30	0:30																																				
	p3	sp2	6400	1600	7:00	11:00	4:00																																				
y3	p3	sp2	8000	2000	7:00	11:00	4:00																																				
EAST:																																											
y4	p2	sp2	2784	1600	6:00	7:44	1:44																																				
	p3	sp2	5226	2002	8:14	10:51	2:36																																				
y5	p2	sp2	3480	2000	6:00	7:44	1:44																																				
	p3	sp3	4182	1602	8:14	10:51	2:36																																				
y6																																											

Table 5.3: Bunker monitoring charts presenting the detail actions per bunker

Sasol Coal Supply Scheduling Model Output

sasol
reaching new frontiers

Output Sheet 3: Bunker monitor charts

Chart per mine bunker indicating mine production (t), possible bunker throw-outs, extraction destination yards (t) and bunker levels

Event points		p1	p2	p3	p4
Br	Mine production	1347.13	0	1902.88	0
	Throw out	0	0	0	0
	y1	0	0	0	0
	y2	0	0	0	0
	y3	3145	0	0	0
	y4	0	0	0	0
	y5	0	0	0	0
	y6	0	0	4655	0
	Level %	**90**	**75.02**	**75.02**	**52.08**
Mb	Mine production	0	661.13	933.87	0
	Throw out	0	0	0	0
	y1	0	0	0	0
	y2	0	0	0	0
	y3	0	0	0	0
	y4	0	0	0	0
	y5	0	0	0	0
	y6	0	0	0	0
	Level %	**30**	**30**	**35.51**	**43.29**
Bo	Mine production	0	1432.1	2022.9	0
	Throw out	0	0	0	0
	y1	0	0	0	0
	y2	0	0	0	0
	y3	0	0	0	0
	y4	0	0	0	0
	y5	0	0	4655	0
	y6	0	0	0	0
	Level %	**40**	**40**	**51.93**	**30**

Event points		p1	p2	p3	p4
Tw	Mine production	763.93	831.77	1984.3	0
	Throw out	0	0	0	0
	y1	0	0	4009.5	0
	y2	0	0	0	0
	y3	0	0	0	0
	y4	0	0	0	0
	y5	0	0	0	0
	y6	1920.5	0	0	0
	Level %	**50**	**37.54**	**50.12**	**30**
Syf	Mine production	1483.91	0	2096.09	0
	Throw out	0	0	0	0
	y1	0	0	0	0
	y2	0	0	0	0
	y3	0	0	0	0
	y4	2196	0	3724	0
	y5	0	0	0	0
	y6	0	0	0	0
	Level %	**60**	**55.25**	**55.25**	**44.4**
Mdl	Mine production	937.98	382.2	1864.82	0
	Throw out	0	0	0	0
	y1	0	0	0	0
	y2	0	0	0	0
	y3	0	0	0	0
	y4	0	0	0	0
	y5	1767	0	0	0
	y6	0	0	0	0
	Level %	**70**	**53.42**	**61.06**	**98.36**

Table 5.4: Factory monitoring statistics and blend detail

Sasol Coal Supply Scheduling Model Output

sasol reaching new frontiers

Output Sheet 4: Synfuels supply and Blend statistics

Total coal supplied to Synfuels

WEST	p1	p2	p3	p4	TOTAL
Reclaimed	0	430	14400	0	**14830**
Bypassed	0	0	0	0	**0**
Total	0	432	14400	0	**14832**

EAST	p1	p2	p3	p4	TOTAL
Reclaimed	0	6260	9410	0	**15670**
Bypassed	0	0	0	0	**0**
Total	0	6264	9408	0	**15672**

Blend supplied to Synfuels compared to the planned blend

WEST:	PLAN %	p1	p2	p3	p4
Total feed (t)		0	432	14400	0
Br	**21**		**27.03**	**32.6**	
Mb	**22**		**27.03**	**12.86**	
Bo	**12**		**15.45**	**26.61**	
Tw	**9**		**15.45**	**6.65**	
Syf	**20**		**7.71**	**9.98**	
Mdl	**9**		**3.47**	**4.64**	
Bleedin	**7**		**3.87**	**6.65**	

EAST:	PLAN %	p1	p2	p3	p4
Total feed (t)		0	6264	9408	0
Br	**15**		**24.74**	**27.33**	
Mb	**15**		**23.46**	**21.27**	
Bo	**25**		**22.16**	**18.65**	
Tw	**8**		**13.22**	**11.07**	
Syf	**21**		**11.08**	**13.11**	
Mdl	**10**		**3.63**	**3.9**	
Bleedin	**6**		**1.71**	**4.66**	

5.2.2. Human related implementation aspects

Prior to this project, no schedules existed at SCS to guide operational activities. Therefore, the implementation of this scheduling model required a lot of human related actions. A change management process was followed to ensure that the operators, the supervisors and the managers understand and agree with the advantages and user requirements when using the scheduling model.

Training and coaching sessions were presented where the relevant parties was trained and given the opportunity to experiment with the scheduling model. Different scenarios as well as real-time production data were used during these sessions.

This aspect of the scheduling model's implementation is still in progress.

5.3. VALUE ADDED TO SCS

The SCS scheduling model is still in an implementation phase, thus making it difficult to report quantified improvements at SCS. However, since scheduling of the SCS operations was a total new concept to everyone at SCS, the very existence of the scheduling model opened a whole new paradigm for the optimal management of SCS, which cannot be quantified in Rand value at this stage in the project.

If the proposed optimised actions from the scheduling model are adhered to, SCS will definitely experience the following value added to the day-to-day operations:

- The bunkers will be managed optimally. Throw outs at the bunkers will be minimized and the maximum possible amount of coal will be extracted to the stockpile yards.
- The blend limits will be adhered to and the reclaiming operations will be scheduled to minimize variation from the blend plan.
- Stacking and reclaiming sequences, incorporated with bypass and strategic throw out and load back options, will be optimized.
- The maintenance and operational schedules will be integrated and optimized.
- The factory's demand per day will be adhered to.
- The blend of the supplied coal will be predicted for the length of the scheduling time horizon. The factories never had predicted blend information prior to this project.
- Machine runtime will be minimized, resulting in minimized operational cost.

The SCS scheduling model has the potential to transform the way SCS is operated, to improve the blend management and to add significant value to the Sasol Group.

5.4. TECHNIQUE EVALUATION

The unit-specific event based time representation was applied to the SCS problem. From the start the model had to be divided into periods (compromising the very advantage of the unit-specific continuous time method) to account for the variation in the mines' production rates. The results were very disappointing. The model did not solve within a time fit for operational use. The size of the model was too big for the specific mathematical formulation described in Chapter 3. The formulation resulted in too many binary variables and a complex model structure in terms of sequencing constraints.

In Chapter 4 various improvement techniques were applied to the scheduling model. The reduction of binary variables by using SOS1 variables, the restructuring of the timing constraints, the reduction in time points by solving a single-period model and the improvement of the NLP solution time were the most important contributions. The improved model's solution time was very fast, but global optimality may have been compromised (refer to section 4.5).

The conclusion from the results and the effort it took to improve the model solution time can be summarised as follows:

The unit-specific event based MINLP continuous time formulation method, as presented in the literature, is not robust enough to be applied to an operational industrial-sized scheduling problem such as the SCS problem. Customised modifications to the formulation were necessary to ensure that the model solved in a time acceptable for operational use.

However, it has been proved that Mixed Integer Non-linear Programming (MINLP) can be successfully applied to optimise the scheduling of an industrial-sized plant such as Sasol Coal Supply. Although more research is required to derive robust formulation techniques, the principle of using mathematical methods to optimise operational scheduling in industry can dramatically impact the way plants are operated. The optimisation of daily schedules at SCS, by applying the MINLP continuous time scheduling technique, has made a significant contribution to the coal handling industry.

5.5. FUTURE RESEARCH WORK

5.5.1. SCS specific application

a. Schedule according to real-time coal quality

The SCS scheduling model may be expanded in future to include the coal quality properties of the different coal sources. The optimal scheduling of operations, based on the coal qualities and its variation, may have a significant impact on the gas volume produced at Synfuels. In the future, real-time quality information for each source will be available. This will present an ideal opportunity to integrate the scheduling model with a Statistical Process Control model screening the quality deviations and prompting the scheduling model when coal quality has changed significantly.

b. Improved mine production forecasting

The mines' production has a critical impact on the accuracy and usability of the scheduling model. Therefore it is recommended that more research be done to forecast the mines' production rates on a real-time basis. Positive and negative deviations in mine production must pro-actively be forecasted and the operational schedule adjusted accordingly.

c. Scheduling model input extended to the mines' activities

A long-term recommendation is to incorporate the different mines' activities as an input into the scheduling model. Mining sections' production rates, underground bunker levels, maintenance downtimes and equipment breakdowns are examples of information that have an influence on the supply of coal to be scheduled at SCS. The incorporation of this information will further enhance optimised scheduling at SCS.

5.5.2. General scheduling research

As stated above, the robustness of the unit-specific event based formulation needs to be researched further. The application of the formulation needs to be tested on industrial-sized problems, not only in the chemical industry (used in most of the available literature), but also in other industries. Some of the aspects to be included in possible further research in the scheduling field are:

- Improved formulation of binary variables to improve the size of industrial-sized models.
- Improved time representation and sequencing to improve the structure of large models.

5.6. CONCLUSION

In this document, the specific scheduling problem at Sasol Mining's coal handling facility, Sasol Coal Supply (SCS), was presented and solved using Mixed Integer Non-Linear Programming (MINLP) continuous time representation techniques.

The most recent MINLP scheduling techniques were presented and applied to an example problem. The assumption was made that the results from the example problem will display trends which will apply to the SCS scheduling problem as well. Based on this assumption, the unit-specific event based continuous time formulation was chosen to apply to the SCS scheduling problem.

The detail mathematical formulation of the SCS scheduling problem, based on the chosen technique, were discussed and the necessary changes were made to customize the formulation for the SCS situation. This first phase model did not solve within 72 hours, which is not acceptable for operational use.

Various improvement approaches were applied during the second phase of the model development. Special Ordered Sets of Type 1 (SOS1) variables were successfully applied in the model to reduce the amount of binary variables. The time and duration constraints were restructured to simplify the structure of the model. A specific linearization and solution technique was applied to the non-linear equations to ensure reduced model solution times and reliable results.

The improved model for one period solved to optimality within two minutes. This dramatic improvement ensured that the model will be used operationally at SCS to optimise daily operations. The scheduling model is currently being implemented at SCS. Examples of the input and output from the operational model were presented.

Finally, it can be concluded that the SCS scheduling problem was successfully modelled and the operational scheduling model will add significant value to the Sasol Group.

LIST OF REFERENCES

Brooke, A., Kendrick, D., Meeraus, A., Raman, R. (1998). GAMS, a user's guide. *GAMS Development Corporation, Washington D.C.*

Floudas, C. A., Lin, X. (2004). Continuous-time versus discrete-time approaches for scheduling of chemical processes: a review. *Computers and Chemical Engineering, vol 28, p2109-2129.*

Glover, E. (1975). Improved linear integer programming formulations of nonlinear integer problems. *Management Science, vol 22, p455-460.*

Grossmann, I. E., Viswanathan, J., Vecchietti, A., Raman, R., Kalvelagen, E. (2002). DICOPT: A discrete continuous optimisation package. *Engineering Research Design Centre, Carnegie Mellon University, Pittsburgh, PA. GAMS Development Corporation, Washington D.C.*

Ierapetritou, M. G., Floudas, C. A. (1998a). Effective continuous-time formulation for short-term scheduling. Part 1. Multipurpose batch processes. *Industrial and Engineering Chemistry Research, vol 37, p4341-4359.*

Ierapetritou, M. G., Floudas, C. A. (1998b). Effective continuous-time formulation for short-term scheduling. Part 2. Continuous and semi-continuous processes. *Industrial and Engineering Chemistry Research, vol 37, p4360-4374.*

Kondili, E., Pantelides, C. C., Sargent, R. W. H. (1993). A general algorithm for short-term scheduling of batch operations. Part 1. MELP formulation. *Computers and Chemical Engineering, vol 17, p211-227.*

Majozi, T. (2003). Memorandum on the acquisition of GAMS within MDAS. *Sasol Technology official document.*

McCormick, G. P. (1976). Computability of global solutions to factorable nonconvex programs – Part 1 – convex underestimating problems. *Mathematical Programming, vol 10, p146-175.*

Pinto, J. M., Grossmann, I. E. (1995). A continuous time mixed integer linear programming model for short term scheduling of multistage batch plants. *Industrial and Engineering Chemistry Research, vol 34, p3037-3051.*

Quesada, I., Grossmann, I. E. (1995). Global optimisation of bilinear process networks with multicomponent flows. *Computers and Chemical Engineering, vol 19, p1219-1242.*

Zhang, X., Sargent, R. W. H. (1996). The optimal operation of mixed production facilities. Part A. General formulation and some approaches for the solution. *Computers and Chemical Engineering, vol 20, p897-904.*

Zhu, X. X., Majozi, T. (2001a). A novel continuous-time MILP formulation for multipurpose batch plants. Part 1. Short-term scheduling. *Industrial and Engineering Chemistry Research, vol 40, p5935-5949.*

Zhu, X. X., Majozi, T. (2001b). A novel continuous-time MILP formulation for multipurpose batch plants. Part 2. Integrated Planning and Scheduling. *Industrial and Engineering Chemistry Research, vol 40, p5621-5634.*