An MILP approach to optimizing capacity on rail lines: A NatCor case study

Nolusindiso Ntwana (1231115)

Supervisor: Dr Joke Bührmann

A research dissertation submitted to the Faculty of Engineering and the Built Environment, University of the Witwatersrand, in fulfilment of the requirements for the degree of Master of Science in Engineering.

Johannesburg, March 2023

Plagiarism Declaration

I Nolusindiso Ntwana with student number 1231115 am a student registered for the degree of Master of Science in Industrial Engineering, full research, in the academic year 2023.

I hereby declare the following:

- I am aware that plagiarism (the use of someone else's work without their permission and/or without acknowledging the original source) is wrong.
- I confirm that the work submitted for assessment for the above degree is my own unaided work except where I have explicitly indicate otherwise.
- I have followed the required conventions in referencing the thoughts and ideas of others.
- I understand that the University of the Witwatersrand may take disciplinary action against me if there is belief that this is not my own unaided work or that I have failed to acknowledge the source of the ideas or words in my writing.

Signature:____________________ Date: 14 March 2023

Abstract

Rail capacity planning and management is becoming a topic of interest around the world. Rail companies are now researching ways to increase capacity and are therefore investing resources in architecture expansion, which may result in high capital costs. There has been a lack of research and resources invested in optimising existing capacity, which could save these extra costs.

Using the Natal Corridor (NatCor) as a case study, this research analyses the capacity on the rail line and the effects that optimisation of the line will have on the Durban Container Terminal (DCT). The research investigates the current state of NatCor as a train timetabling problem (TTP). A mathematical model is proposed using mixed integer linear programming (MILP) in order to optimise the line. The TTP is then converted into an equivalent vehicle routing problem with time windows (VRPTW). Current schedules of the NatCor are used as an input to the model and the model is solved using Python and MS Excel Open Solver. The initial feasible solution is improved using the Tabu Search (TS) algorithm.

The results suggest that frequent updates to company data will help keep the train timetables clear of any cancelled trains, which take up capacity on the line. The MILP models provided an optimal maximum travel time that trains could incur on the line without causing any further delays to operations at the DCT. The models will contribute to future research on similar topics and can be further developed to model large rail networks.

Dedication

In memory of my grandmother,

Nowine Elizabeth Bilatyi.

Forever number one in my heart.

Acknowledgements

I would first like to thank my dissertation supervisor Dr Jacoba Bührmann of the School of Mechanical, Industrial and Aeronautical Engineering at The University of the Witwatersrand. I appreciate the time she took to help me put this dissertation together – always ensuring that the research was my own work.

I would also like to thank my colleagues Mr Letlotlo Phohole, Mr Nic Cloete-Hopkins and Mr Paul Bester for their unrelenting support. I am grateful to them for helping me contact the experts who were involved in the data collection and data cleaning process for this research project: Mr Nkosinathi Gumede and Miss Sentse Ramosolo from the Operations Department of Transnet.

I would also like to acknowledge Mr Phakamani Mavela and Mr Tshepo Molebiemang for assistance with the Python programming.

Lastly, I would like to thank my family for all the support they have provided to me. Special mention goes to my parents Peliwe and Vuyani Ntwana and my sister, Bukiwe Ntwana.

Table of Contents

Abstract i

Dedication iii

Acknowledgements iv

Table of Contents v

Table of Figures ix

List of Tables xi

List of Acronyms/Initialisms xii

1. Introduction 1

1.1 Research background 1

1.2 Problem statement 11

1.3 Critical research question 12

1.4 Objectives 12

1.5 Importance of the study 12

1.6 Delimitations and limitations 13

2. Literature review 15

2.1 Operations research methods 15

2.1.1 Linear programming 16

2.1.2 Integer programming 16

2.1.3 Mixed integer linear programming (MILP) 17

2.1.4 Mixed integer non-linear programming 18

2.1.5 Goal programming 19

2.1.6 NP, NP-complete and NP-hard combinatorial problems 19
2.2 Solution techniques for combinatorial problems 21
2.2.1 Exact methods 21
2.2.2 Heuristics 25
2.2.3 Metaheuristics 27
2.3 Optimising rail line capacity 31
2.3.1 Train types 32
2.3.2 Schedules 34
2.3.3 Infrastructure 36
2.3.4 Simultaneous parameter modelling 37
2.4 Train timetabling problem models for train schedules 39
2.4.1 Mathematical formulation of the TTP 39
2.4.2 TTP solution methods 43
2.5 Vehicle routing problem models for train schedules 46
2.5.1 Mathematical formulation of the VRPTW 47
2.5.2 VRP solution methods 49
2.6 Summary 51

3. Methodology 53

3.1 Research approach 54
3.2 Data collection 55
3.3 Data analysis approach 57
3.3.1 Data correlation 57
3.3.2 Data cleaning 57
3.3.3 Model formulation 58
3.3.4 Model verification 59
3.3.5 Validation 59

3.4 Input data 59
3.4.1 Line variables 59
3.4.2 Train types 61
3.4.3 Train slots 63
3.4.4 Current train schedule 66
3.4.5 Final train schedule 66
3.4.6 Train paths 70

4. Model development 72

4.1 Model assumptions 72
4.2 Data sets, parameters and variables' definition 72
4.3 The MILP Formulation 74
4.4 Converting the TTP to a VRPTW 75

5. Results 77

5.1 Time matrices 77
5.2 Timetable results 80
5.3 MILP models' results 83
5.3.1 Results from the model using Table 6 83
5.3.2 Results from clean timetable data 85
5.3.3 Results using the old timetable 85
5.3.4 Model verification 86
5.3.5 Model validation 87
5.4 Results using the Tabu Search metaheuristic 88
5.5 Results interpretation 93
5.6 The theoretical effect of the improved timetable on the DCT 94

6. Conclusion and recommendations 97

6.1 Conclusion..97
6.2 Recommendations 100

7. References...102

Addendum A – Catalogue 114

Addendum B – Clean timetable..118

Addendum C – All Possible Paths on the NatCor 120

Addendum D – Train allocation to a path...123

Table of Figures

Figure 1: Transnet freight rail network (Transnet Freight Rail, 2021). 3

Figure 2: The TFR corridor operating model (Transnet Freight Rail, 2021)............................ 4

Figure 3: A map of major freight demand areas (Presidential Infrastructure Coordination Commission, 2013). 5

Figure 4: A map of the NatCor line (Transnet, 2016)... 6

Figure 5: A map of the NatCor to show feeder lines (Transnet, 2019). 7

Figure 6: An aerial photo of the DCT (Tristan, 2015). 9

Figure 7: The DCT capacity constraints (Transnet SOC Ltd, 2016). 10

Figure 8: Euler diagram (Esfahbod, 2007). 21

Figure 9: Pseudocode for a VNS for a course timetabling problem (Abdullah et al., 2005)... 27

Figure 10: A simulated annealing algorithm flow chart (Corana et al., 1987). 30

Figure 11: An image of the parameters affecting capacity (Landex, 2008). 32

Figure 12: Railway network scheduling process (Luethi et al., 2007). 35

Figure 13: A three-dimensional representation of capacity utilisation (Landex, 2008). 38

Figure 14: Flow chart for the branch-and-price algorithm (Liu and Han, 2016)..................... 44

Figure 15: A flow chart of the TTP heuristic (Lee & Chen, 2009). 45

Figure 16: The GA pseudocode for a VRP (Ombuki et al., 2004). 50

Figure 17: the operations research framework (Zandin and Maynard, 2000). 53

Figure 18: The Research Design Onion (Saunders et al., 2006)... 55

Figure 19: An illustration of how trains run on the NatCor line (Transnet, 2019). 56

Figure 20: Train types classified by comby commodity 62

Figure 21: A classification of trains by length 63

Figure 22: The NatCor slots availability analysis (Transnet, 2019). 65

Figure 23: Current NatCor train schedule diagram (Transnet, 2019). 67

Figure 24: Train schedule derived from a clean timetable. 69

Figure 25: A screen shot of part of the Python code used to calculate train paths. 70

Figure 26: A screen shot of the Python, which assigns trains to paths 71

Figure 27: Schedule using distance matrix. 79

Figure 28: A screenshot of the $\boldsymbol{x i, j}$ matrix for train T1 84

Figure 29: A screenshot of the variables, constraints and objective function. 84

Figure 30: A summary of the results 86

Figure 31: TS algorithm steps 89

Figure 32: Iterations for the Rietvallei to Durban path 90

Figure 33: Iterations for the Newcastle to Durban path 91

Figure 34: A comparison of all results obtained for the minimum total travel time 93

Figure 35: A causal loop diagram of the NatCor and DCT delays. 96

Figure 36: A graph summarising the final results 99

List of Tables

Table 1: A list of variables for the VRPTW (Akgul , 2020)... 47

Table 2: A summary of the NatCor line characteristics.. 60

Table 3: The NatCor line sections and length (Transnet, 2019)... 60

Table 4: Current average travel times for trains on the NatCor.. 68

Table 5: Variables and parameters for the TTP model... 72

Table 6: Travel times between stations (in hours).. 78

Table 7: Train delays experienced in real-time.. 80

Table 8: A matrix of new estimated travel times... 88

Table 9: Travel time matrix for the path from Rietvallei to Newcastle.................................. 90

Table 10: Travel time matrix for the path from Newcastle to Durban 91

Table 11: Travel time matrix for the path from Danskraal to Durban..................................... 92

Table 12: Total travel time improvement 94

List of Acronyms/Initialisms

B&B	Branch and bound
CBC	COIN-OR branch and cut
COIN-OR	Computational Infrastructure for Operations Research
DCT	Durban Container Terminal
DP	Dynamic programming
GA	Genetic algorithm
GP	Goal programming
IP	Integer programming
LP	Linear programming
LTPF	Long Term Planning Framework
MIBLP	Mixed integer bi-level linear programming
MILP	Mixed integer linear programming
MINLP	Mixed integer non-linear programming
MS	MS
NatCor	Natal corridor
NNH	Nearest neighbour heuristic
NP	Non-deterministic polynomial
OR	Operations research
SA	Simulated annealing
TS	Tabu search
TSP	Travelling salesman problem
TTP	Train timetabling problem
VNS	Variable neighbourhood search
VRP	Vehicle routing problem
VRPTW	Vehicle routing problem with time windows

1. Introduction

1.1 Research background

Railroads have become an attractive transportation mode with a variety of applications such as large freight and passenger transportation. The freight includes commodities such as iron ore, coal, motor vehicles, bulk products and manganese ore. As the world becomes more aware of various issues affecting the environment, such as increased air pollution, road traffic congestion and global warming, the attraction of rail transportation has increased. This is because trains have become a popular alternative to road transportation given that such transport avoids being stuck in road traffic. Trains are more environmentally friendly and can receive power through alternative energy sources such as electricity and, most recently, solar energy.

The increased popularity of rail has led to rail line and yard capacity planning becoming an area of interest for rail companies across the world. The main goal for transport freight rail is to attract the maximum number of customers. As rail transportation demand increases, the traffic load (cargo being transported) increases, causing the rail line (running train tracks that form a route between two points) and the rail yard (where trains are stored, loaded, unloaded, shunted and assembled) capacities to become insufficient. This insufficiency has prompted companies to research ways to increase capacity.

'Capacity' has several different definitions. The term is sometimes associated with a weight or measurement depending on different techniques and objectives. Examples of the definition of capacity are that capacity indicates:

- ''a measure of the ability to move a specific amount of traffic over a defined rail corridor in the U.S. rail environment with a given set of resources under a specific service plan, known as level of service (LOS)'' (Barkan & Lai, 2005).
- "The count of valid train paths over a fixed time horizon within an optimal master schedule" (Harrod, 2009).

- “The maximum number of trains that may be operated using a defined part of the infrastructure at the same time as a theoretical limiting value is not reached in practice” (Hansen & Pachl, 2008).
- “The number of trains that will cause the system to lock up” (Kieran, 2001).
- “the level of traffic (i.e. number of trains per day) that a rail line can accept without exceeding a specified limit of queuing time” (Peat Marwick and Partners, 1977).
- “The ability of the carrier to supply as required the necessary services within acceptable service levels and costs to meet the present and projected demand” (Kahan, 1979).

In this research, capacity is defined as the maximum number of trains running on the rail line within a 24-hour time window. This definition is a combination of the Hansen and Pachl (2008) and Peat Marwick and Partners (1977) definition. This definition is based on the available train slots per day and these are unique for every line or corridor.

Transnet SOC Ltd, is a South African state-owned bulk freight transport and logistics company. Transnet comprises of rail, ports, engineering, terminals and pipeline businesses. The company has noted Transnet Freight Rail (TFR) as its largest Operating Division (OD). According to Transnet Freight Rail (2021), the OD provides rail network infrastructure and controls rail services over major rail corridors. These corridors are used to transport different commodities for export, and local markets. TFR manages and maintains a complex rail network which extends across South Africa. The network is made up of approximately 31000 track kilometres (20911 route kilometres) over which commodities are hauled by locomotives in wagons specifically suited to the commodity type. The rail network is diverse and consists of approximately 1500 km heavy-haul lines. The network also consists of 3928 km of branch lines that serve as feeders to main lines. The TFR network is illustrated in Figure 1.

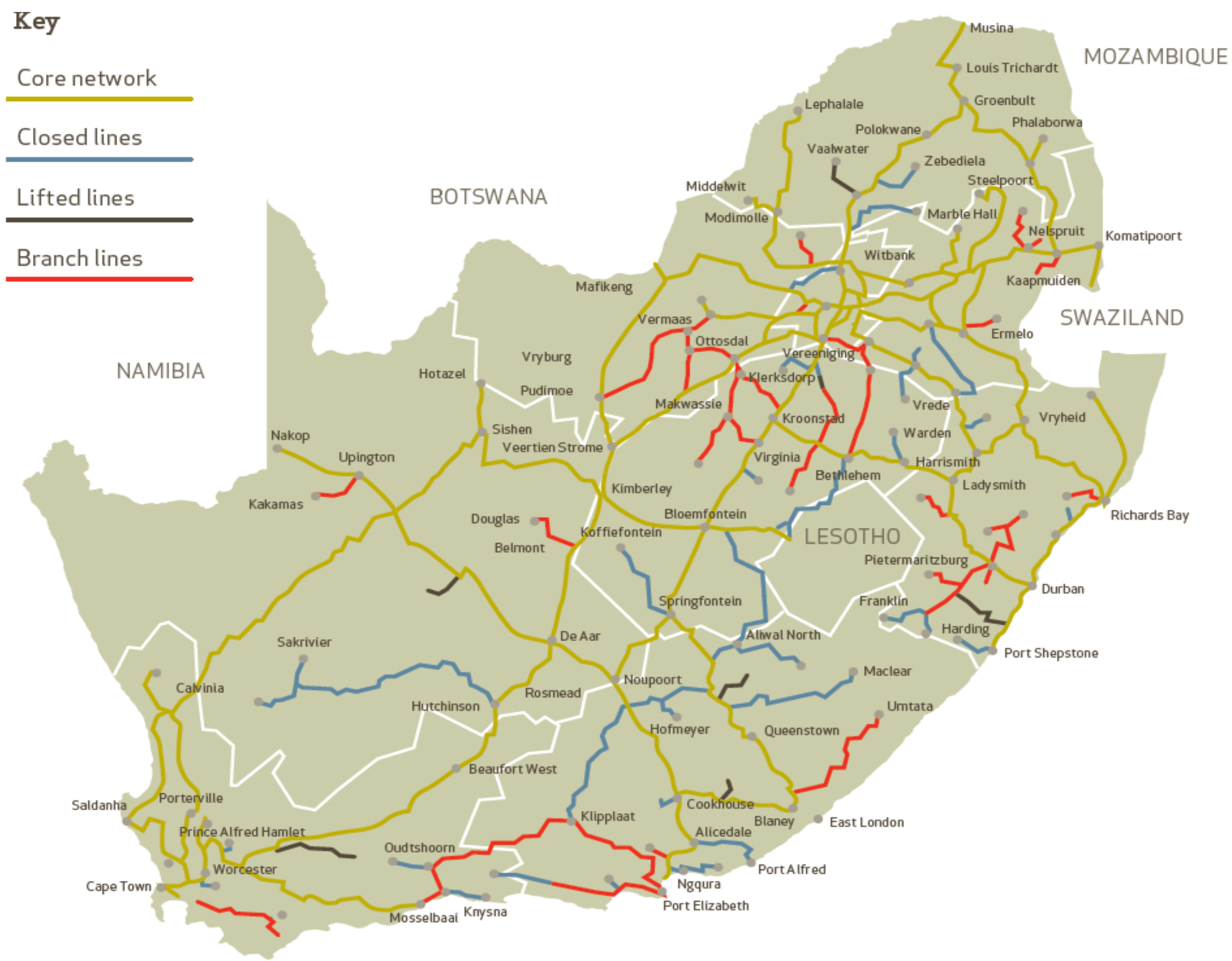


Figure 1: Transnet freight rail network (Transnet Freight Rail, 2021).

TFR has applied the Corridor Operating Model as a response to fast changing business, market and policy environments. The operating model was implemented to improve the company's performance and competitiveness. This model was created to improve decision-making, responsiveness to customer needs and integrated problem solving, the latter being one of the company's major challenges. Each corridor in the model is managed by an executive. Additionally, the model highlights the close alignment of freight rail corridors with port terminals and the company ceased this as an opportunity to improve operational connections and to free up capacity for growth of volumes on rail (Transnet Freight Rail, 2021). The corridors and ports are shown in Figure 2.

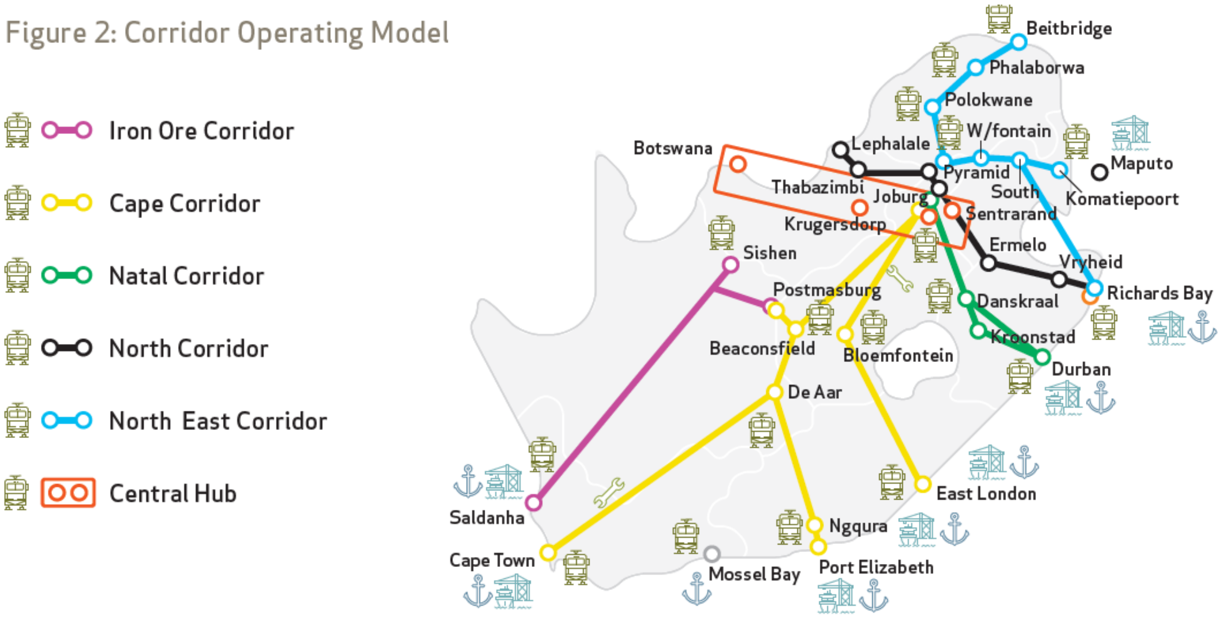


Figure 2: The TFR corridor operating model (Transnet Freight Rail, 2021).

The company has stated that most of the freight rail demand comes from the Gauteng area. Figure 3 illustrates the concentration of economic centres in South Africa, as well as the movement of cargo from main metropolitan areas to areas with limited access to services. From Figure 3, the two major metropolitan areas experiencing these constraints are Gauteng and KwaZulu Natal. These metropolitan areas are connected by the Natal Corridor (NatCor).

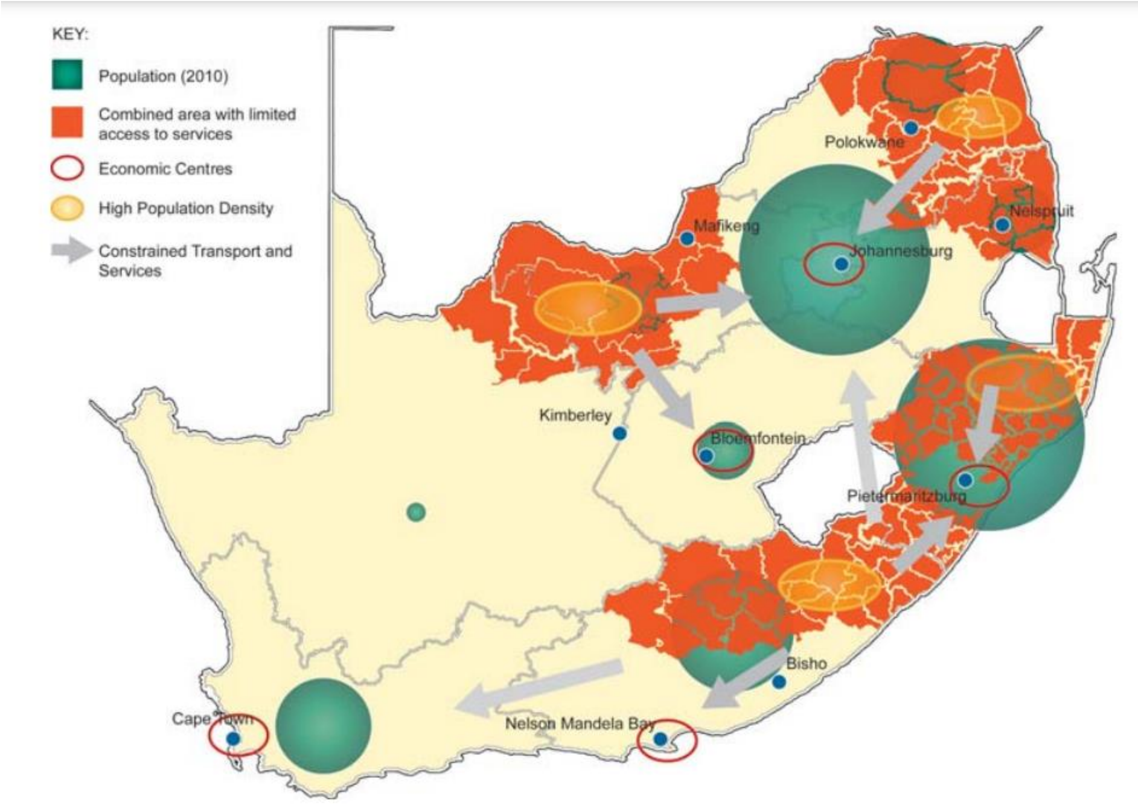


Figure 3: A map of major freight demand areas (Presidential Infrastructure Coordination Commission, 2013).

The NatCor line is a railway line running from Rietvallei, Gauteng, to Durban Container Terminal (DCT) in KwaZulu Natal – as illustrated in Figure 4. In the Long-Term Planning Framework (LTPF), the company forecasts an increase in demand for container trains (Transnet SOC Ltd, 2016). These trains take up more than 50% of the capacity on the NatCor line. The LTPF discusses how investing in current infrastructure and expanding on it can, in the long run, increase capacity on the NatCor line, which will make it possible for the company to increase the number of trains that can run on the line in order to meet increasing demand (Transnet SOC Ltd, 2016).

Figure 4: A map of the NatCor line (Transnet, 2016).

The Natal Corridor is approximately 688 km long (718 km if the yards are included). It is a double-track line, with train movement controlled by colour-light signalling from four central traffic control centres (CTC) in Durban, Ladysmith, Newcastle and Standerton. These are then connected to and monitored by the Operational Control Centre (OCC), which will be replaced by a National Command Centre (NCC) in the next five years (Transnet, 2019).

The trains move at an average speed of 50–80 km/h (50 km/h being for the longer trains). The longest train running on the line is 105 wagons, while the shortest is 40 wagons. The headway for 40 to 65 wagon trains is currently 20 minutes and 30 minutes for the 75–105 wagon trains. Some trains join the NatCor from other rail networks. This is shown by the feeder lines in Figure 5 (Transnet, 2019).

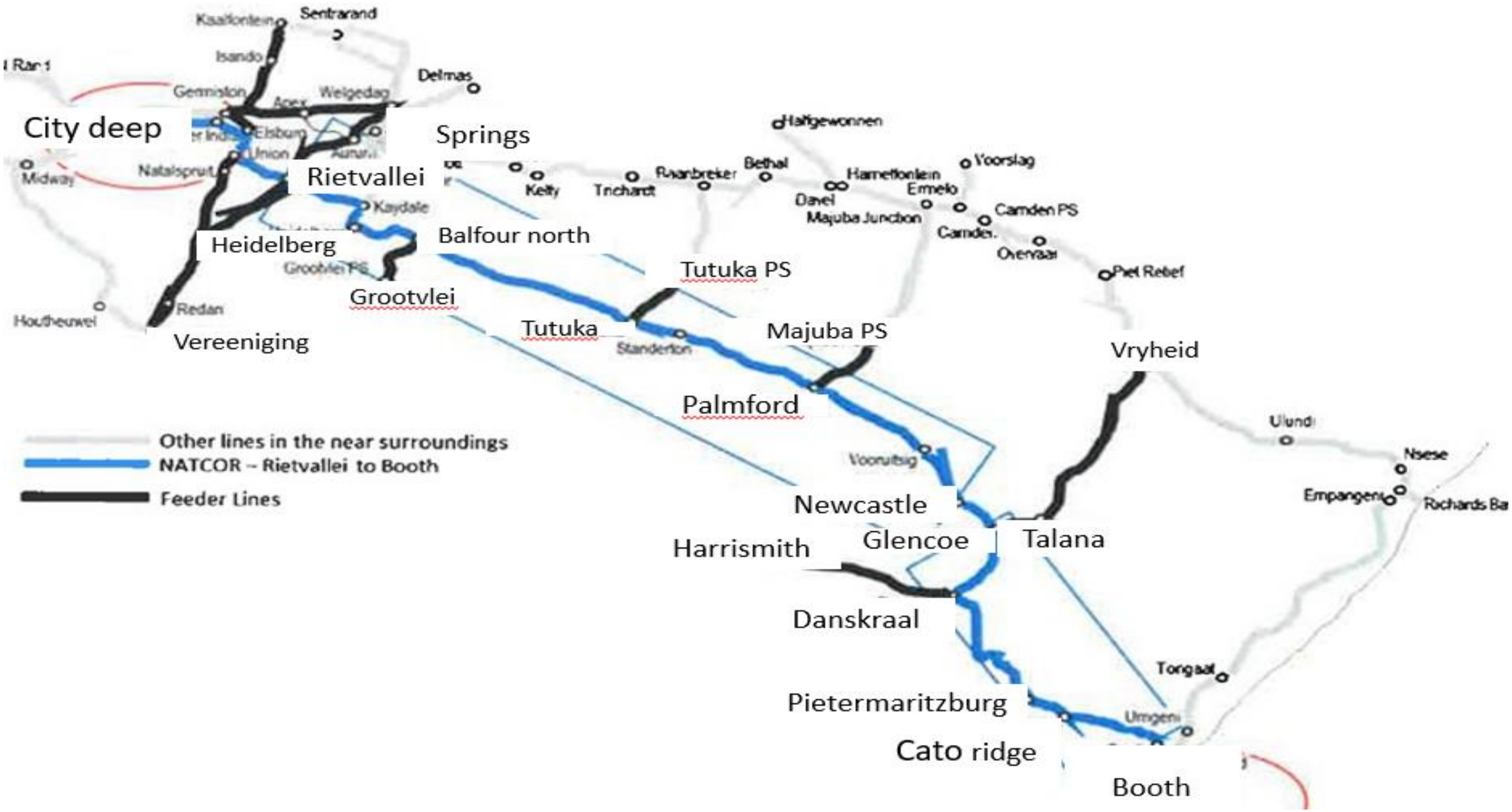


Figure 5: A map of the NatCor to show feeder lines (Transnet, 2019).

When wanting to increase capacity to meet client demands, rail operators find it easier to add another rail line that runs between two towns or cities, to add a track on the existing rail line, or to create more space at the yards essentially only looking at the infrastructure. This can be very expensive and the lack of available vacant land for the creation of space may be a huge constraint. Ideally, freight companies prefer to load trains on the existing lines with minimum investment cost. Other more cost-effective options may include optimisation of personnel use by assigning staff to tasks, optimisation equipment usage, and optimisation of operations and maintenance costs (Hansen, 2010). Luethi et al. (2007) suggested that production-based strategies need to be developed to increase capacity on the railway line. Landex (2009) countered this, suggesting that railway line capacity depends on infrastructure usage, timetables and rolling stock usage.

However, because capacity differs for different types of trains (depending on the length of the train and the commodity being carried) and the sequential order they run in, this research performed an analysis using mathematical modelling to determine the impacts of different train types (in wagon numbers) on the railway line and rail yard. It examined the NatCor and the effects that train delays on this line have on the DCT. Different commodities are transported on the line, requiring different types of rolling stock of different lengths. This research therefore also considered different methodologies, including using real data from the NatCor line to analyse how different factors influence available capacity and train delays, and how these will affect DCT operations.

The DCT is South Africa's biggest container terminal. As with the NatCor line, the increase in container throughput volumes has put some pressure on the DCT. Figure 6 illustrates the current layout of the Durban port and the location of the DCT piers 1 and 2, which are the container terminals, and Salisbury Island, which is the naval base.

Figure 6: An aerial photo of the DCT (Tristan, 2015).

Not only is it currently congested, it has also reached maximum capacity in terms of the number of containers it can accommodate and is struggling to keep up with the market. There is also only one shunting machine working at the terminal, which creates delays in the turnaround time for container trains.

According to the Transnet SOC Ltd (2016), the DCT can no longer accommodate the arrivals and storing of container trains arriving at the yard from the NatCor. Over the years, the DCT has housed container trains, including those that have been discontinued and empty containers. The company has listed the berth capacity and container stacking yard capacity as the most critical constraints to capacity at the DCT. Container crane, road and rail terminal capacity are identified as potential factors that can help decrease the current congestion. These capacity constraints are illustrated in Figure 7.

Figure 7 depicts the congestion of the DCT and confirms that overcrowding at the DCT is not only for trains but also for road vehicles awaiting cargo. This too, has contributed to turnaround times for trains being too long – causing more delays on the NatCor line. Shunting, loading and off-loading of trains at the DCT is then subsequently delayed which in turn increases train dwell times, causing even further delays on the corridor, thus affecting the entire rail network.

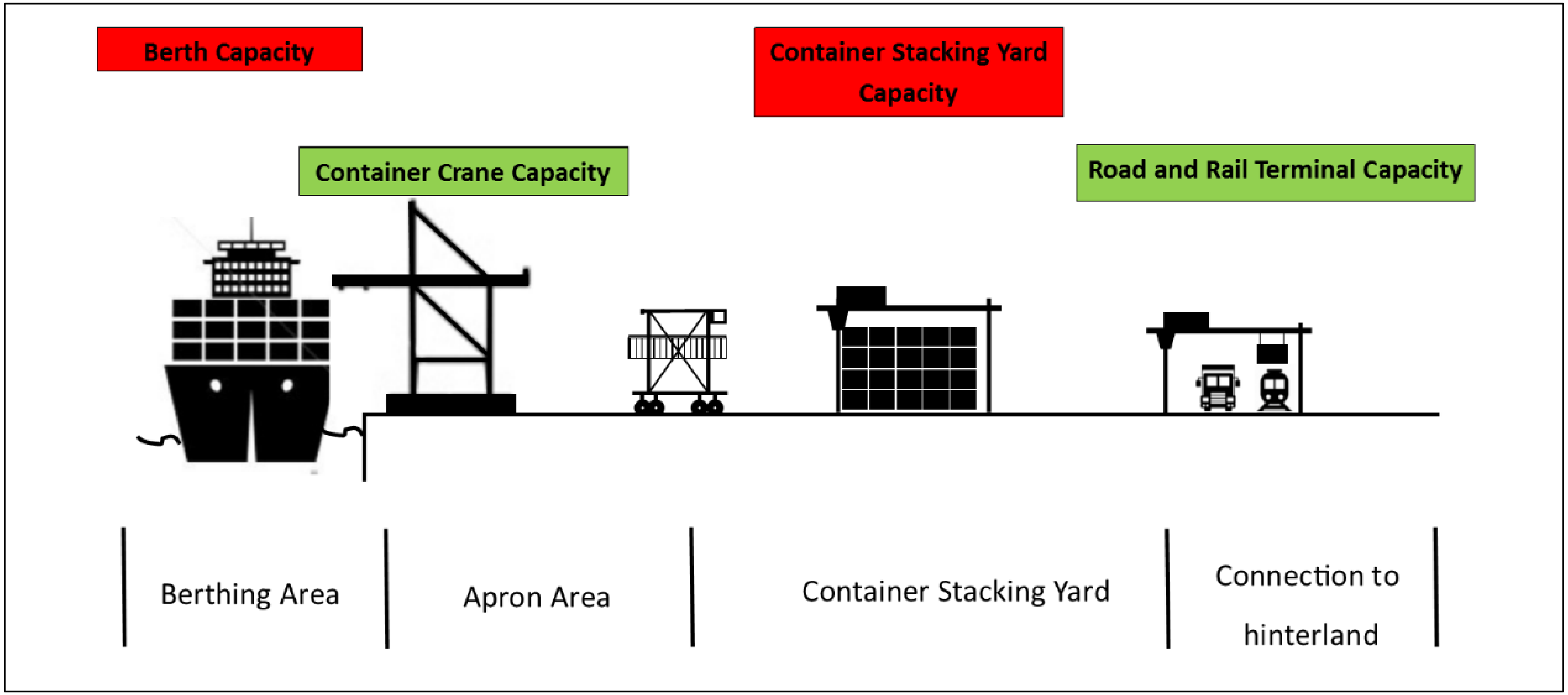


Figure 7: The DCT capacity constraints (Transnet SOC Ltd, 2016).

.

The company has been searching for possible solutions to this problem, including plans to expand the infrastructure at the stacking yards and expand the berth capacity to increase overall capacity at the DCT. This desired increase in infrastructure also included a discussion about using a dry port at a different location for excess containers that cannot be accommodated at the DCT (Scholtz, 2017).

The research presents the NatCor line as a real-life case study. Mathematical modelling and optimisation techniques were used to estimate capacities. The NatCor line was modelled as a train timetabling problem (TTP) and was formulated as a mixed integer linear programming (MILP) model to help achieve the goal of capacity optimisation. This was done without expanding or making any infrastructure changes, and therefore minimising cost using scheduling tools to analyse influences of delays and inter alia traffic density to give an in-depth understanding of railway operations on the line. The TTP was translated into an equivalent vehicle routing problem with time windows (VRPTW) to determine an initial solution for optimising capacity on the line.

1.2 Problem statement

Different commodities with different train lengths run on the rail lines to reach the port. On the NatCor line from and to Durban, container trains (running 50 wagon trains) take up more than 50% of the capacity. The LTPF recorded a 35-year demand plan, which shows that container train lengths will increase to 75 wagons in the next five years. Increasing the number of wagons will require longer headways (the time span between two consecutive trains), which will add to the delays experienced on the line. Transnet's biggest constraint has been that the NatCor line is currently already congested. This has made it impossible to increase the number of wagons and to run longer container trains.

At the DCT, loading and shunting of trains takes too long and has proven to be a major constraint. This has led to delays, overcrowding and congestion. A mathematical model of the line needs to be formulated to determine how to optimise the schedules.

1.3 Critical research question

How can MILP can be used to model and optimise the capacity for the NatCor line to reduce the delays on the NatCor line?

1.4 Objectives

The main objectives of this research are to:

1. Formulate the capacity problem at the NatCor as MILP mathematical models. This formulation helped with the analysis of the corridor and can be used to estimate optimal capacity.
2. Estimate the optimal capacity of the NatCor by using the branch-and-cut method through the COIN-OR branch and cut (CBC) algorithm to solve the formulated MILP models.
3. Attempt to find a better estimate of the objective function by using metaheuristic methods.
4. Analyse and compare the results with the performance of the line from previous years.
5. Determine the effects that the results may have on the DCT at a high-level. This will help with recommending future research on the NatCor line and the DCT.

1.5 Importance of the study

There is currently limited research done on capacity optimisation using mathematical programming in South Africa. To the researcher's knowledge, simultaneous investigations of the lines and effects on yards have never been done. Previous research looked at expanding infrastructure and increasing capacity without considering other options, such as operations efficiency and scheduling. These alternative options may in the long run deliver cost-savings and improve infrastructure efficiencies. This research highlights why looking at these options helps with making an informed decision on where to invest resources.

Another benefit is that the proposed model can be used as a template to help rail companies find optimal ways to meet forecasted demand increases. It can also be used to ensure more effective capacity management. Furthermore, the model can be used to determine which parameters, such as infrastructure and timetables, affect capacity. Dingler et al. (2009) highlighted that being able to identify these significant parameters would help companies maximise the use of their assets, while trying to provide sufficient network capacity to accommodate future demand at the desired service level.

The data collected for the MILP models was used to study the effects that any changes to the performance of the rail line would, theoretically, have on the DCT rail yard. It was therefore important to understand what happened on the NatCor line and the DCT. It was also important to collect data and to analyse the NatCor line and the DCT simultaneously, to ensure that the correct data and assumptions were used, and to validate the accuracy of the data.

1.6 Delimitations and limitations

The case study covered the NatCor line and the effects that the study would have on the DCT. The data of other lines and yards would have clouded the focus of the study and it was therefore left out of the scope of the research – given the time constraints. Investigation of the Durban port was limited to the DCT and the research focused on rail activities that could affect the DCT exclusively.

Of note is that the research was conducted on the existing infrastructure, as the goal was to optimise the use of capacity with the current infrastructure in place. This was done to investigate whether the current infrastructure is sufficient and whether use of resources is efficient.

The study used real company data. Some information was aggregated or left out as it was deemed confidential and detailed company data could not always be revealed. The exclusion of the data did not affect the accuracy of the models and the data accuracy.

The data provided had the following limitations:

- There were inconsistencies in the data provided by the operations and scheduling departments of the company.
- The schedule had trains that were no longer being used and the list of trains in use included some that were not slotted into the provided schedule.
- Some trains on the schedule and its associated timetable were duplicated.

It was therefore clear that a cleaning up process was needed to perform a sanity check of the data – to ensure accuracy. This process is explained in more detail in Chapter 3.

2. Literature review

This chapter reviews how line capacity can be optimised using MILP mathematical modelling and optimisation methods. It begins with a brief explanation of what operations research (OR) is and explains the different OR optimisation methods relevant to this dissertation. It also investigates different parameters that influence rail capacity and how to analyse these using optimisation methods. This is followed by a summary of the findings from the literature. The chapter closes by introducing different problem structures in MILP.

2.1 Operations research methods

Ramamurthy (2007) described OR as a branch of mathematics used to take timely and effective decisions for management problems. It uses logical analysis and analytical techniques to study the behaviour of systems in relation to the overall working considering any operational and functional constraints – alternative decisions are then derived.

Ramemurthy added that OR is a quantitative technique to problem solving that originated during World War II. It consists of optimisation methods such as mixed integer linear programming (MILP), linear programming (LP) and integer programming (IP). These are all used to provide scientific evidence in the decision-making process, with various company parts all working together to obtain optimal decisions. Monks (2015) agreed with this definition and added that OR uses models that can be quantitative and qualitative in order to help with decision-making processes for complex implementation problems.

According to Hillier and Lieberman (2010), the origin of OR can be traced back to military services where scientists were asked to do research on military operations. This was done because of the urgent need to allocate effectively limited resources to different operations and different activities within these operations. The success of this research sparked interest in the subject which led to the growth of OR, and progress was made in improving the techniques that existed at the time.

Over time, OR methods were adapted in different fields and today it is used in healthcare, banking and retail. This has led to it becoming an inter-disciplinary discipline with tools taken from fields such as statistics, economics, engineering, mathematics and psychology to create a new set of knowledge in decision-making (Taha, 2017).

2.1.1 Linear programming

Winston (2004) formally defined a linear programming (LP) problem as “an optimization problem for which we attempt to maximize or minimize an objective function, and this is the linear function of the decision variables”. Sharma (1989) elaborated that decision variable values need to adhere to the constraints set for the problem. Because the problem is an LP, each constraint has to represent a linear relationship in the form of an equation or inequality. Both Winston (2004) and Sharma (1989) stated that each variable has a sign restriction. For a variable a_i, the sign restraint specifies that a_i must be greater than zero ($a_i \geq 0$), meaning it is non-negative or the sign of the variable can be unrestricted shown with an unrestricted sign (URS).

Todd (2002) explained that linear programming is the study of the optimisation of a linear function. This optimisation is over a viable set governed by constraints in the form of inequalities. Todd called the problem insignificant, since it is only necessary to examine a limited number of vertices and possibly edges.

Most authors, including those cited above, state that LP is used most frequently in problems where limited resources are allocated among activities in an optimal way. However, Sharma (1989) noted that the allocation of resources is not the only application for LP. There are numerous applications for LP and any problem that can be mathematically modelled as a LP is considered as such. LP models with two variables are the easiest to solve. The solutions can be obtained graphically using an iso-profit line to find an optimum (ideal solution), from which can be derived an infinite number of solutions (Hillier and Lieberman, 2010).

2.1.2 Integer programming

Integer programming (IP) is a growing area of optimisation that is applied currently in many human industries because of high quality software. IP was developed over numerous decades and is still fast evolving (Conforti et al., 2014).

Wolsey (1998) stated that IP is a solution-seeking methodology for optimisation problems where only integer variables and binary variables (called axillary variables representing inter alia yes or no decisions) are present. Such problems arise in all sectors, whether, for example, developing a planning of work schedule for a production line, train or aircraft timetables, scheduling a maintenance team, or planning national or regional daily or weekly production of electricity.

Eck (1979) added that IP models are almost identical to LP models. The models differ at best at the level where in IP models all the decision variables take on integer (whole number) values. Although the integer specification is an apparently light alteration from LP, it stretches the capacity to model and resolve key real-world problems.

According to Winston (2004), although IP models are similar to LP models, IP models are harder to solve. Therefore, an algorithm like the simplex algorithm would not be enough to give an optimal solution. Solution methods for IP and other complex mathematical models are discussed in section 2.1.3

Hillier and Lieberman. (2010) stated that the reason that IP models are more difficult to solve than LP models is that for LP models non-integer values are permissible for decision variables. In practice, however, decision variables make more sense if they are integer. The author gives examples such as assignment problems where workers are allocated to machines or vehicles assigned for certain tasks or activities.

2.1.3 Mixed integer linear programming (MILP)

Chinneck (2015) stated that a MILP problem results when it is possible that some of the variables in an IP model have real values. This means that they can be fractions while others are integers. This makes the model mixed. In practice, IP problems are often encompassed and referred to under the more general MILP problem definition.

Smith and Taskin (2007) wrote a tutorial guide for MILP and stated: “Mixed-integer programming is a subset of the broader field of mathematical programming.” These mathematical modelling formulations include variable sets, which denote permitted actions in the system being modelled. The functions of variables are called objective functions. The objective function records each feasible set of options into a single value that assesses the solution quality, which can be optimised through maximisation or minimisation. Richards and How (2005) added that MILP models provide a "general framework for capturing problems with both discrete decisions and continuous variables". These problems include assignment, scheduling and transportation problems.

2.1.4 Mixed integer non-linear programming

Fernandes et al. (2009) wrote that many optimisation problems include integer variables and continuous variables that can only be modelled using a non-linear objective function, and therefore becoming mixed integer non-linear programming (MINLP) problems. This has led to a large range of applications, especially in a few planning and design areas.

Belotti et al. (2012) agreed with the engineering theory and added that numerous optimal choice problems in logical, designing and open division applications include both discrete choices and non-linear framework flow and system changes that influence the standard of the ultimate plan or design. These problems in decision-making may lead to MINLP problems that merge the combinatorial challenges of optimisation over discrete sets of variables with the difficulty of dealing with non-linear objective functions.

Bussieck (2003) stated that MINLP has recently experienced tremendous growth and a flourish of research activity. In this research, a brief overview of past developments in MINLP are provided. The work also discusses some of the future work that can foster the development of MINLP in general and, particularly, robust solver technology for the practical solution of problems.

The outer approximation method was part of the early MINLP solution methods used to solve these types of problems by breaking them down them into an arrangement of MILP problems and non-linear programming problems. Other algorithms, however, use another approach. An example is the extended cutting plane (α-ECP) algorithm. This algorithm only solves a set of MILP problems, rather than solving a set of MILP and non-linear problems (Westerlund & Petterson, 1995).

2.1.5 Goal programming

Deb (1999) defined goal programming (GP) as a technique often used in the field of engineering and planning of activities, most importantly to obtain a compromised solution. This technique aims to define and simultaneously satisfy several design goals. Deb explains that to solve GP problems, classical methods are applied to reduce the multi-objective goal programming problem into one objective, which consists of minimising a weighted sum of deviations from goals.

Iyengar and Srivatsava (2000) compared LP models and linear GP models. They noted that LP models have a single objective and linear GP models have numerous objectives. They added that both types of models have many suppositions in common. For example, how proportional, divisible, additive and deterministic the models' coefficients are. The similarities between the two modelling techniques make LP model construction studies a reference point when addressing modelling.

Sen and Nandi (2012) also noted the commonalities between GP and LP, calling GP an extension of LP that is used to optimise multiple objectives or goals. They formulated a GP model for a rubber production plant. The model consists of six goals in order of importance: maximise rubber tree survival, minimise expenditure, minimise number of paid workers, maximise the number of unpaid workers, minimise the quantity of chemical fertilizer, and maximise the quantity of bio-fertiliser.

2.1.6 NP, NP-complete and NP-hard combinatorial problems

Brucker (1979) described non-deterministic polynomial (NP) problems as discrete optimisation problems with a high level of complexity. These are typically problems like the TSP, quadratic assignment problems, and the knapsack problem. For these problems, a polynomial time algorithm can be used to verify a solution, but to date no polynomial bounded algorithms have been found to solve these highly complex problems and therefore heuristics or approximation algorithms are used to solve them.

Kokash (2005) wrote that NP problems as a class consist of two subclasses of problems including NP complete and NP-hard problems. It is unclear whether polynomial algorithms are available to find a solution to the problems. The NP class of problems is those problems whose solutions may be found on non-deterministic Turing machines in polynomial time. The NP-hard subclass of problems is understood to be a complex class of problems where heuristic and metaheuristic techniques are used to find a solution for the problems. The NP-complete subclass comprises problems where none are known to have a polynomial time algorithm, but if one does exist for any of the problems then it can be converted to polynomial time algorithms to solve all other NP problems.

Garey and Johnson (1979) described NP-complete problems as a class of problems with no known deterministic algorithmic solution. The authors added: “any known algorithms for solving these problems have the property that as the problem size increases, the number of steps necessary to solve the problem increases exponentially.”

Tindell et al. (1992) developed a real-time task allocation model for a distributed real-time system. In their research, they determined that that there were P^T ways to allocate T tasks to P processors. This led them to classify the problem as NP as it could not be solved with simple solution techniques. An algorithm called the simulated annealing algorithm was then developed to find a solution to the task allocation problem. The algorithm is discussed in detail in section 2.2.3.

Juan et al. (2015) wrote that many real-world combinatorial optimisation problems in logistics, healthcare, transportation, telecommunications and production are either NP-complete or NP-hard. These are usually largely scaled problems and require solutions in a short amount of computation time. This computation is often achieved through heuristic and metaheuristics. The author described a methodology which is an extension of metaheuristics through simulation called to solve real-world NP problems.

Dai et al. (2017) added that good approximations obtained from heuristics for NP-hard problems often require a trial-and-error approach. The authors developed a framework based on embedding graphs and reinforcement through repetition. Their method used the greedy algorithm to determine a solution and the result is then obtained through the graph-embedding algorithm which captures the current state of the solution. This method was applied to obtain a solution to the TSP.

Esfahbod (2007) created the Euler diagram in Figure 8 to illustrate the classification of Polynomial-time (P), NP, NP-complete and NP-hard set of problems. From the diagram, it can be deduced that the more complex a problem becomes, the more it rises to NP-hardness levels. The author explained that the left side of the diagram is valid under the assumption that P≠NP, while the right side of the diagram is valid under the assumption that P=NP with the exception that the empty language and its complement are never NP-complete.

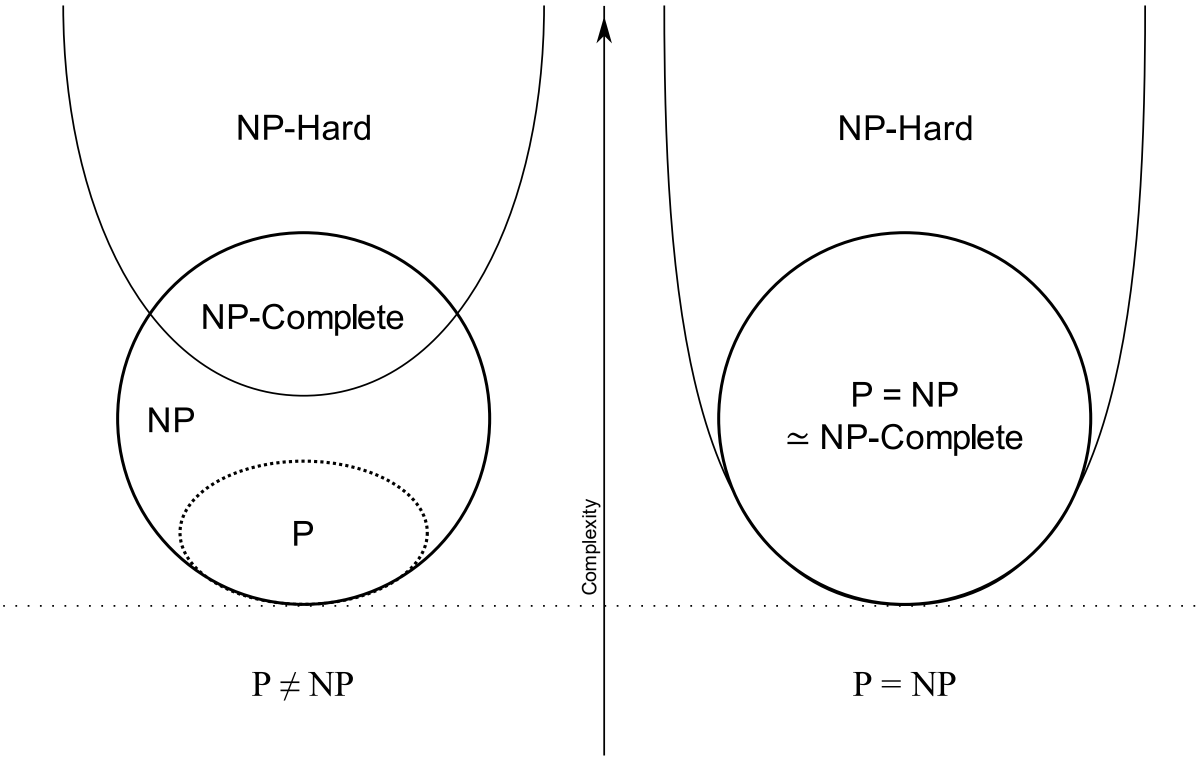


Figure 8: Euler diagram (Esfahbod, 2007).

2.2 Solution techniques for combinatorial problems

In this section, the literature on methods to find solutions to combinatorial problems is reviewed. The techniques are used for problems ranging from simple models which can be solved using techniques identified in the exact method section, to complex NP problems that can only be solved using techniques identified in heuristics and metaheuristics.

2.2.1 Exact methods

Rothlauf (2011) defined exact methods as optimisation strategies that ensure finding an ideal or optimal solution. Rothlauf added that these are usually the technique of choice if they can solve an optimisation problem with exertion that develops polynomially with the size of the problem.

a) Simplex algorithm

Dantzig and Thapa (1947) developed an algorithm called the simplex method to solve LP models. Hillier and Lieberman (2010) defined the simplex method as a general procedure for solving LP models. Martinich (1997) wrote that most real-life problems are however extremely complex and have too many variables to solve using the Simplex algorithm. The simplex algorithm can be programmed using computer tools such as Excel, QR and LINGO – to name a few

The simplex method examines the extremums in the viable solution space (known as the feasible region) in an orderly manner, going over a duplication of steps of the algorithm until the best solution is found. This is why it is also called the iterative method. The simplex method examines these corner points using matrix row operations, until an optimal solution is found (Žilinskas, 2006).

Reeb and Leavengood (1998) noted the simplex algorithm as being the most widely recognised method for resolving large LP problems, where "simplex" is a term in mathematics defined as follows: In one dimensional space, a simplex is a line fragment interfacing two focuses. In two-dimensional space, a simplex is in triangular form and is framed by joining the points. In three-dimensional space, the simplex is a pyramid with four sides and corners. The fundamental ideas are geometrical. However, the resulting algorithm to find the solution is an algebraic process.

The simplex method finds the best optimal point of the solution space to determine the solution to the LP problem. The optimal solution to LP problems must be a corner, despite there being many or other good solutions. The starting point for the simplex is usually the idle corner (where all variables are equal to zero). The algorithm then moves to the corner adjacent to the starting point, which best improves the solution. This is repeated iteratively, and each time the greatest feasible improvement is made. When there are no more points of improvement left, the best corner corresponds to the optimum for the LP problem (Winston, 2004).

Pan and Pan (2000) explained the simplex method as a Gaussian elimination technique where the coefficients of the linear system are collected in an augmented matrix for systems of linear equations. The authors first defined the basis as a "square non-singular sub-matrix from the coefficient matrix. The order of the basis is equal to the number of rows of the coefficient matrix". Some basic variables should have zero values. This is because the number of columns for the basis is set to a constant value n. This is a result of the right-hand side belonging to a subsection of the space for the range of the basis. Pan and Pan summarised the main idea behind the simplex method as being that the preceding matrix will be modified continually by pre-multiplying it by Gauss row operations until the matrix is optimal.

Swietanowski (1995) discussed new advances in the application of the simplex algorithm for LP. The author explained that the method "decouples" an idea of the basic solution to the simplex into two individual parts: a basis and a solution. This rule is what makes the possibility of incorporating new strategies into the algorithm possible. This occurs because the iterations do not necessarily have to be the corners of the simplex. Furthermore, it was stated that the advantage of approaching the simplex this way was the possible chance of stepping along paths that are not edges of the simplex.

b) Branch and bound

Taha (2017) wrote that Land and Doig formulated the first branch and bound (B&B) algorithm in 1960 for MILP and pure IP problems. Five years later, E. Balas formulated the additive algorithm to find solutions for Ips with variables that take on binary values. The additive algorithm was initially praised as being a great breakthrough in solving the general IP because its calculations were very simple (mainly adding and subtracting). Sadly, it was unsuccessful when it came to producing wanted computational advantages. Furthermore, the algorithm at first seemed to have no relation to the B&B method but was proven to be another version of the Land and Doig algorithm which had been developed five years prior to its development (Taha, 2017).

Winston (2004) stated that most IP problems are solvable through use of the B&B algorithm. The technique determines the optimal IP solution by creating a numbered list of the points in a sub-problem's feasible region. Mitten (1970) agreed, calling the B&B method a technique that is especially well known for its application to IP problems. Mitten added that the basic concept of the B&B method is to divide and conquer. The primary problem is broken down into small sub-problems that can be evaluated and eliminated as not containing potential optimal solutions easily until these sub-problems can be solved to find an optimum (i.e., conquered). The branching is done by parting the whole set of feasible solutions into small subsets. A bounding process to evaluate the feasibility level of the best solution within the subset follows the branching and then disregards the subsets if it is where the bound indicates the subset cannot have an optimal solution for the original problem.

He et al. (2014) agreed that the B&B method is a classic optimisation method for combinatorial problems, including MILP and MINLP problems, structured prediction and maximum a posteriori (MAP) inference. The authors added that while most research has been based on formulating “problem-specific” methods, there is not much knowledge on how to effectively design a strategy to search for nodes on a B&B tree.

i) Branch-and-cut

There are other versions of the B&B algorithm namely the branch-and-cut algorithm. Elf et al. (2001) noted that the branch-and-cut algorithms are part of the most effective techniques to solve MILP and other combinatorial problems for optimality. Mitchell (2002) agreed and added that branch-and-cut algorithms are very effective methods for finding solutions to a wide range of IP problems, and can also ensure optimality. The author described how the branch-and-cut method could be formulated for a distinct IP problem.

Tahernejad et al. (2020) developed an algorithmic framework to find solutions to mixed integer bi-level linear programming problems (MIBLPs) by through usage of a generalised branch-and-cut technique. The scheme the authors introduced blends characteristics from current algorithms (for regular MILP optimisation and MIBLPs) with advanced methods – therefore producing an adjustable and robust framework with an ability to solve an assortment of bi-level optimisation problems.

Bard et al. (1998) used the branch-and-cut technique to solve a MILP model for the VRP. The branch-and-cut method consisted of eight steps. The first step was to obtain an upper bound for the objective function, and then to solve the LP relaxation in step 2. Step 3 was an optimality check (if optimality steps 4–7 could be skipped). Step 4 was for finding an improved upper bound, and step 5 fixed the binary variables. Step 6 was to determine the solution to the separation problem and to generate valid inequalities, while step 7 creates a new node by using the branch-and-bound logic. The final step ensures that no sparse variables are fixed at zero.

c) Dynamic programming

Bellman and Kalaba (1957) described dynamic programming (DP) as the mathematical theory of multi-stage decision processes. Sklenar (2017), in contrast, called DP a general principle rather than an optimisation method. Sklenar explains that DP is an optimisation method focused on breaking complex combinatorial problems down to an arrangement of easier problems in a way where solving the easier problems needs less time than solving the original complex problem. Eddy (2004) agreed and added that DP algorithms are a great place to begin understanding what really goes on inside computational biology software. The core for many popular programs is a DP technique or a fast estimation of one, including sequence database investigation programs.

Giegerich et al. (2004) introduced an algebraic style to DP over sequencing data. They called DP a classical programming method, which can be applied to a variety of domains such as operations research, stochastic systems analysis, flow problems combinatorial discrete structures, and to dissect and resolve ambiguous languages.

2.2.2 Heuristics

Dale (2015) defined a heuristic as “a word from the Greek meaning ‘to discover’”. The author stated that it could be considered a technique to problem-solving that considers one’s personal experience. This means that every heuristic is unique to the model it is trying to solve. Gigerenzer and Gaissmaier. (2011) added to the definition of heuristics by writing that, “heuristics are efficient cognitive processes, conscious or unconscious, that ignore part of the information.” Because of the partial ignorance of certain information, using heuristics saves effort, but it can end up causing more errors than logical decision-making, which is conditioned using logic mathematical and/or statistical models. Hillier and Lieberman (2010) agreed with Gigerenzer and Gaissmaier (2011) and stated that heuristic methods are processes that are likely to produce “a very good feasible solution, but not necessarily an optimal solution, for the specific problem being considered”. This means that it is unclear how good (or bad) the obtained solution is. Hillier and Lieberman (2010) added that well-developed heuristic methods can usually lead to a nearly optimal solution (or end in no existing solutions for the problem). This was similar to the theory of Dale (2015) that heuristics are often used to develop strategies which scan a limited number of alternative options.

Heuristics are used to estimate and find solutions deemed as “good enough”. These solutions are usually for difficult NP-hard and NP-complete problems that alternatively cannot be determined by exact solution-finding techniques. Heuristics are direct search methods which use favourable approximation to determine better solutions. Heuristics have an upper hand in finding (good) results fast. The drawback is that the standard of the result (compared to the optimum) is usually not known until the optimal solution is found with certainty (Taha, 2017).

Choosing which heuristic method to apply depends on several factors. These factors include knowledge of whether the solution space is operational, tactical or strategic. Heuristic choices also depend on how analytically qualified the decision-maker is, the problem size, how frequent a calculation decision needs to be made, the available development time, and finally on whether significant stochastic elements are absent or present (Silver, 2004).

i. Variable neighbourhood search

Sevkli and Aydin (2006) named the variable neighbourhood search (VNS) as being among the latest heuristics used to find solutions for problems where a systemic neighbourhood alteration within a local search is done. Hansen et al. (2009) agreed and added that the VNS is based on systemic changes in neighbourhoods in order to find a local minimum.

Hansen et al. (2010) described the VNS as a framework based on neighbourhood changes for building heuristics to determine a local minimum. The aim of the VNS is to solve NP problems. The authors further stated that the VNS relies on local minimums from different neighbourhoods not necessarily being equal. The global minimum is then the smallest value from the set of local minimums. The authors stated that the VNS could be applied to the knapsack, MILP, scheduling, TSP and timetabling problems.

Abdullah et al. (2005) applied the VNS to a university timetabling problem. The objective of the model was to assign lectures to a time slot and room. The heuristic was based on a random-descent local search. The pseudocode for the algorithm is found in Figure 9, where n_k is a predefined set of neighbourhoods. The local search begins by generating a solution s' from the k^{th} neighbourhood until an optima s is obtained. The VNS was tested over 11 data sets and the results were favourable.

```
Initialisation:
(1)  Select the set of neighbourhood structure nk, k=1,…,K
     that will be used in the random descent local search;
     find the initial solution s; choose a termination
     criteria;
(2)  Record the best solution sbest ← s and f(sbest) ← f(s);
Repeat until the termination criteria is met:
(1)  Set k ← 1;
(2)  Until k = K, repeat:
(a)  Shaking: Generate a point s' at random from the kth
     neighbourhood of s (s'∈ nk(s));
(b)  Local search: Apply a random-descent local search with
     s' as an initial solution until local optima s" is
     obtained.
(c)  Move or not: Accept s" (s ← s") if it is better than
     incumbent solution s. Set sbest ← s  and continue the
     search with n1;
     otherwise,
        if f(s") is accepted by the acceptance criteria,
        then
           s ← s";
           Continue the search with n1;
        otherwise
           Set k ←k+1
```

Figure 9: Pseudocode for a VNS for a course timetabling problem (Abdullah et al., 2005).

2.2.3 Metaheuristics

A metaheuristic is defined as a general solution technique providing a basic composition, as well as strategy principles for formulating a heuristic algorithm to fit a specified problem type. Metaheuristics have evolved into a prominent technique in the list of optimisation tools for OR professionals (Hillier and Lieberman, 2010). Yang (2011) wrote that the "meta" in metaheuristics indicates a higher level and goes beyond the solution finding by traditional heuristics. This means metaheuristics are considered to be far better performers than simple heuristics.

Osman and Laporte (1996) wrote that metaheuristics comprises methods such as natural evolutionary computation, constraint programming, greedy algorithm, tabu search, neural networks, simulated annealing, non-monotonic searches, searching of space methods, and their combinations. Sörensen et al. (2018) agreed and added that a metaheuristic is an advanced algorithmic plan which is independent of problems. It provides a set of principles or policies to help develop heuristic algorithms in optimisations. The authors noted that metaheuristics include techniques such as tabu search, genetic algorithms, simulated annealing, and ant colony optimisation techniques; however, many more algorithms exist.

Metaheuristics are fundamentally designed to break free from being trapped at local optimal points by allowing lower moves where necessary. There is hope for the added flexibility of the search to generate better solutions. Stopping a metaheuristic can be based on a prescribed rule such as assigning a limit to the number of search iterations. This is different to the greedy algorithm which always stops immediately after reaching a local optimal point. Other termination rules can be when the neighbourhood of the current search point is empty and cannot generate a new viable search move when the current best solution is of good quality or when the number of times the algorithm has repeated itself in order to find the next best solution exceeds a number that has been pre-specified (Taha, 2017).

Sörensen et al. (2018) identified the tabu search, genetic algorithm and simulated annealing as the most commonly used and studied metaheuristics by OR practitioners. The characteristics of each are listed below.

a) Genetic algorithm

Paszkowicz (2013) stated that the genetic algorithm (GA) is a "heuristic search technique that mimics the process of natural evolution". Forrest (1996) agreed and stated that the GA is a biological evolution computational model. Forrest elaborated that Gas can be used as search techniques in problem-solving and can be used to model evolutionary systems. In Gas, a computer's memory stores what is known as binary strings, and over time these strings are adapted in much the same way that populations of individuals evolve under natural selection.

Taha (2017) described the GA as an algorithm that mimics the process of "survival of the fittest" in biological evolution. Each possible result for a problem is regarded as a chromosome inscribed by a gene set. Common gene codes consist of binary (0, 1) or integer sequences representing solutions of the problem. Carr (2014) agreed and added that Gas are used to find the feasible solution(s) to a given mathematical problem that maximise or minimise a specific function. Gas represent a growth calculation field of study where biological procedures such as reproduction and natural selection are imitated to search for the best/'fittest' solutions.

b) Simulated annealing

Henderson et al. (2003) wrote that simulated annealing (SA) is a common local search metaheuristic method which addresses discrete and continuous problems, though continuous problems are not addressed to the same extent as the discrete combinatorial problems. The most important characteristic of SA is how it enables the ability to be free from local optimal points. This is done by allowing uphill movements (these movements make the value of the objective function worse), while hoping to find a global optimal point.

Corana et al. (1987) agreed and further described SA as an essentially repetitive random search process with moves that are adaptive along the directions of the coordinates. The authors added that it permits hill-climbing moves controlled by a stochastic basis, and thus tends to avoid stopping at the first local minimum points determined by the algorithm as shown in Figure 10.

Xinchao (2011) defined SA as a generic metaheuristic algorithm with probabilistic features for the global optimisation problem. Xinchao added that Kirkpatrick et al. (1983) and Cerny (1985) presented this algorithm independently. It is further described as a method to determine a good enough solution to combinatorial problems using an arbitrary variant of the current result. A bad version is taken on as the new solution at a decreasing probability as the calculation progresses. The algorithm has a higher probability of finding an optimal or almost optimal solution when the rate of decrease or “cooling time” is slow.

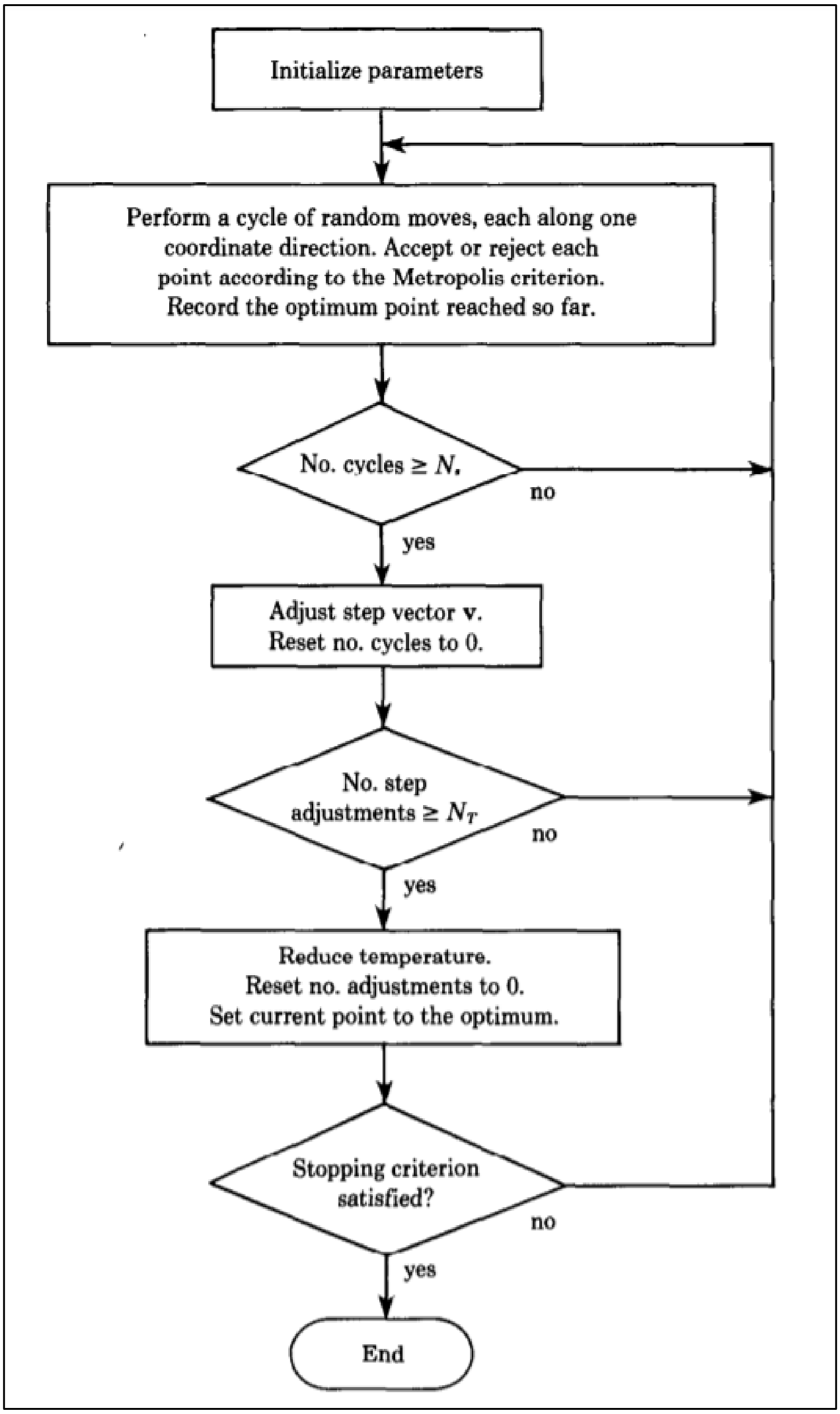


Figure 10: A simulated annealing algorithm flow chart (Corana et al., 1987).

c) Tabu search

The tabu search (TS) is a method used to solve NP-hard and NP-complete problems. The application of these problems ranges from matroid settings and graph theory to pure general and MILP problems. It is procedure that can adapt and use numerous other methods like exact methods and specialised heuristic techniques to direct the algorithm to overcome constraints of local optimality (Glover and Laguna 1998).

Gendreau (2002) presents the basic idea behind the TS in the form of a tutorial. The research places special emphasis on presenting the connection with classic local search algorithms and on fundamental parts of any TS heuristic, such as defining the space for the search, the neighbourhood arrangement, and memory of the search. Application of TS is popular for finding solutions to NP-complete problems such as the facility location problem (FLP) and the VRP.

Pirim et al. (2008) qualified this by writing that problems such as the ones mentioned by Gendreau (2002) are encountered in fields like vehicle routing and scheduling. These are mostly NP-complete problems that are very complex. TS is one of the most effective heuristic methods as it determines solutions of good quality in a comparatively short time.

Hillier and Lieberman (2010) described the process as starting with a feasible initial solution. Iterations are then done with a suitable local search process to specify the possible moves into the local neighbourhood of the latest provisional solution. The stopping criterion, is set as a specified amount of computation time, a specified number of times the algorithm repeats calculations a specific number of consecutive iterations without improving the best value of the objective function or any iteration where no possible solutions are found.

2.3 Optimising rail line capacity

From the literature, three parameters are classified as the most important factors that influence the railway line capacity. These are train types, schedules and railway infrastructure. Abril et al. (2008) state that capacity depends on these factors and the authors reviewed methods to analyse capacity while researching the effects these might have on different types of capacity i.e., theoretical and actual capacity).

Landex (2008) supported the classification of the three parameters by stating that “railway capacity depends on the infrastructure, the rolling stock (train types) and the timetable shown”. This is illustrated in Figure 11. It was stated that for the capacity challenge to be managed more effectively, we must examine these factors in detail in order to define, analyse and improve capacity utilisation. In this research, only the parameters without the inclusion of human factors and external factors were analysed. The three parameters are now described in more detail.

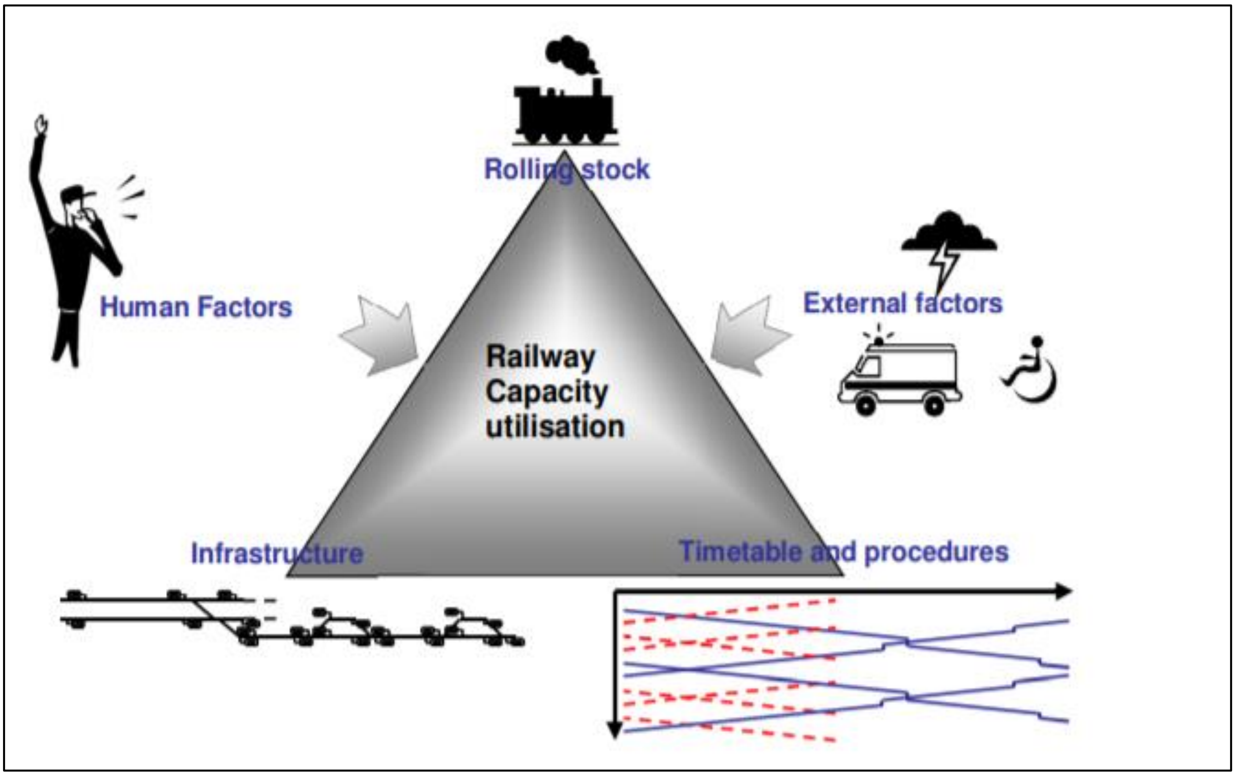


Figure 11: An image of the parameters affecting capacity (Landex, 2008).

2.3.1 Train types

A mixture of different train types can have considerably different operating elements. Dingler et al. (2009) studied the North American railroads and noticed the rapid growth in freight demand, which led to a need to increase capacity to accommodate this demand. They stated: “efficient planning of new capacity requires understanding how the mixture of traffic interacts to affect capacity”. The heterogeneity in the mixed characteristics of the varied types of trains creates longer delays than those occurring as a result of homogeneous traffic. This means that having a diverse mix of elements like priority and average speed creates longer delays than when traffic is not mixed. The purpose of Dingler et al.’s research was to “attempt to provide a better understandIng of the impacts of various aspects of train type heterogeneity to enable more effective planning and efficient rail operations”. Efficient rail operations alleviate unnecessary delays on rail lines, which decreases available capacity. The authors conducted their research using freight and passenger train simulation software. Their findings demonstrated that individual delays resulting from heterogeneity of train types can be mitigated by operating strategies. Their research also highlighted that priority seemed to have the biggest effect on delays.

Yaghini et al. (2014) partially agreed with this approach and developed an IP model to analyse the impact of heterogeneous traffic on railway line capacity for the Tehran–Zanjan route in Iran. They suggested that rail line capacity depends on traffic on the line, infrastructure to handle this traffic, and operations. They argued that an increase in combining different trains generates more interference and this reduces the railway line capacity and therefore identifies train types as having a key impact on line capacity. They formulated an IP model for a multi-commodity network. The inputs of the IP model are the train type attributes and infrastructure, and the outputs are the line capacity for each railway line section. The objective function for the model seeks to maximise the number of train paths over a rail line by considering priorities. The authors demonstrated an increase in capacity without the use of a detailed timetable. The model also showed that an increase in the number of low-speed trains decreases line capacity. This was because trains moving at low speeds take up capacity on the line for longer periods of time.

Burdett et al. (2006) developed methodologies to analyse and improve capacity by studying how different train types, together with their speeds and lengths, can be used to estimate the absolute carrying capacity of railway system facilities. The authors introduced the term "absolute capacity" which they defined as the number of trains that could pass through the critical section of the line. This capacity value was then optimised and used in multiple planning purposes. A mixture of different train types was used as an input to the MILP model, and it gave an output value for absolute capacity as well as different arrangements of the different trains – referred to as the "actual" train mix. The authors suggested that the model can also be used to determine actual capacity for lines and networks.

Concerning train types, Landex (2008) wrote that "It is not only double track railway lines with homogeneous traffic that achieve higher consecutive delays when the capacity consumption increases. However, double track with homogeneous traffic is the simplest situation of calculating delays and delay propagations and illustrates how one initial delay can affect the succeeding trains." The author analysed trains that operated at the same speed, trains that move at different speeds, and trains which run on a single track. The research proved that there was a higher risk of consecutive train delays on heterogeneous trains operating on the lines, which would then cause a significant loss in capacity.

2.3.2 Schedules

Dick and Mussanov (2016), like Dingler et al. (2009), studied North American railways and how to use flexible operations with varying departure times to determine optimal infrastructure. Most developed models used to minimise cost by matching infrastructure investments to projected traffic levels using rigid timetables. It was argued that a flexible schedule increases the number of possible meet locations (where trains going in opposite directions meet) especially on single line railways, and this increases delays. Their research used simulation techniques to model operations onto single-track routes to investigate the relationships between infrastructure and flexible timetables. A minimum delay schedule and flexible model was developed. The research suggested that adding even small amounts of flexibility to train schedules resulted in a significant increase in delays on the line, which may lead to unwanted effects on rail yards and terminals.

Luethi et al. (2007) argued that rescheduling combined with train control is a low cost and effective approach to increasing capacity in cases of heavily used heterogeneous traffic railroad networks and yards. The method was designed to increase capacity without changing the infrastructure in any significant way. They mapped the rescheduling process depicted in Figure 12. The approach comprises a step-by-step method of rescheduling operations on the rail line. This approach combines rescheduling with network traffic control to minimise schedule reserve times without reducing stability and thereby increasing network capacity.

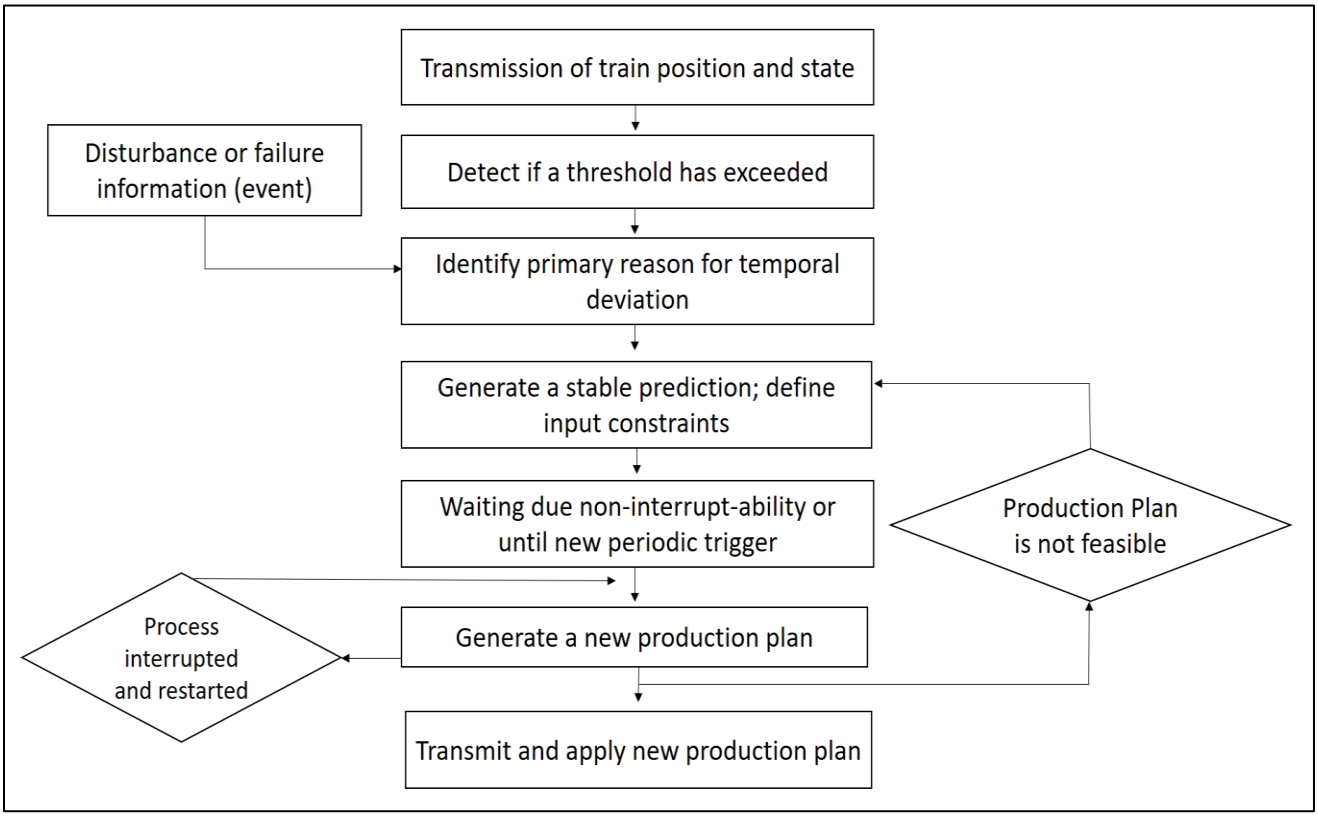


Figure 12: Railway network scheduling process (Luethi et al., 2007).

Zhang et al. (2018) developed an MINLP model for the scheduling of trains. They emphasised that scheduling plays an important part in the improvement of rail capacity when trying to remove or manage traffic pressure. The MINLP model was used to optimise train services for given headways determined by demand. The MINLP model was then converted to a MILP model and solved. The result was simulated and showed evidence of a reduction in train use and therefore a reduction in rail line congestion.

Hansen (2004) investigated how capacity on rail lines could be increased on Dutch railways by optimising timetables and schedules. The author listed minimal headway time, slack, and waiting times as the timetable variables. The suggested approach was used to estimate the blocking times and buffer times at bottlenecks. The model used simulation models and queueing theory to estimate waiting times. The study results could be used to improve the quality of operations by blocking times as a function of the operations punctuality levels.

Andersson et al. (2015) used an optimisation approach to reduce timetable delays. They proposed a MILP model that reallocates margin time to increase the robustness of an existing timetable. This increases the robustness at critical timetable points that are sensitive to any type of delay on the rail lines. Increased robustness ensures that a delay experienced by one train will not affect the rest of the trains running on the line. Increased robustness and decreased delays resulted in more effective use of the railway capacity.

Higgins et al. (1997) studied the use of heuristics for single train line scheduling. They formulated a MILP model for the optimisation of train schedules for single track lines. This is a difficult problem to solve and so they used a combination of heuristics and metaheuristics. This included the Greedy algorithm, TS and GA, which was used to create a hybrid algorithm to solve the MILP model. They produced a better train timetable using the genetic algorithm in their developed hybrid heuristic.

Furini and Kidd (2013) presented a heuristic which solves the scheduling problem on a railway network, and which resulted in no conflicts. Results were determined using real data from a Milan case study. The problem allowed trains to be re-routed when no possible paths are available in the space–time graph. The method searched for alternative routing possibilities and takes these as input. Furthermore, other than fixing conflicts in given timetables the algorithm could be used as a tool for planning. The planning can aid in testing capacities for networks or for resolving conflict in real-time in the case of railway network disruptions.

2.3.3 Infrastructure

Khandem Sameni (2012) listed four infrastructure parameters that affect how trains flow on the corridors and rail networks. These are the number of tracks on the line, the average speed the infrastructure can handle, the axle load, the loading gauge, and the maintenance of the infrastructure. These parameters affect the capacity of the lines. The author explained that the number of tracks determines the theoretical capacity on the line. Trains running in opposite directions on a single-track line increase waiting times because of the conflict they create. The speed of trains affects the capacity in such a way that the faster the trains, the less time is spent occupying the infrastructure. The maximum axle load of lines helps determine how heavy freight can be and how much freight can be carried on the tracks. Loading gauge determines the maximum height and width of trains on the line. This is because some lines have bridges and tunnels that trains need to travel through. Lastly, consistent maintenance is needed to minimise derailments and to monitor wear and tear of the infrastructure. This interferes with the railway operations and affects capacity utilisation (Khandem Sameni, 2012).

Singh et al. (2012), like Yaghini, used MILP models to determine the capacity needs of the Hunter Valley coal chain in Australia. They also note most affordable capacity expansion plans to meet the increasing demand, while minimising infrastructure and demurrage costs. Constraints were used to limit completion times for stacking, stockpiles, receival capacities, load-point capacities, and train jobs. The objective of the model is to minimise the cost of the sum of delays and the cost of using additional paths and stations on different sections of the track and adding loading stations at the terminal. The authors applied the GA, squeaky wheel heuristic, and large neighbourhood search to solve the models.

Garrisi and Pastor (2019) studied the problem of rail infrastructure congestion in the European Union (EU). Most rail companies in the EU, like others around the world, were considering investing in new infrastructure. Specifically, the company was considering adding a new diversion track, but the cost was high and they needed to investigate alternative solutions. The case study showed that the use of a diversion track in the case of insufficient planned rail line capacity may not be advantageous for trains – even with a longer waiting time for capacity to be released. They suggested that one of the ways to eliminate the negative effects of congested railway infrastructure is to invest in the increase of its capacity through infrastructure expansion. This increase was achieved by improving the rail operations, which freed up some capacity. However, the models created also show that as the performance of the companies grows, the demand will increase and eventually new infrastructure will be needed.

Burdett (2016) considered the expansion of railway infrastructure to increase capacity on an Australian rail line. The suggested expansions may include the building and positioning of new sidings and signals on the line (these usually result in delays as trains come to a complete stop at both), the replicating of existing sections of rail through track duplication, and the advancement or replacement of existing corridors to various gauge tracks. The cost of these expansions is high and therefore the author applied IP models to identify the bottlenecks and critical points where the expansions will be implemented and will have a greater positive impact on the capacity.

2.3.4 Simultaneous parameter modelling

The current study investigated the improvement of railway capacity. It is focused on analysing effects related to capacity utilisation in structured railway operation. Most MILP models used in the above literature to optimise or analyse rail line capacity only focus on optimising or investigating the parameters mentioned in section 2.2 individually, in order to determine how each will affect the capacity. However, the parameters are collectively important in developing an accurate model for optimising capacity. It is therefore important to investigate an MILP model where all the parameters are analysed.

Landex (2008) supported the statement above and provided a three-dimensional representation of capacity utilisation – as shown in Figure 13. The author noted the four parameters affecting the capacity utilisation as stated by the International Council for Railways. The number of trains represent the schedules, average speed of the trains is the speed that the infrastructure can handle, heterogeneity pertains to the types of trains, and the added parameter of stability refers to the robustness of schedules. The author noted that the parameters in the diagram affect each other. For example, decreasing heterogeneity may increase speed or create more stable timetables. This leads to increased capacity, and a more stable timetable can accommodate train heterogeneity but that may lead to a decrease in average speed and therefore an increase in capacity utilisation and a decrease in the total number of trains running on the line.

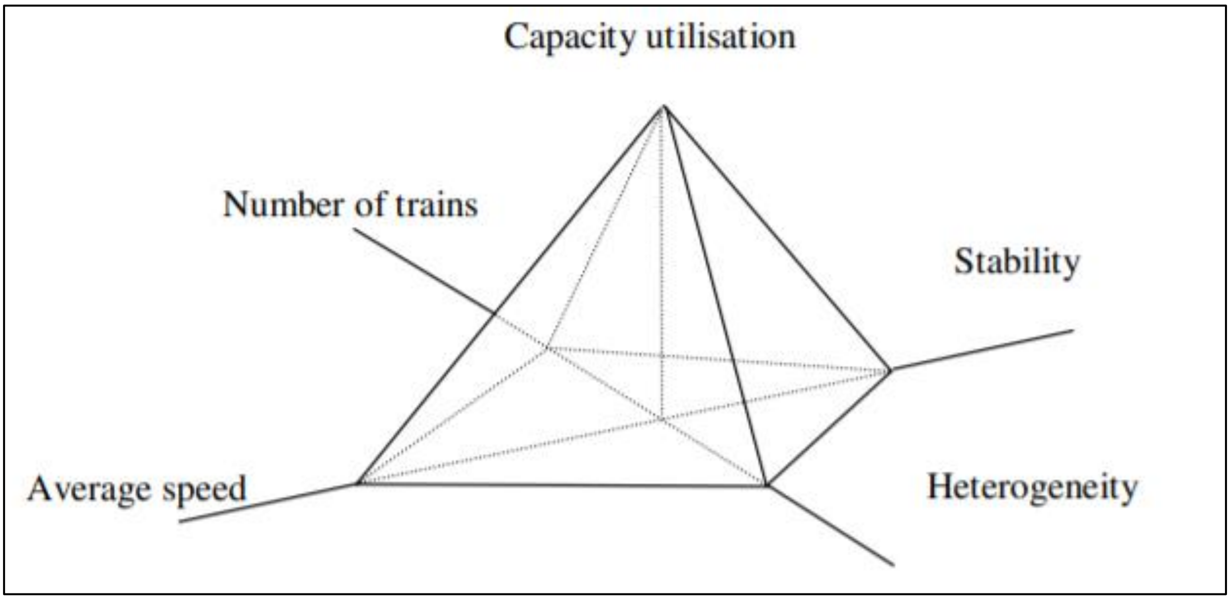


Figure 13: A three-dimensional representation of capacity utilisation (Landex, 2008).

Lindfeldt (2011) looked at two parameters simultaneously and developed the Time-Varying Effect Model (TVEM), which models the train schedule and the line infrastructure as variables to evaluate the capacity for different train types. This was supported by further research by

Lindfeldt (2015), where the author added that by changing conditions and making new timetables, general capacity relationships can be established.

2.4 Train timetabling problem models for train schedules

The above literature review of capacity modelling and optimisation of railway capacity clearly indicates that schedules and timetables are important. Schedules speak to how the infrastructure is used and contain information on the types of trains that run on the line. Thus, it was important for this research, while studying the capacity of the NatCor line, to analyse the schedules in detail. To perform a thorough analysis of the schedules, this study considered a train timetabling problem (TTP). This is a variation of the assignment problem where all variables are modelled in the NatCor line to minimise delays on the line. The resulting MILP model was then used to compute the maximum number of trains that can run from the various origin points to the intended destinations at optimal delay.

Another model that was considered for this research was the vehicle routing problem with time windows (VRPTW). This problem is discussed in section 2.5. This problem, together with the TTP, forms part of the TSP class of problems. Thus, different variations of the VRP were studied and the final problems were defined and formulated to solve the MILP models.

Section 2.4 begins with a detailed description of the TTP and VRP.

2.4.1 Mathematical formulation of the TTP

Caprara et al. (2002) analysed the TTP concentrating on single, one-way track problems for a company in Italy – which links two key stations with numerous stations in-between. All trains connect two stations down the lines. The trains may have to stop midway at some of the stations along the line. The authors proposed a formulation for the problem by using graph theory which includes a directed pseudograph in which nodes represent the departures/arrivals at a given station at a specified time interval of instance. They formulated an ILP focusing on one Italian corridor. Their model was adapted from a TTP developed by Brännlund et al. (1998). The objective function for the IP model aims to maximise the sum of the profit from each of the arcs traversed by the trains (Caprara et al., 2002).

Robenek et al. (2014) proposed an alternative TTP, which they referred to as an ideal train timetabling problem (ITTP). The ITTP was used to analyse and improve the planning process using current passenger railway service data. They observed that the current practice did not define ideal and profitable timetables and formulated the ITTP using an analysis of passenger behaviour. The resulting MILP model has an objective function minimising the cost of the journey and additional weighted values based on the demand and value of time. The model output consisted of the passengers' routes and thus the efficiency when planning the rolling stock assignment (i.e., the rolling stock planning problem – rolling stock being the vehicles, wagons and carriages running on the railway) was measured using the train occupation. This output was then used as an input for the traditional TTP.

According to Stanojevic et al. (2010), the traditional TTP seeks to determine a cyclic timetable for a train set; it does not disobey track capacity constraints and satisfies a number of operational limitations. They developed MILP models using the real-time data of Serbian railways as a case study. The authors suggested that the three most important properties of the rail network are the cars (train carriages), locomotives (engine vehicle for trains) and crew, and the flow of these determines the quality of the timetable. An IP approach was used to determine an optimal train timetable for a single, one-way track which links two key stations with numerous stations in-between. The MILP model is shown below with the decision variables $x \in Arcs$ representing the flow of a train on an arc (arcs help to trace the different train paths *Path(t)* traversed by each train *t*, through different stations, *s*). This arc set includes arcs $x_{t,s,k}^{wait}$ and $x_{t,s,k}^{out}$. These arcs represent the trains waiting at and exiting stations *s* at time *k.* Other arcs such as $x_{t,\,a^t,k}^{arr}$ and $x_{t,\,d^t,k}^{arr}$ denote the train *t* arriving and departing at time *k*, respectively. Variables ld_t and ed_t represent the latest and earliest departure times and ea_t and la_t are the earliest and latest arrival times, respectively. Q is a set of discrete time intervals *k* and sets *S* and *T* are the sets of stations and trains, respectively. The variables md_t, mi_s and mw_s represent the maximum allowed delay for train *t*, the maximum number of train stations *s* at the same time, and the maximum number of trains waiting at station *s.*

$$minimize\ z = \sum_{t \in T} \left(\sum_{k=ed_t}^{ld_t} (k - ed_t)\, x_{t,d^t,k}^{dep} + \sum_{s=Path(t)} \sum_{k=ea_{t,s}}^{ea_t + md_t} x_{t,s,k}^{wait} \right) \qquad (1)$$

Subject to:

$$\sum_{k=ed_t}^{ld_t} x_{t,d^t,k}^{dep} = 1, \forall t \in T \quad (2)$$

$$\sum_{k=sa_t}^{la_t} x_{t,\,a^t,k}^{arr} = 1, \forall t \in T \quad (3)$$

$$\sum_{x \in Enter(t,s,k)} x - \sum_{x \in Exit(t,s,k)} x = 0, \forall t \in T, \ \forall s \in Path(t), \forall k \in Q, ea(t,s) \leq k \leq ea(t,s) + md(t) \quad (4)$$

$$\sum_{t \in T} x_{t,s,k}^{wait} \leq \ mw_s \ , \forall s \in S, \forall k \in Q \quad (5)$$

$$\sum_{t \in T} (x_{t,s,k}^{wait} + \ x_{t,s,k}^{out}) \ \leq \ mi_s \ , \forall s \in S, \forall k \in Q \quad (6)$$

$$\sum_{t \in T^{out}} \sum_{l \leq k-1, l+f(t,s) \geq k} \ x_{t,s,l}^{out} + \ \sum_{t \in T^{in}} \sum_{l \leq k-1, l+f(t,s+1) \geq k} x_{t,s+1,l}^{out} \leq 1 \ , \forall s \in S, \forall k \in Q \quad (7)$$

$$x \in \{0,1\}, \forall x \in Arcs \quad (8)$$

The objective function minimises the sum of total delays with arc costs representing the delays as discrete time units. Constraint sets (2) and (3) ensure that each train can leave only one departure point and one arrival point at a given time instance. This helps track the path the trains traverse. Constraint set (4) notes the constraints that preserve the flow. Constraint set (5) ensures that the waiting train number at a station does not exceed the total number of tracks used at the station. Constraint set (6) ensures that the maximum number trains which dwell at the station and those that pass through the station at that same time are less than the total number of tracks. Constraint set (7) ensures that capacity limits between sections are upheld. The results when solving the MILP model were an output of optimal departure times for all trains individually and showed that 80% of the trains on the Serbian railways should leave earlier than currently scheduled.

Liu and Han (2016) formulated an ILP for the TTP using the Beijing–Shanghai high-speed railway line as a real-life problem. They studied the capacity utilisation for the section between Xuzhou East Station and Nanjing South Station. The section has two tracks and runs high-speed trains. There is heavy rail activity, traffic movement and trains running at different speeds that operate on the line. The TTP seeks to determine a possible timetable with a high efficacy where different lengths of headway times are differentiated in the train timetabling process. The TTP model in equations sets (9) – (12) by Lui and Han (2016), similar to the model suggested by Stanojevic et al. (2010), seeks to minimise the deviation to the earliest departure time, thereby minimising delays. In the model, *C* represents a conflict set of pairs of train paths and each pair of train paths violates headway time constraints. L is a set of trains and P_l is a set of train paths for a train *l* where $\forall l \in L$ and P is a set of paths for all trains

$$minimise\ z = \sum_{l \in L} \sum_{p \in P_l} w_p x_p \qquad (9)$$

Subject to $\quad \sum_{p \in P_l} x_p \leq 1 \quad \forall l \in L \qquad (10)$

$$\sum_{p \in C} x_p \leq 1 \quad \forall c \in C \qquad (11)$$

$$x_p \in \{0,1\} \qquad (12)$$

The objective function in equation (9) for the model minimises the total weight w_p for each train traversing on a certain path *p*. This total weight represents the sum of the weight assigned to the trains for dwell and deviation times. x_p is a binary decision variable where the value is one if the train *t* traverses the path *p*, and zero otherwise. Constraint set (10) ensures that at most only one train path is generated per train. Constraint set (11) removes all conflicts linked to the violation of headway time constraints for any two train paths belonging to different trains. If the time interval between any two consecutive trains is not adhered to, the train paths are added to set C, which is a set of conflicts occurring on the line. Constraint set (12) assigns the decision variables x_p to a binary value.

Garrisi et al. (2019) investigated looking beyond the TTP and developed a model that can be used for scheduling problems in different industries. They formulated a MILP model to optimise the scheduling of trains on a railway network and developed a heuristic algorithm to solve the scheduling model. The model should, ideally, produce schedules that have zero conflicts. They provided a disclaimer that although the MILP model is solvable for simple cases, complex cases may be impractical to model and solve – especially in the case of creating new optimal timetables. The final model is a multi-objective problem that is modelled by a cost function. The total train delays are minimised through the objective function for this problem. Delays were measured as absolute deviances between real/actual times and scheduled times of departure and arrival times of the route sections.

Cacchiani and Toth (2011) studied literature that dealt with TTPs in their nominal and robust state. The nominal version of the TTP aims to optimise the objective function, which can have different forms depending on the company requirements for the model. Most companies would want to maximise customer satisfaction and so the TTP would produce schedules according to that objective. The nominal TTPs are characterised by features such as cyclic and congested non-cyclic schedules, one-way corridors with single tracks or a railway network, transportation of freight or passengers (where passenger trains are often scheduled first), and objective functions like minimising costs for the rail company or minimising total travel times for passengers. The objective function for the nominal TTP modelled in the study aims to maximise the total value or profit for the feasible timetables. This model was adapted from the model by Caprara et al. (2002). Cacchiani and Toth (2011) added that recent studies have been dedicated to solving robust TTPs, where the objective is to minimise delays on the rail lines as much as possible. Approaches used to solve these robust TTPs include stochastic optimisation, metaheuristics, Lagrangian methods, light and recoverable robustness, managing delays, two criteria method. The robust timetabling technique, which aims to minimise train delays, is an adaptation of the TTP by Liebchen et al. (2009) using weighting of delays similar to the model in equations (9)-(12) from Liu and Han (2016).

2.4.2 TTP solution methods

Liu and Han (2016) solved the TTP using a branch-and-price algorithm that they developed themselves. The algorithm is outlined in the flow chart in Figure 14. The technique mixes the B&B method with column generation. In the branch-and-price tree for their model, every node *n* corresponds to an LP, P_n. The flow chart starts at point n_r which is the root, and the LP P_{n_r} is the relaxed LP for the TTP. Generating columns solve the LP P_n. The set of active nodes A originally had n_r and nodes, with no decision values. The node n_a is then branched and determined by a search algorithm. If the minimum best solution exceeds the value of one, the node *n* with the highest value *One*(x_n), which is the number of decision variables which satisfy $x_p^* = 1$ and $x_p^* \in X_n$, is then chosen as n_a. It is advantageous to find a possible solution of the TTP. The output of the model showed favourable results when different headway times were considered.

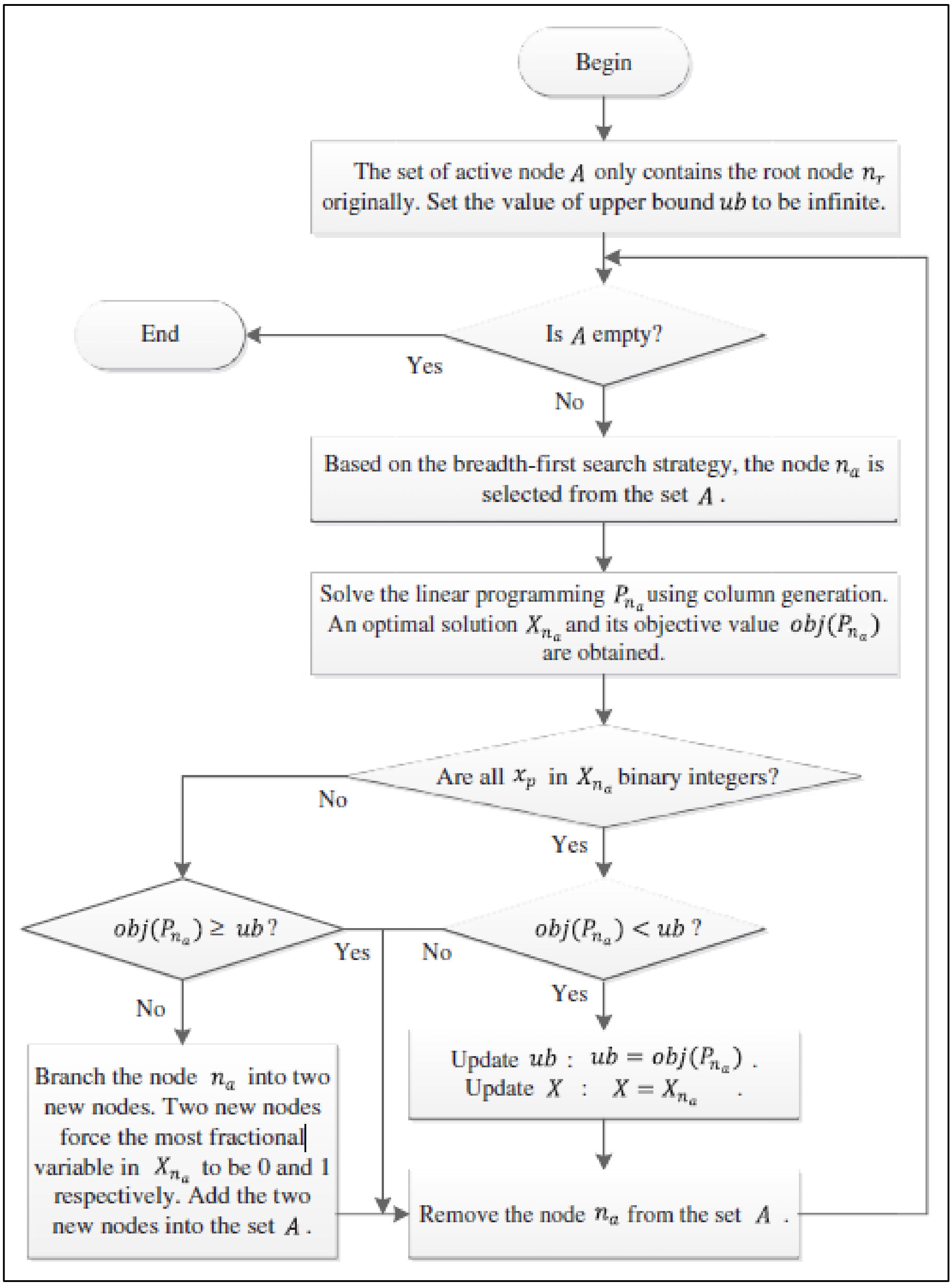


Figure 14: Flow chart for the branch-and-price algorithm (Liu and Han, 2016).

Garrisi et al. (2019) developed a GA metaheuristic to solve a MILP problem with multiple objectives. The GA is made up of two parts. The first part of the algorithm chooses a path using the earliest-exit rule. Part 1 of the GA is executed concurrently based on the following objectives for service: each intent of service is handled as an individual job and true parallel utilisation of the method avoids dividing the work between the physical units unequally. Every constraint set except allotment of resources and links between services is then implemented through assembly. The allocation of resources then follows. The allocation is completed using a different method. The second part of the algorithm allocates a resource (not necessarily trains) to a time interval. This is done sequentially to prevent conflicts on a resource. A delay is introduced when the allocation of resources cannot be completed for an interval.

In addition to solution approaches for TTPs, Lee and Chen (2009) developed a heuristic which includes both train timetabling and train pathing, and can solve real-life instances so making it scalable. The reason for this was the problems to be addressed with automation of train pathing and timetabling were continually growing and the processes were gradually moving away from being done manually. The heuristic lets the trains' operation time depend on the allocated track and let the lowest headway time between consecutive trains depend on the current standing of the train. The solution heuristic formulated by Lee and Chen (2009) is mapped out in Figure 15.

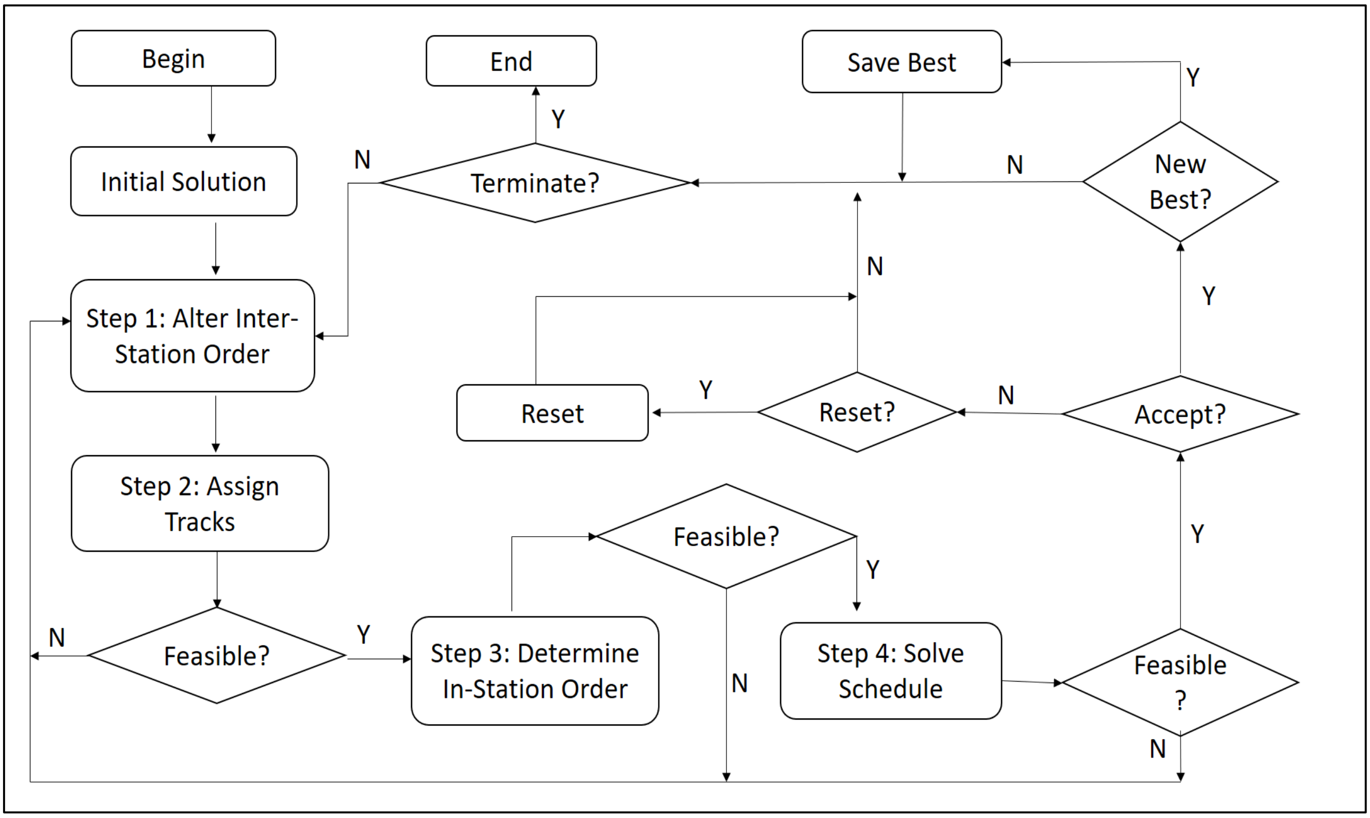


Figure 15: A flow chart of the TTP heuristic (Lee & Chen, 2009).

The algorithm was used to develop a primary solution with an uncomplicated act, and derives the solution repetitively through a process which consists of four steps. The heuristic begins by generating a feasible solution. This is done by running the services in sequence by order of departure times. In step one, the algorithm orders the trains blocks in-between stations. In this step, the algorithm seeks to find new results by changing the latest possible result by altering the train sequence at inter station blocks. In step two, trains are assigned to tracks at the stations with a binary IP model, and this is done for stations with more than two tracks. Trains are then ordered on intra-station tracks in step 3. This is done using a LP model to obtain train ordering that can be used when going through the different stations. In step four, the algorithm solves for the schedule using a LP model, where appropriate headways are applied to all blocks on the line. The threshold rule is then accepted if the newly calculated solution for the model is superior to the initial and current result.

2.5 Vehicle routing problem models for train schedules

Christofides (1976) stated that vehicle routing problems (VRPs) are a supplement for the TSP. The author defined the VRP as an order of problems, which involves a number of vehicles visiting customers to supply or deliver goods, usually from a common depot. The key goal or aim of VRP models is to keep the total costs incurred for the delivery of the goods at a minimum value.

In their book, *The vehicle routing problem*, Toth and Vigo (2002) agreed with this definition and listed various types of VRPs, naming the three most popular as the VRP for TSP, the capacitated VRP and the VRP with time windows (VRPTW).

Desrochers et al. (1991) states that the VRPTW is a generalised version of the VRP where the service to the customers is performed between the earliest and the latest times in which the customer permits the beginning and end of service. The authors formulated a new method to find a solution to the VRPTW optimally by generating columns. Possible columns are then included, as necessary. This is done by finding a solution to a shortest path problem with periods of time and restrictions on capacity through DP. The objective function was to minimise the total cost incurred. The method found results for 100 customers, which became a benchmark solution suggested by the authors for future research.

Kallehauge (2006) noted that the VRPTW is a VRP, which is used to optimise the routing of numerous vehicles between a starting point (the depot) and numerous customers who have to be visited within specified time spans. The thesis was based on the evolution of new solution techniques to solve the VRPTW. In the research, he stated that the VRPTW inherited most of its aspects from one of the longest running combinatorial problems, the TSP. Ombuki et al. (2004) identified the VRPTW as an extension of the capacitated VRP. The authors formulated the VRPTW as a problem with multiple objectives and developed a GA to find a solution. The objective functions minimised the total distance travelled by the vehicles and minimised the total number of vehicles used to service customers. The final objective was to minimise the sum of total costs incurred to transport goods to customers in a specified time window.

2.5.1 Mathematical formulation of the VRPTW

Akgül (2020) developed a VRPTW model with a homogeneous fleet, similar to that of Kallehauge (2006). The parameters and variables used in this model are found in Table 1:

Table 1: A list of variables for the VRPTW (Akgul , 2020).

Variables and parameters	**Description**
$(\boldsymbol{a_i}, \boldsymbol{b_i})$	Customer service time intervals
$\boldsymbol{c_{ij}}$	Cost between *i* and *j*
$\boldsymbol{d_i}$	Customer demand
$\boldsymbol{s_i}$	Service time of customer i
$\boldsymbol{t_{ij}}$	Travel times between *i* and *j*
c	Number of customers
n	Nnumber of nodes, 0 …. n+1

Q	vehicle capacity
v	Number of vehicles

The formulated model is given as:

$$minimise\ z = \sum_{k \in V} \sum_{i \in N} \sum_{j \in N} c_{ij} x_{ijk} \quad (13)$$

$$\sum_{k \in V} \sum_{j \in N} x_{ijk} = 1 \quad \forall i \in C \quad (14)$$

$$\sum_{i \in C} d_i \sum_{j \in N} x_{ijk} \leq Q \quad \forall k \in V \quad (15)$$

$$\sum_{j \in N} x_{0jk} = 1 \quad \forall k \in V \quad (16)$$

$$\sum_{i \in N} x_{ihk} = \sum_{j \in N} x_{hjk} \quad \forall k \in V, \forall h \in C \quad (17)$$

$$\sum_{i \in N} x_{i,n+1,k} = 1 \quad \forall k \in V \quad (18)$$

$$g_{ik} - g_{jk} \leq M\left(1 - x_{ijk}\right) - \left(t_{ij} + s_i\right)x_{ijk} \quad \forall i,j \in N, \forall k \in V \quad (19)$$

$$a_i \leq g_{ik} \leq b_i \quad \forall i \in N, \forall k \in N \quad (20)$$

$$x_{ijk} \in \{0,1\} \quad \forall i,j \in N, \forall k \in V \quad (21)$$

The objective function for the model in equation (13) aims to minimise the total cost incurred to transport the goods to the customers. Constraint set (14) shows that each vehicle visits a customer exactly once. Constraints sets (15), (16) and (18) contain information on vehicle capacity and record the starting point and endpoint for the trips as the depot. Constraint set (17) is where the arcs store information on the arrival and departure of the train. The final three constraint sets are the calculation of service time for the customers and assigning the arcs x_{ijk} to binary values.

With the intention of minimising the route duration, Savelsburgh (1991) researched implementing edge-exchange optimisation methods for the VRP. This was in addition to previous work which showed that in special cases of single vehicles, i.e., the TSP with time windows, constructing a feasible route is already NP-hard (Savelsburgh, 1986). As a result, most of the research in this area has been directed towards controlling the feasibility of a set of routes. The author argued, however, that instituting time spans for customers also makes it possible to define objective functions which are more practical (in comparison to minimising the range or distance), such as minimising waiting time, minimising the time the service is completed, and minimising the time period for paths.

2.5.2 VRP solution methods

To solve the VRP with an objective of minimising cost for the delivery of goods to customers using a homogeneous fleet of vehicles, Osman (1993) developed a hybrid method of the TS and SA algorithms. The SA part of the algorithm imposes a random search, which includes the acceptance and stopping criteria to find a better local minimum on the local search. The TS seeks to improve on the local optimality and is based on a function which chooses the maximum estimation move in terms of the objective function TS limits.

Ombuki et al. (2004) formulated a VRPTW as problem with multiple objectives and developed a GA to find a solution. The GA is outlined in Figure 16. In the GA, the transformation of each chromosome in the population occurs. These chromosomes are transformed into a group of routes. The chromosomes are iteratively processed in a repetitive developmental process until a minimum feasible number of route groups is determined or until the stopping condition is met. The algorithm used location, time windows, vehicle capacity and customer demand as inputs and created an output of routes to be travelled by the vehicle during the given time windows. The final results for this model showed a reduction in cost and were perceived as positive.

```
procedure Genetic routing system;
begin
  Read instance data(customer size and locations, time windows,
capacity);
  Set GA parameters;
  Generate randomly an initial population, POP of size popSize;
  for gen = 1 to maxGen do
    Transform each chromosome into feasible network configuration
by apply-     ing the routing scheme;
    Evaluate fitness of the individuals of POP;
    Rank the individuals of POP and select new population with
the elite     retaining strategy;
    Apply GA operators (crossover and mutation);
  endfor;
end;
```

Figure 16: The GA pseudocode for a VRP (Ombuki et al., 2004).

Akgül (2020) developed a VRPTW model with a homogeneous fleet similar to that of Kallehauge (2006). This model was solved using MS Excel Open Solver. Mason (2012) wrote that MS Excel Open Solver allows LP and IP models developed in Excel to be solved using the CBC (COIN-OR branch and cut) solver. The CBC is written in C++ and the model can be programmed and used in other tools such as A Mathematical Programming Language (AMPL) (Mason, 2012). The objective was to minimise the total delay that the vehicles were allowed to incur, given the constraints.

Savelsburgh (1991) used the neighbourhood search algorithm, commonly used in solving the generic TSP, to solve the VRP mentioned in section 2.3.2. The aim of the output is to show how the neighbourhood can be used to minimise route duration, travel time and completion time, and what effect each objective will have on each of the variables. From the results, it is evident that the objective function variables are minimised for the function that the variables are a decision variable in. Looking at the table, where the objective is to minimise travel time, the travel time as the lowest of the three objective functions.

Fleszar et al. (2009) also used the neighbourhood search algorithm to find an optimal solution for an open vehicle routing problem (OVRP), where the vehicle does not travel back to the depot after service to the final customer, unlike normal VRP models. The authors based the neighbourhoods on withdrawing segment routes and replacing segments between the different routes. The initial current result, a, is determined by using the local search technique for the first result, a_0. The process then iteratively determines an arbitrary solution from a neighbourhood with a size, k, and it determines a current local minimum beginning at the arbitrary solution, which was previously generated and keeps it only if it has a value which is better than the value of the new solution respective to (O, T). This is repeated r times. The model's solutions were compared to 16 other heuristics and Fleszar et al. (2009) concluded that the neighbourhood search algorithm could be used to solve many other transport and logistics problems.

Jolai and Aghdaghi (2008) proposed a model for a multi-objective single vehicle VRPTW with multiple routes to traverse. The objectives were to minimise the total cost to serve customers, maximise the number of served customers, minimise the total waiting time, and to avoid underutilisation of the vehicle capacity. The authors developed a heuristic technique to find solutions to the model. The heuristic is an adaptation of the shortest path algorithm with resource constraints. The consumed resource is part of the criteria used to compare performance of the routes. The algorithm is broken down into two phases. Phase 1 is where a feasible route is generated. In phase 2, the path generated in phase 1 is used to create a full workday for the vehicle. The heuristic could only solve VRP to a maximum of 50 customers and therefore the authors suggested future work where the algorithm can be developed for more than 50 customers.

Dhoruri et al. (2013) also presented the capacitated VRPTW as a multi-objective model to be solved using a GP model. The capacitated VRPTW vehicles have a limited carrying capacity of goods. The objectives were maximising the vehicle capacity and number of clients but to minimise costs and waiting times. The results were obtained by programming the model on LINGO. The solution showed that at least three vehicles were needed to satisfy the customer demands, but this decreases to two vehicles when the total capacity of the vehicles increases.

2.6 Summary

This research is about using MILP to optimise railway capacity on the NatCor. The research is focused on analysing how capacity is used on the NatCor by studying the parameters listed in section 2.3. In section 2.4 it was noted that from the parameters discussed in section 2.3, schedules were an important factor for railway capacity optimisation. Lindfeldt (2015) states that the timetable is essential for the analysis of structured operation and analysing an existing or creating a timetable to be analysed would be a “natural first step” in the process. From the literature it was evident that an effective way to utilise train timetables and schedules would be to formulate a TTP as a MILP approach to analyse the schedule. Cacchiani and Toth (2012) noted that the TTP aims to achieve efficiency in rail operations by minimising travel times for customers Lindfeldt (2015) agreed and noted that the timetable contains information on all operations that happen on the line. The TTP is a complex problem to solve and Arenas et al. (2015) defined it as an NP-hard complex and a time-consuming optimisation process.

The literature also identifies the VRPTW as NP-hard complex, but it is easier to solve and model. The VRPTW has been studied for more than 20 years and, because of this, multiple solution methods for the model have been identified.

In this dissertation, the timetable will be used to analyse the capacity utilisation on the NatCor. A TTP will be formulated to optimise the capacity utilisation on the NatCor. To find an initial feasible solution to the TTP, it will be converted to an equivalent VRPTW to minimise total elapsed time for the trains.

3. Methodology

This chapter explains the steps that were followed to conduct the research. The research followed the seven-step decision-making framework of Zandin and Maynard (2000). The framework is shown in Figure 17 and consists of a feedback loop to keep updating the progress of your work with the passage of time.

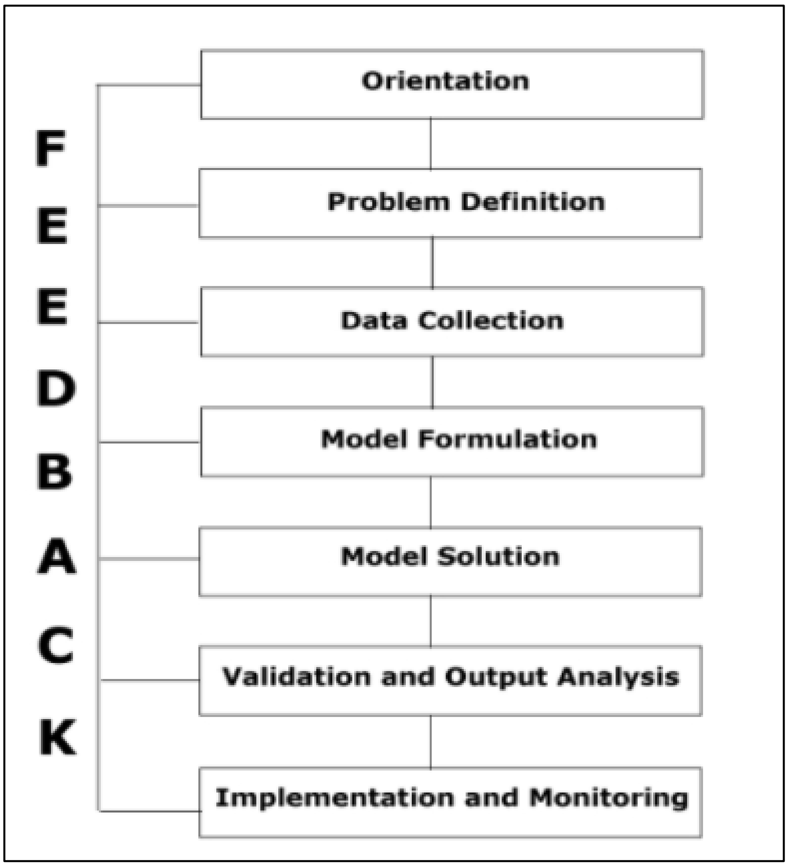


Figure 17: the operations research framework (Zandin and Maynard, 2000).

The first step is where the problem was introduced, and a clear picture of the issue was provided. The second step is where a clear definition of the problem was stated. The first two steps were covered in the first two chapters and were restated in section 3.1. Section 3.1 explained the research approach. This approach serves as a detailed description of the type of study this was and the reasons behind this choice; this included the scope and desired results. Data was then collected in step 3, with the aim of translating the problem into a model that can be analysed. This data was cleaned and prepared to be used in the model. The problem was then formulated into a solvable model, being a mathematical model. The model formulation step was useful because models are easier to analyse than the original system. Next, the model was solved using the CBC algorithm, and the solution was validated in step 6 by first verifying that it makes sense and then ensuring that the model is an accurate representation of the system. The final step was the implementation of the result and recommendation and monitoring of the system.

3.1 Research approach

The approach taken to conduct this research is illustrated using an adaptation of the Research Design Onion (Saunders et al., 2006): for an illustration see Figure 14. The full research onion provides a rather exhausting description of the main layers or stages which must be accomplished to formulate an effective methodology (Raithatha, 2017). However, the adaptation in Figure 18 is a summary containing only the final research design used in this dissertation.

Based on Figure 18, the study used an inductive approach to determine how capacity can be optimised on the rail line. Although this approach does not always give a fixed solution, it is still advantageous as it allows the development of an optimisation model that will not be specific to the NatCor but can be adapted and used for other lines across the world. It allows for the observation of patterns on the line that might affect capacity.

The study was also a quantitative assessment of rail line data and how changes made to the rail line will affect the rail yard in the short term. The quantitative nature of the research helps to make the validation process of the data easier. Although most of the research was based on data received from the company, as opposed to results obtained strictly through observation, this was overcome by considering only raw data from the company that had not been processed, sorted or packaged to suit any specific narrative.

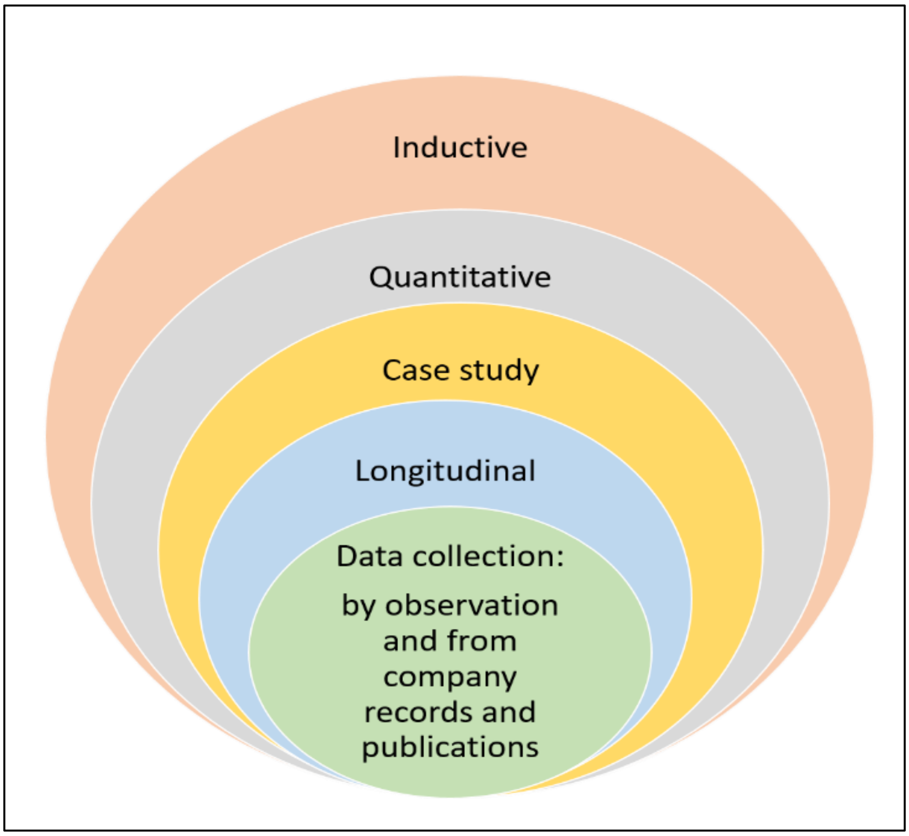


Figure 18: The Research Design Onion (Saunders et al., 2006).

To determine patterns on the line accurately, in relation to time, a longitudinal study was conducted using the NatCor and DCT as a real-life case study. The data used in this study was provided by a Transnet SOC Ltd and was compiled by the Operations and the Scheduling Departments. Using real data from the NatCor as a case study makes it more factual – as opposed to using arbitrary data. This also made the research easier to verify, which will make the research more credible.

3.2 Data collection

The data collection process is the third step in the seven-step decision-making framework described in Figure 17. The process started with meetings with the operations team to understand different operations on the NatCor line and at the DCT. Documentation of the as-is model of the line was received from the team. This was followed by a site visit to the City Deep yard. This yard visit was important because the City Deep yard is the last yard where trains from Capital Park pass before joining the NatCor at Rietvallei, and this is illustrated in Figure 19. The City Deep yard visit also helped validate whether the timetable data provided was accurate. A visit to the DCT was not necessary for the study because it does not include an analysis of the DCT.

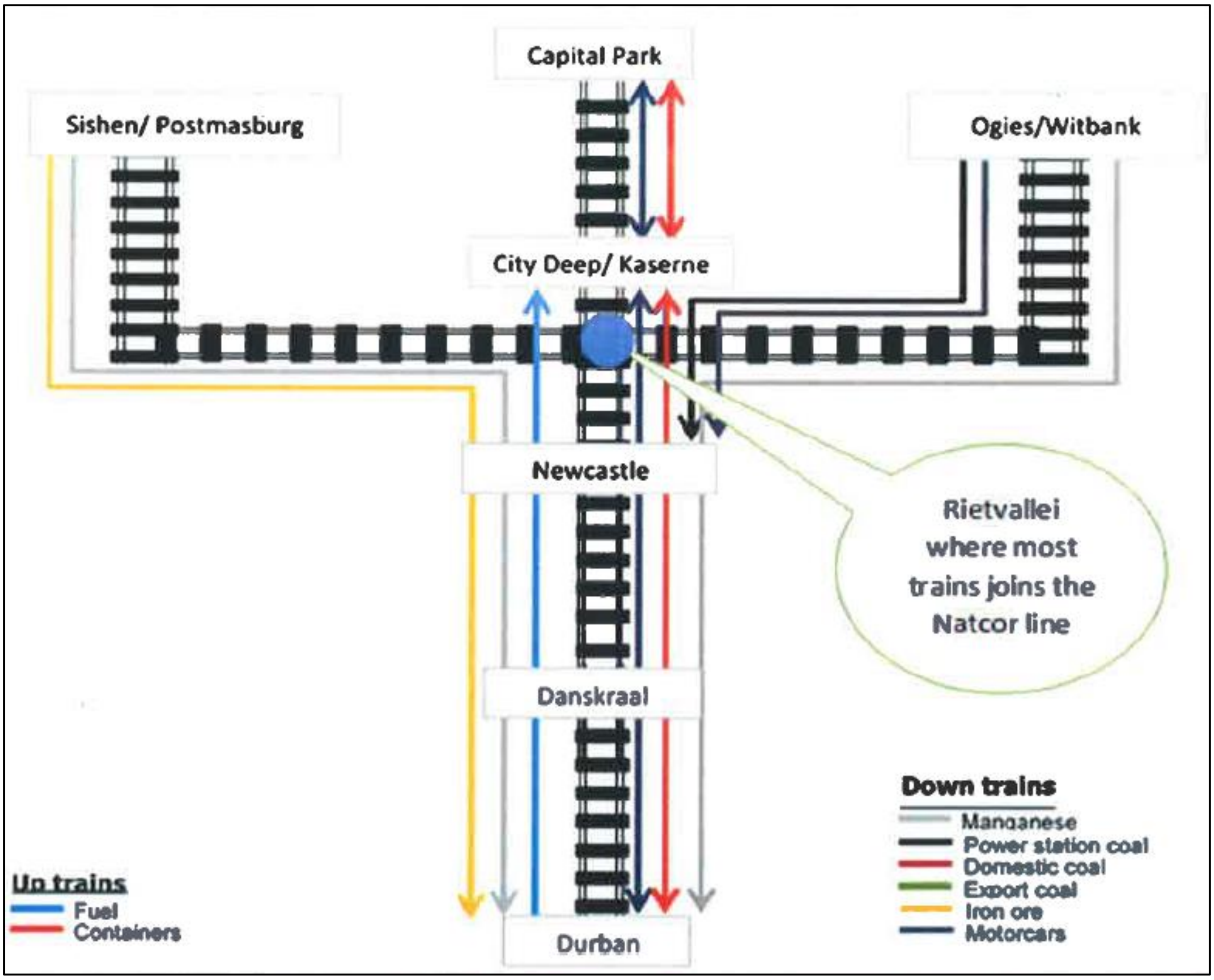


Figure 19: An illustration of how trains run on the NatCor line (Transnet, 2019).

The next stage of data collection was gathering signalling data. A meeting with the Control and Signalling team was held and documents were shared explaining all the controls found on the corridor and how they affect the schedules. There are a total of 26 train signals which ensure that headway times are adhered to and there are six major signals (traffic lights) at the end of each section of the corridor safely and efficiently regulate the train movement. These signals feed back to a central Operations Command Centre, where they are monitored.

A third meeting was scheduled with the scheduling team. In this meeting, all current schedules and timetables relating to the NatCor were provided in raw data format. The schedules included staff book-offs (shift information) and train maintenance schedules, as well as a list of all the trains (referred to as the catalogue) that have been running on the NatCor line since as far back as 2009. This was to keep track of all active and decommissioned trains in the data clean-up process.

3.3 Data analysis approach

3.3.1 Data correlation

The first step to analysing the data was to compare the data received from the different departments to each other in order to check if the information received correlated. This process revealed that the departments had different records of the same data due to a lack of communication between the departments and the action of working in silos.

Some of the data received from both the operations and scheduling departments was old and outdated, so rendering it unusable. Data older than three years was not a reflection of what had been happening on the NatCor and was not relevant for the plans mentioned in the LTPF (Transnet SOC Ltd, 2016). This made the data cleaning process lengthy and this prompted more communication with the relevant departments to confirm data accuracy. The removal of the duplications was an effective way of decongesting the rail line where non-existent trains were taking up capacity slots

Identifying and fixing the limitations assisted in building more accurate mathematical models that were usable for the company. This was done to avoid tainted results that could have influenced the accuracy negatively. This undoubtedly resulted in more efficient and reliable decision making.

3.3.2 Data cleaning

Some of the data received from the company contained inaccurate schedules and incorrect train catalogues. Duplicate train scheduling was clogging the schedules and some trains were missing from either the schedule or the catalogue. Receiving raw data from the company made it difficult to process and more meetings were held with the relevant department to understand what was causing the inaccuracies. The department head confirmed that the company had been using the same data files without updating them for years. This meant that trains that had been discontinued were still recorded and scheduled for a slot. It was also confirmed that trains that had been replaced with new ones, because they were decommissioned, were not removed from the schedules, and thus the new ones added caused duplication on the catalogue and the schedule and confusion about clashing running times.

After confirmation of which information was still relevant, the data was cleaned and freed of all duplication and inaccuracies. This process was done by using MS Excel and Python 3.9. These tools were chosen because they are user-friendly, and Python supports analysis of large and complex data sets very well and is compatible with many other software tools used for mathematical modelling (McKinney, 2012).

The new timetable resulting from this process can be found in Addendum B.

3.3.3 Model formulation

In Chapter 2, three parameters were discussed as important parameters. These affect rail line capacity. The parameters were identified as train types, schedules, and infrastructure. Schedules were identified as the most important parameter to analyse as they contain information on how and when the rail line infrastructure is used, the types of trains scheduled to run on the rail line, and the journeys travelled by the trains. Thus the NatCor line was modelled as a TTP. The MILP model minimises the total delays experienced on the corridor. The timetables encompass all the information from the different departments that were sourcedfor the corridor data provided in Chapter 3.

From Chapter 2, it was evident that the TTP and the VRP are derived from the TSP and therefore belong in the same family of NP-hard problems (Christofides, 1976; Abbas-Turki et al., 2011). The VRP family of problems share more similarities with how the TSP models are formulated and solved. This makes the VRP easier to solve than the TTP, which is seen as a far more complex problem to solve. Owing to this high level of complexity of the TTP compared to the VRP, the TTP model was converted into an equivalent VRPTW model in order to find an initial feasible solution.

The results of the VRPTW were further improved using the TS metaheuristic. The heursiw

3.3.4 Model verification

The model verification process started with the processing and cleaning of the data. The process also involved testing if the formulated models work using an existing code from the models analysed in the literature. The models were formulated on MS Excel Open Solver 2.9.3 64 bit, which uses the CBC algorithm to solve large MILPs (Mason, 2012; Akgül, 2020). MS Open Solver is a software tool that handles models with a large number of variables. These models are often not solvable on the regular Excel solver. The main aim of the verification process was to make sure that the model was built as accurate as possible.

3.3.5 Validation

The model validation process was done to ensure that the model used was indeed a good model to estimate the estimated travel time. The validation process done took the results obtained from the improved model and compared these results to those obtained using real-life data from the clean timetable in addendum B while receiving input from the Scheduling department to confirm the accuracy of the results. The main aim of the validation process was to ensure that the model used was the correct model and provided correct estimates of the travel time. The full description of this process is discussed in section 5.3.5.

3.4 Input data

This section contains input data collected from Transnet.

3.4.1 Line variables

A summary of the line characteristics is found in Table 2. Although the line is considered a double line, it consists of only one track of trains going up from Durban to Rietvallei and one track of trains from Rietvallei to Durban. The average speed of the most of the trains is 50km/h, however there are shorter trains that run at speeds that reach 80km/hr. The 50 km/hr is used to accommodate the longer trains which travel at lower speeds. It has been tested and verified that decreasing the current speed for the 40–65 wagon trains makes decreasing headways possible, and the slower the train the shorter the time it takes to slow and cool down to a stop, so implying shorter headways (Transnet, 2019). However, decreasing the speed causes an increase in delays and therefore the research will test the implications of decreasing the train running times on the corridor and how it will affect delays, headways and capacity on the NatCor line.

Table 2: A summary of the NatCor line characteristics.

Line Characteristics	
Line type	Double track line
Average train running time (between Johannesburg and Durban)	20 hours
Average operating speed	50 km/h
Axle load	20 ton/axle
Number of train slots	72 slots (20 minute intervals on the lines to schedule trains)

Table 3 lists the NatCor line section divisions.

Table 3: The NatCor line sections and length (Transnet, 2019).

Section	**Distance**
Rietvallei to Perdekop	189 km
Perdekop to Newcastle	107 km
Newcastle to Glencoe	68 km
Glencoe to Danskraal	64.31 km
Danskraal to Marianhill	267 km
Marianhill to Booth	20.75 km

These sections have different capacity slots available. This is because of the number of temporary speed restrictions on the different sections and trains that join the NatCor from feeder lines on those sections, which are shown in Figure 5. Transnet has identified the critical points on the line where most delays occur to be the sections between Glencoe and Marianhill. The Danskraal–Marianhill section, in particular, has passenger trains operated by PRASA (Passenger Rail Agency of South Africa) joining the line at Cato Ridge going to Durban. This causes more delays for Transnet trains.

3.4.2 Train types

The trains running on the NatCor range from 25 to 105 wagons. Depending on the size of the train, they run at different speeds and have different headway times. The shorter trains, at 25–75 wagons have a headway time of 20 minutes. Trains longer than 75 wagons have a headway time of 30 minutes. This because the longer the train, the more time it takes to slow down and cool off when coming to a stop.

The types of trains running on the NatCor line are also be differentiated by the types of wagons they use, and the commodities being carried in the wagons. There are different commodities running on the line. These range from jet fuel, motor vehicles, coal, manganese ore, cement, bulk products, iron ore and lime to steel – and there are two passenger trains that run on the NatCor. The types of trains are listed in Addendum A and Figure 18 shows the number of trains on the schedule classified by commodity.

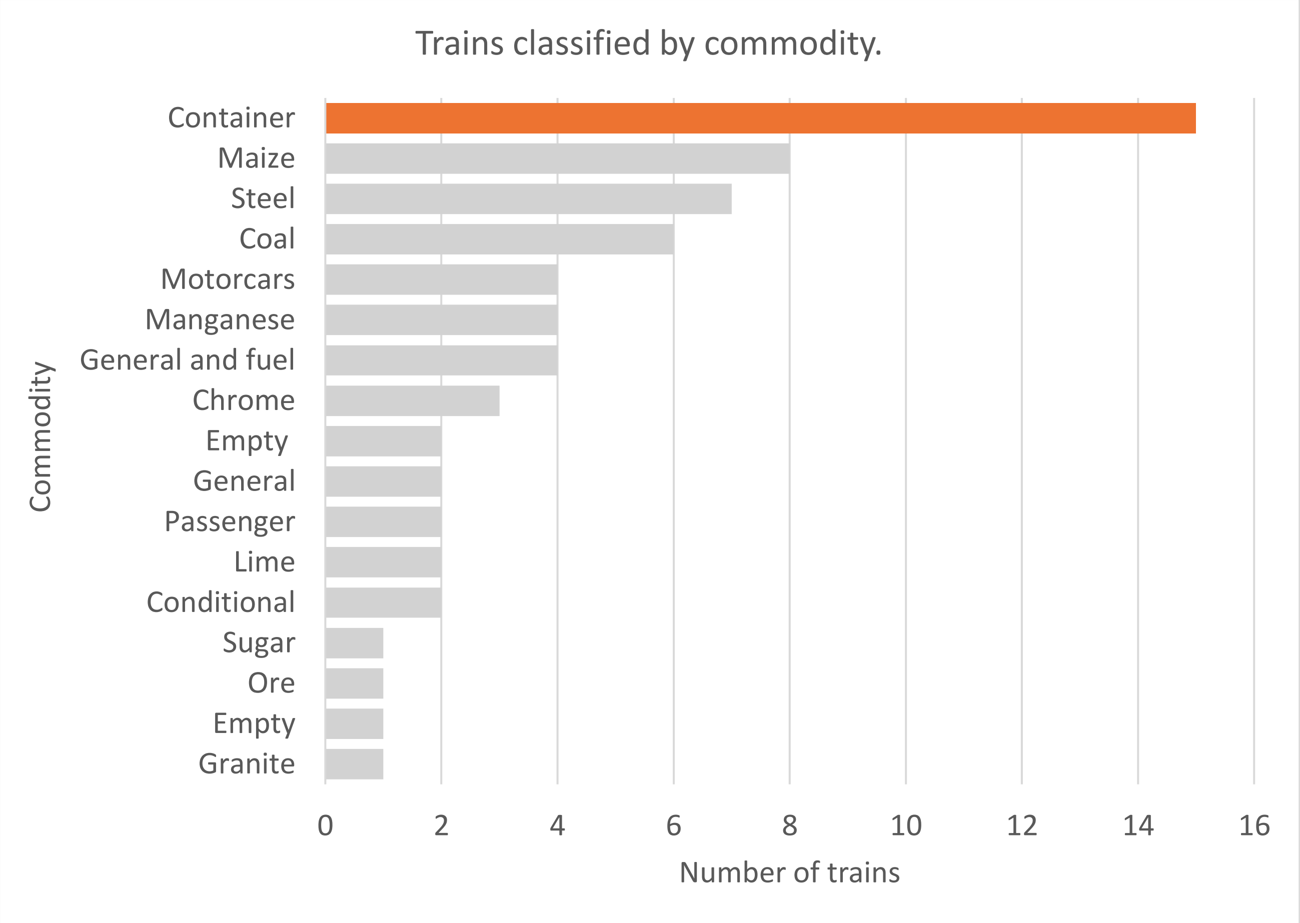


Figure 20: Train types classified by comby commodity

From Figure 20, it is evident that container trains take up most of the capacity on the line. These are 50 wagons trains that end up at the Durban Container Terminal. Classification of the trains by lengths is shown in Figure 21. From the graph, most capacity on the line is taken up by the 40 and 50 wagon trains. The 50 wagon trains are the container trains that the company has stated that it plans to make longer in the future.

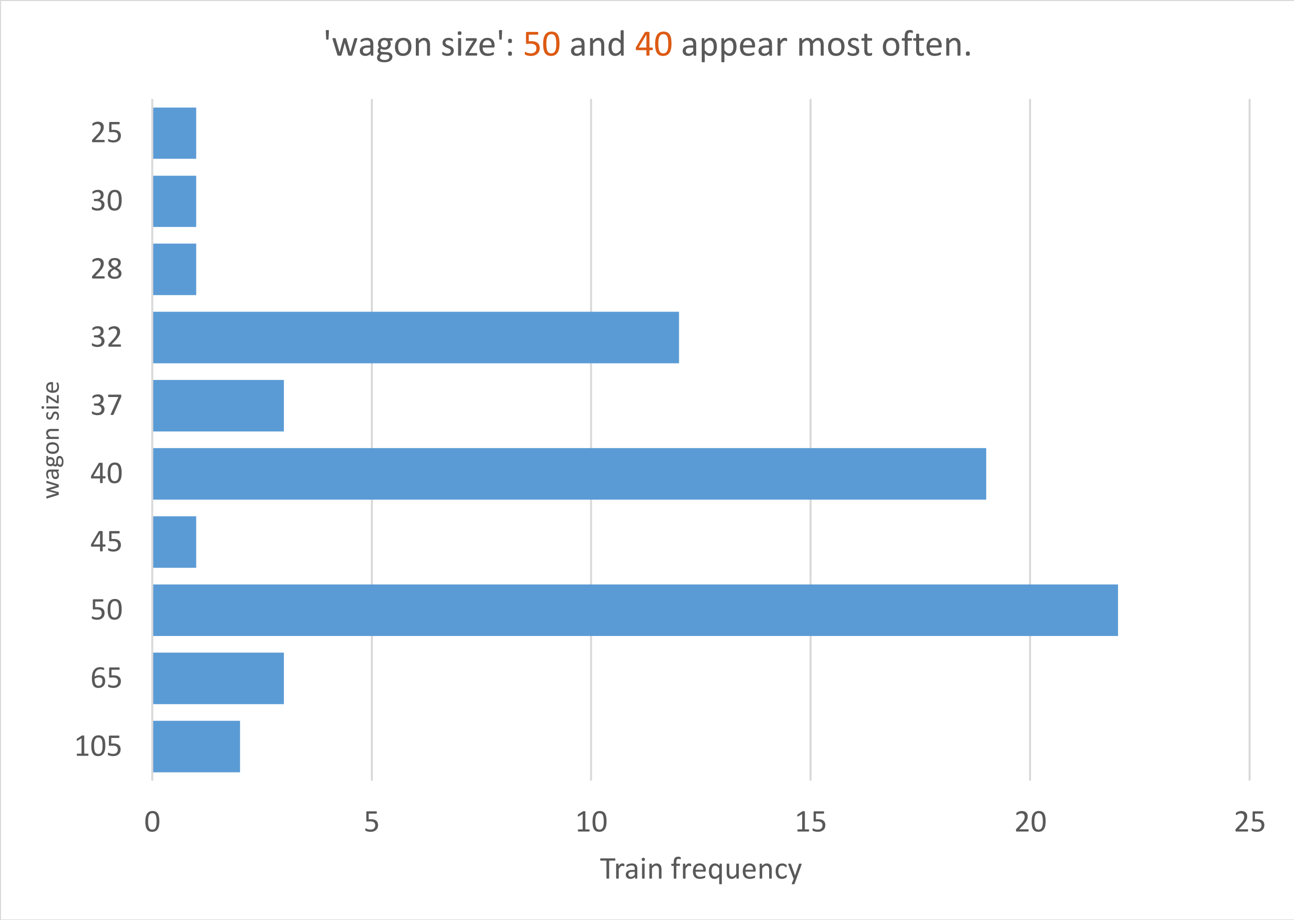


Figure 21: A classification of trains by length

3.4.3 Train slots

Only 65% of the total 72 slots is currently used for freight services after considering the slots lost to the temporary speed restrictions and slots reserved and used for maintenance. This leaves an average of 39 slots available on the NatCor line per day. Figure 16 explains the slot losses and shows the availability of slots on the NatCor line.

The lack of available capacity slots on the NatCor currently poses a problem for the rail company. This, together with the delays experienced on the line, causes capacity over a 24-hour period to drop even further. The drop in capacity means the company cannot meet its demand requirements.

Addendum A shows the current timetable of trains going from Rietvallei to Durban. It consists of 67 trains that run over a 24-hour cycle. This means that with the current available capacity on the line, shown in Figure 22, 28 trains on the timetable cannot be accommodated on the corridor.

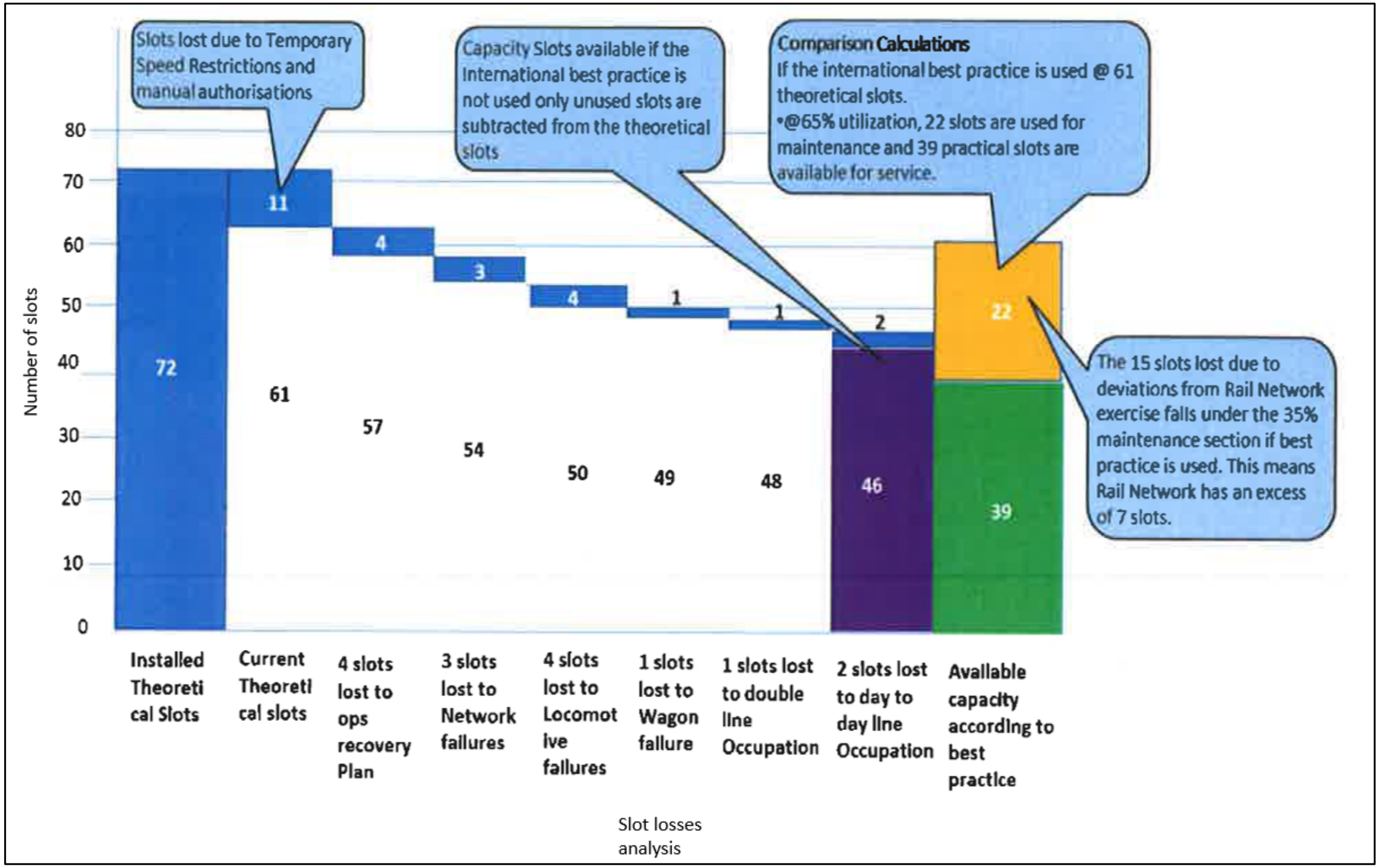


Figure 22: The NatCor slots availability analysis (Transnet, 2019).

3.4.4 Current train schedule

Figure 23 illustrates the portion of the current schedule where trains are running from Rietvallei to Newcastle. The lines running down diagonally from the left- to the right-hand-side represent the trains moving from Rietvallei to Newcastle (and some eventually moving to Durban). The trains are labelled in the yellow blocks on the diagram. The time from 00:00 to 24:00 is shown on the x-axis and all the stations passed by the trains between Rietvallei and Newcastle are written in blue on the y-axis. Rietvallei is the top station and Newcastle is at the bottom.

A comparison of the timetable in Addendum A and the schedule in Figure 23 reveals that some trains are missing on the schedule and there are trains on the schedule that are missing on the timetable. This is because the company is using an old schedule and is just adding new trains without removing old trains with cancelled services on the line – as explained in Chapter 1 and Chapter 3.

3.4.5 Final train schedule

The final timetable (cleaned and free from duplications) for the NatCor line and which was used in the model calculations is shown in Addendum B.

The new timetable shows a significantly lower number of trains on the NatCor line. These trains are more than the current available capacity on the NatCor line and therefore still cannot all be accommodated on it. The average travel times for these trains between the six sections on the line can be found in Table 4, where M is an arbitrarily large number

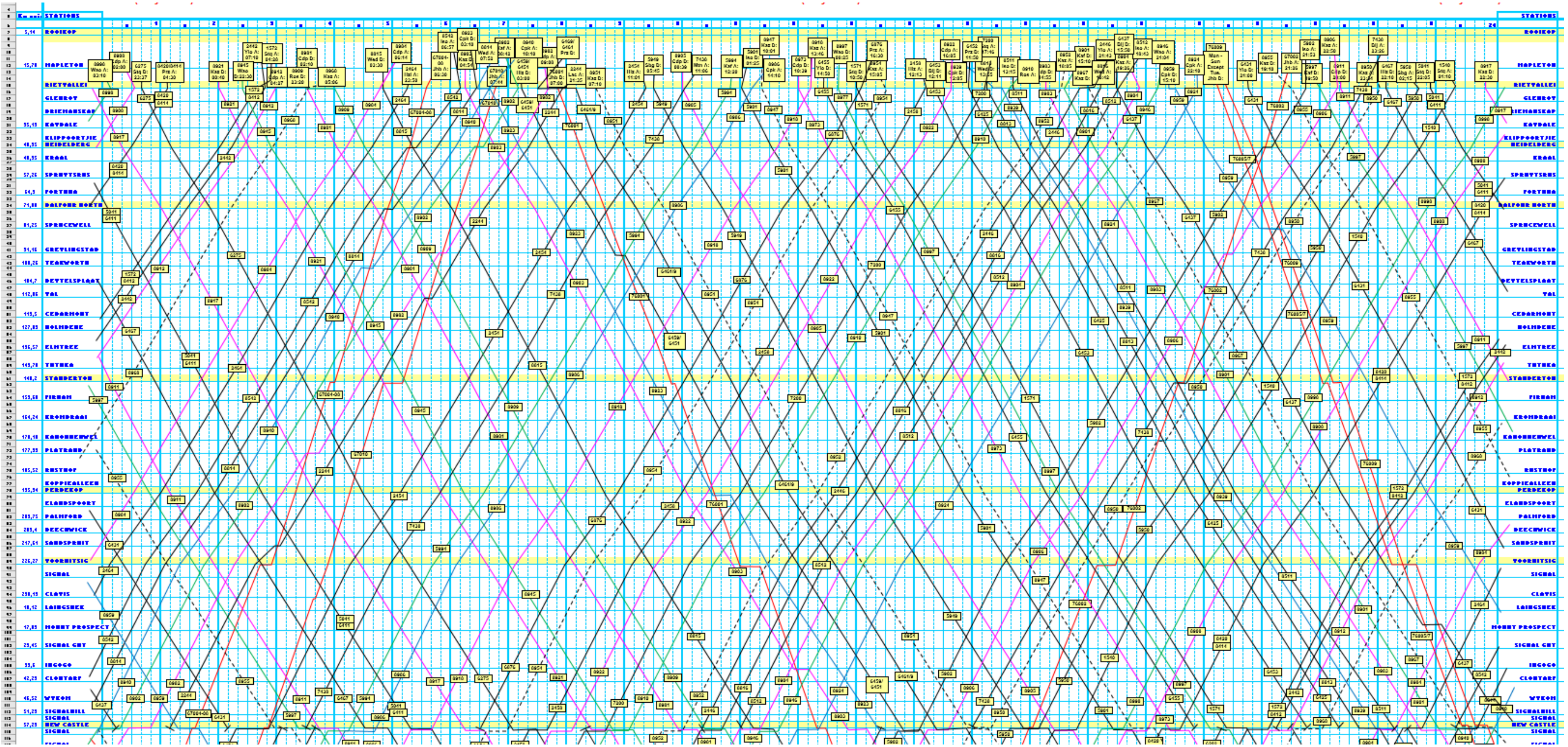


Figure 23: Current NatCor train schedule diagram (Transnet, 2019).

Table 4: Current average travel times for trains on the NatCor.

	Rietvallei	**Perdekop**	**Newcastle**	**Glencoe**	**Danskraal**	**Marianhill**	**Durban**
Rietvallei	M	3.4	5.68	6.91	9.08	15.3	15.88
Perdekop	3.4	M	2.28	3.51	5.68	11.9	12.48
Newcastle	5.68	2.28	M	1.23	3.4	9.62	10.2
Glencoe	6.91	3.51	1.23	M	2.17	8.39	8.97
Danskraal	9.08	5.68	3.4	2.17	M	6.22	6.8
Marianhill	15.3	11.9	9.62	8.39	6.22	M	0.58
Durban	15.88	12.48	10.2	8.97	6.8	0.58	M

Using, jTrainGraph v3.4.1, a software tool that enables users to create train schedules using train timetables, a train schedule diagram was produced for the timetable in Addendum B (jTrainGraph, 2021). The schedule is illustrated in Figure 24. The times are on the x-axis running from 00:00 until 24:00. The main stations for the different sections are on the y-axis and the train labels are shown on the graph diagram. The trains run from left to right.

The schedule shown in Figure 24 highlights a significant gap between some of the trains, which means there is enough headway time and not as much congestion as was evident in Figure 23. This makes it possible to run longer trains or to add more trains to meet demand for the rail company. There are some trains which run in close proximity to each other, and this is because of the maintenance slots illustrated in Figure 22. The maintenance slots take up capacity on the line and therefore cause the 43 trains to be closely scheduled.

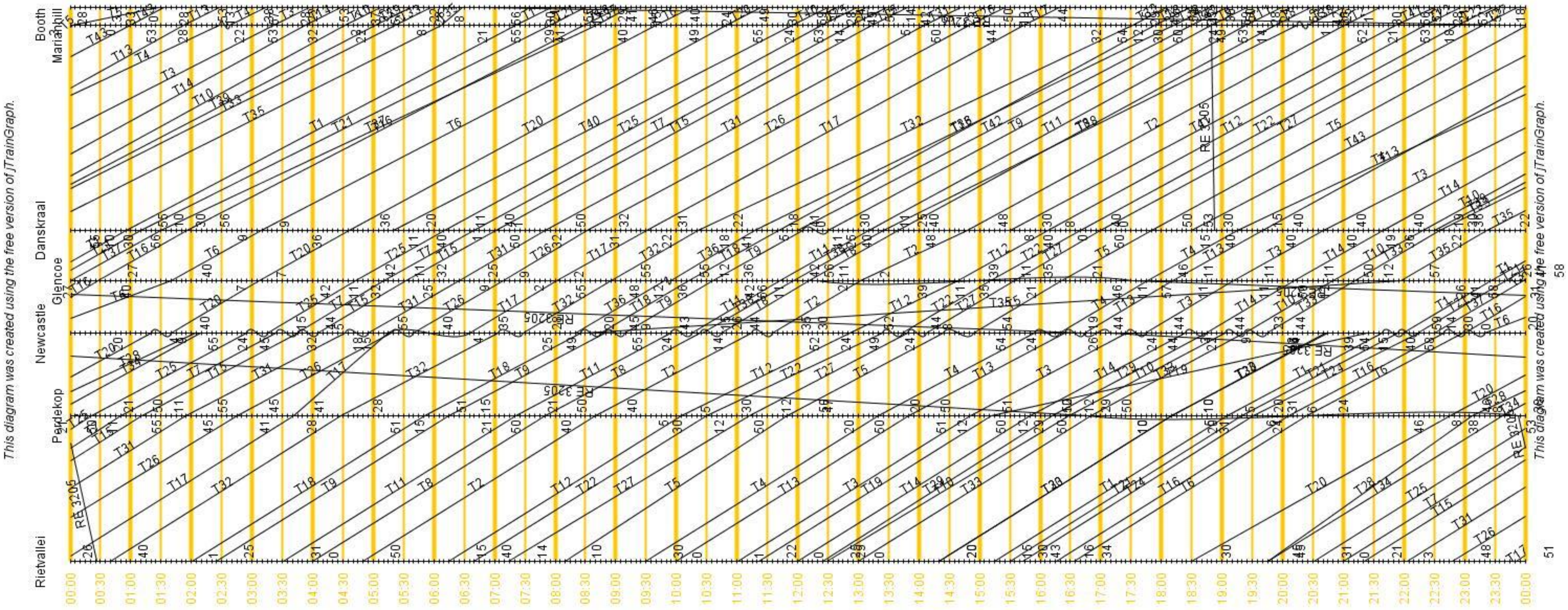


Figure 24: Train schedule derived from a clean timetable.

3.4.6 Train paths

The next step in the data cleaning and sorting process involved tracing the different paths traversed by each train. Addendum C illustrates all the possible different paths that can be travelled by the trains through the different sections highlighted in Table 3 and the total distance of each possible path. From the data provided, there is a total of 31 possible path segments that a train can travel on the NatCor line. The table was generated using Python 3.9. A screen shot of the partial code is illustrated in Figure 25.

```python
#this is just for combinations
import itertools

#just cleaning data
locations = []
for index,row in dfRoute.iterrows():
    locations.append(row[0].replace(' ','').replace('-',',').replace('–',',').split(','))

#seperating data into A and B
locationA = []
locationB = []
for loc in locations:
     locationA.append(loc[0])
     locationB.append(loc[1])

#removing duplicates
tempPaths = list(itertools.product(locationA, locationB))
Paths = []
for path in tempPaths:
    if (path[0] == path[1]):
        pass
    else:
        Paths.append(path)

#calculating distance
distance = []
for path in Paths:
    distance.append(CalculateDistance(path[0]+' ',path[1]+' '))

#route names
RouteNames = []
for x in range(len(Paths)):
    RouteNames.append('R'+str(x))
```

Figure 25: A screen shot of part of the Python code used to calculate train paths.

From the different path segments generated above, a set of paths P(t) was generated for different paths travelled by trains on the Natal Corridor. The set of paths shows how each train is allocated to a specific path. This is summarised in Addendum D and part of the code to achieve this is shown in Figure 26.

```
#Adding Route Codes
codes = []
for index,row in TimeTable.iterrows():
    Origin = row['Origin ']
    Destination = row['Destination ']
    for index,row in dfFinalRoutes.iterrows():
        if (row['Origin']+' ' == Origin and row['Destination']+' ' == Destination):
            codes.append(row['Route Code'])

finalDF = TimeTable
finalDF['Route Code'] = codes
finalDF = finalDF[['Train number ','Route Code','Origin ','Destination ','Departure time ','Arrival time ','Distance (km)']]
✓ 0.1s
```

Figure 26: A screen shot of the Python, which assigns trains to paths

Now that the relevant input data for the model has been cleansed, the formulation of the MILP model can begin.

4. Model development

In this section, the NatCor case study is modelled as a TTP and is converted to an equivalent VRPTW with homogeneous fleet to find an initial solution. The TTP was modelled using a MILP, which minimises the total delays experienced on the corridor.

The model used the clean timetable data in Addendum B as an input and seeks to optimise the capacity on the corridor by minimising the delay experienced on the line. For the NatCor line, the implication of decreasing the headway time to run more trains on the corridor or increasing the headway time to run longer trains on the corridor was studied, and the implication for the forecasted capacity demand was investigated.

4.1 Model assumptions

The following assumptions were considered when formulating the MILP:

- The average speed of trains on the line, which is 50 km/h to accommodate longer trains, is the running speed for all the trains.
- The timetable repeats itself every 24 hours and all trains run every day.
- Headway times are adhered to for the shorter trains as 20 minutes.
- Headway times are adhered to for the longer trains as 30 minutes.

4.2 Data sets, parameters and variables' definition

A definition and explanation for the sets and variables used in the models is provided in Table 5.

Table 5: Variables and parameters for the TTP model.

Sets And Variables	Definition
A_t	Arrival station of train t
B_t	Latest departure time for train t $(E_t + M_t)$

D_t	Departure station of train t
E_t	Estimated departure time of train t
$I_{t,s}$	Earliest arrival of train t to station s $(E_t + f_{t,s})$
$L_{t,s}$	Latest arrival time for train t at station s
M_t	Maximum delay allowed for train t
$X_{t,a}$	Arrival/departure arc of train t at time a
$X_{t,s,a}$	Delay arc of train t at station s and time a
$f_{t,s}$	Travel time of train t to station s
S	Set of stations
T	Set of trains
U	Set of discrete time intervals
X	Set of arcs
$P(t)$	Path of train t

The set of arcs X represents a set of all the arcs including the delay, departure and arrival arcs. This makes it a union for all the arc sets for all the trains in T and all times intervals $a \in U$ (shown on the table as time instances).

In a 24-hour period there is a set of arcs X that connects to different nodes on the corridor, starting from a departure node which represents the origin station of the train to an arrival node which represents the destination station of the train. This arc and node combination represents a path of the train on the line. Therefore, for each train $t \in T$ there is at most one path on the line.

4.3 The MILP Formulation

The final TTP model developed was a hybrid model using some elements from the TTP models developed by Liu and Han (2016) and Stanojević et al. (2010). The objective of the model was to minimise the total delay experienced by the trains on the Natal Corridor. The VRPTW model was developed from an adaptation of the model in Chapter 2 (Akgül, 2020).

The MILP is formulated to model the TTP of the NatCor, which is a double track line. The model seeks to minimise the delay d for every path $p \in P$ traversed by the by train $t \in T$. Ideally, for all train paths, the train would depart its origin station at the earliest possible time and any time deviation from that departure time should be kept to a minimum. Thus, making the final model:

Minimise z = $\sum_{t \in T} \sum_{p \in P_t} d_p x_p$ (22)

subject to $\sum_{p \in P_t} x_p \leq 1 \quad \forall t \in T$ (23)

$$\sum_{a=E_t}^{B_t} X_{t,a} \leq 1 \quad \forall t \in T \tag{24}$$

$$\sum_{a=I_{t,s}}^{L_{t,s}} X_{t,a} \leq 1 \quad \forall t \in T \tag{25}$$

$$x_p = \begin{cases} 1 \text{ if the the path } p \text{ for train } t \text{ is scheduled} \\ 0 \text{ otherwise} \end{cases} \tag{26}$$

$$x \in (0.1), \quad \forall x \in X \tag{27}$$

Where the delay d can be calculated as

$$d_p = \sum_{a=E_t}^{B_t} (a - E_t) * X_{t,a} + \sum_{a=I_{t,s}}^{I_{t,s}+M_t} X_{t,s,a}$$

The calculation for d is found using the scheduled timetable for the trains as well as the actual real-time data for the trains to obtain the total deviation. x_p are the axillary binary variables used to represent whether train t travels on the specified path p.

Constraint set (23) enforces that each train traverses only one path p in a 24-hour cycle. Equation set (24) ensures that for each train there is a unit outflow from its departure node to a node, which is the copy of its origin station. This happens at a time within the train's possible departure window and tracks dwell times at each node. The constraint set of equation (25) ensures that, for each train, there is a unit flow from one of the nodes, which is the copy of its destination node to its arrival node. The Final 2 constraint sets declare x_p as a binary variable and set the arcs x ∈ X as binary values.

The initial feasible solution for the TTP model, determined for the NatCor, will be sought out by converting the TTP model into an equivalent VRPTW. The conversion is a result of the high level of complexity of the TTP and its relation to the VRP, and the ease of solving the VRP compared to the TTP.

4.4 Converting the TTP to a VRPTW

Now that the TTP model has been formulated, the model is converted to an equivalent VRPTW to find an initial feasible solution.

The TTP can then be converted using the following logic:

- The set of trains represents the fleet of vehicles.
- The depot would be represented by the origin station for each train.
- The number of customers visited – this is just the destination point of each vehicle which will then be the equivalent of different destination stations for each train.
- The time interval and time windows are represented by the recorded departure and arrival times for each train from their origin to the destination points, as shown in the timetable.

Akgül developed a VRPTW model similar to that of Kallehauge (2006) in the literature review. An adaptation of this model was used to formulate the TTP as a VRPTW with homogenous fleet. It is important to note that an adaptation of the model was used because some of the constraints from the model were not relevant for the converted TTP. This model can be found in section 2.5.1.

In his work, Akgül (2020) then further converts the VRPTW with minimisation of cost into a VRPTW, where the total elapsed time is minimised.

The new complete model is given as:

$$minimise\ z = \sum_{k\in V}\sum_{i\in N}\sum_{j\in N} ET_{ij}x_{ijk} \quad (28)$$

$$\sum_{k\in V}\sum_{j\in N} x_{ijk} = 1 \quad \forall i \in C \quad (29)$$

$$\sum_{i\in N} x_{ihk} = \sum_{j\in N} x_{hjk} \quad \forall k \in V, \forall h \in C \quad (30)$$

$$a_i \le g_{ik} \le b_i \quad \forall i \in N, \forall k \in N \quad (31)$$

$$\sum_{k\in V}\sum_{i\in N}\sum_{j\in N} x_{ijk} \le \mathrm{U} \quad (32)$$

$$x_{ijk} \in \{0,1\} \quad \forall i, j \in N, \forall k \in V \quad (33)$$

This total elapsed time ($ET_{i,j}$) is given by the travel time ($t_{i,j}$), which is calculated by using the average speed the train travels at and the distances between the different stations and the waiting time at the customer. This can be seen as the equivalent of the total delay d_p in regard to the TTP model.

Constraint set (29) correlates with constraint set (14) on the original VRPTW. It is equivalent to the constraint set (23) on the TTP, which states that a train can only traverse a certain path once, but the big difference is that more than one train can travel the same path. Therefore, the new model conversion for constraint set (32) would be an equality where the summation of $x_{ijk} \le$ U (where U is the total number of trains, in this case 43. This is because all the trains on the line have a possibility of traversing the same path). This is an important and unique constraint set for the NatCor because most trains on the NatCor have the same path this constraint ensure that the paths summation does not exceed the total number of trains. Constraint set (30) corresponds with constraint sets (24) and (25) in the TTP model and ensures that the vehicle will depart from the same customer where it arrived (similar to how the trains will depart from the stations from which they had previously stopped). This makes the paths traversed by the trains/vehicles easily traceable. The final constraint (33) assigns the arcs x_{ijk} to binary values.

The waiting time at each customer is calculated as follows:

$$w_{jk} = Max\ (b_j * x_{ijk} - g_{ik} - t_{ij} - \left(b_j + g_{jk}\right),\ 0) \quad (34)$$

The models were set and are ready to be solved.

5. Results

In this chapter, the input data in Chapter 3 was inserted into the models developed in Chapter 4 and the results are reported. The results were obtained using the CBC algorithm built into MS Excel Open Solver. This method was used by Akgül (2020).

The method starts by creating a time matrix, which records the maximum time it takes to travel through the different NatCor line sections, while traveling at a specified speed. Schedules of the trains using the data from the travel time matrices were created for the NatCor.

The average real-time delay was then calculated using the timetable in Addendum B and actual real-time departure and arrival times to and from the applicable origin-destination pairings. This average real-time delay was recorded in Table 7.

The total travel time calculated by solving the MILP was recorded and time intervals (with minimum delays) were estimated for each train. The model was then verified and validated with real time data results from the clean time table.

The initial solution to the model was then improved by using the TS algorithm to find better estimates for the total travel time for each path.

5.1 Time matrices

There are different train types running on the NatCor line and these trains run at different speeds. The estimated average speed for trains running on the NatCor is 50 km/h. The maximum travel time for trains between each section, using the average speed of 50km/h, was calculated using the formula of Akgül (2020):

$$time\ (in\ hours) = \frac{distance}{\text{speed}} \tag{35}$$

The results were recorded in Table 6, where M is an arbitrarily large number. These results were compared to the results in Table 4. It is clear that the current average travel times (in hours) in Table 4 are higher than those estimated and recorded in Table 6. It is evident that the longest delays are experienced on the critical sections between Glencoe and Marianhill. This agrees with the critical points identified by Transnet.

Table 6: Travel times between stations (in hours).

	Rietvallei	**Perdekop**	**Newcastle**	**Glencoe**	**Danskraal**	**Marianhill**	**Durban**
Rietvallei	M	3.78	5.92	7.28	8.57	13.91	14.32
Perdekop	3.78	M	2.14	3.5	4.79	10.12	10.54
Newcastle	5.92	2.14	M	1.36	2.65	7.99	8.4
Glencoe	7.28	3.5	1.36	M	1.29	6.62	7.04
Danskraal	8.57	4.79	2.65	1.29	M	5.34	5.76
Marianhill	13.91	10.12	7.99	6.62	5.34	M	0.42
Durban	14.32	10.54	8.4	7.04	5.76	0.42	M

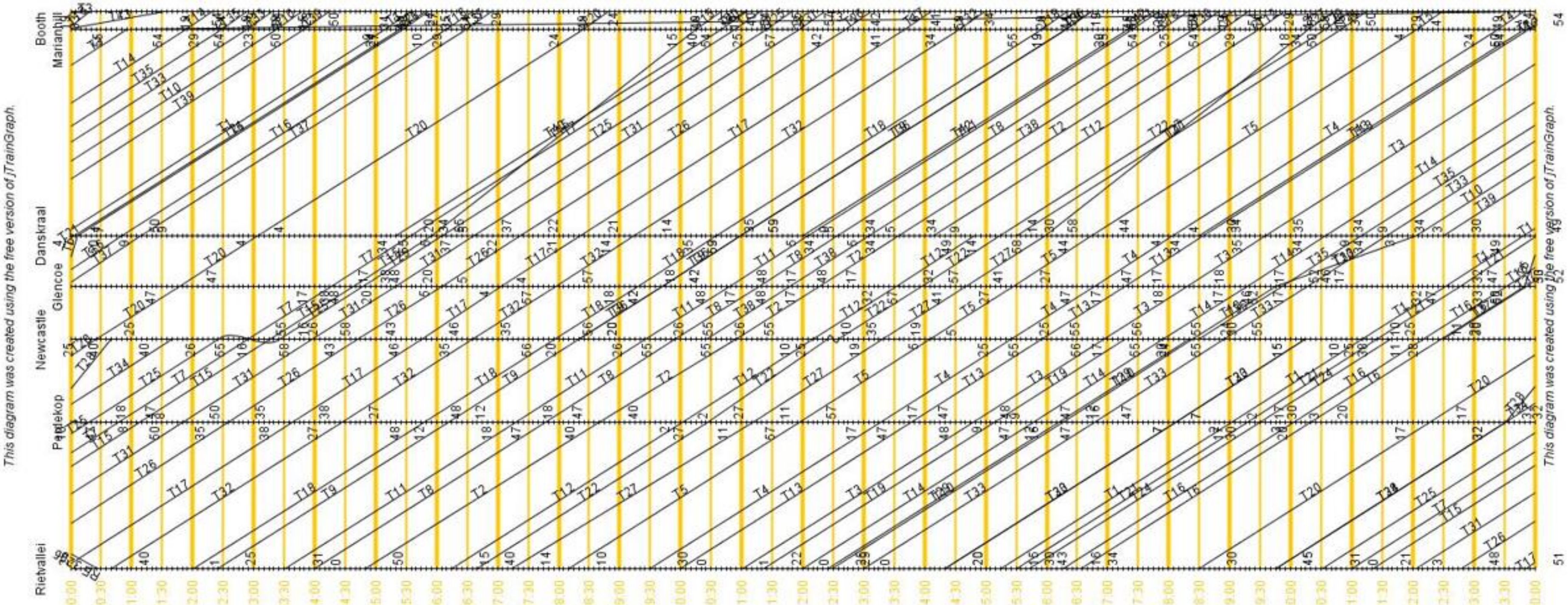


Figure 27: Schedule using distance matrix.

The schedule in Figure 27 is of a perfect scenario where no delays are taken into consideration and trains are assumed to be running at the calculated maximum travel time. This is the reason Figure 27 illustrates larger time intervals between the trains than those intervals in Figure 24. The results in Figure 27 create two options for Transnet:

i) Keep the intervals longer than the required 20 minutes of headway to accommodate an increased train length for container trains.
ii) Schedule more container trains of the current size.

The risk of leaving this schedule with little to no delays is that there is no buffer time for trains should unexpected events happen on the NatCor line that could lead to delayed operations on the line.

5.2 Timetable results

Table 7 contains the recording of average delays for each train on the clean and sorted timetable, over the past 12 months. The delay for each train was calculated using the train’s actual real-time departure and arrival times to and from their respective origins and destinations. The delay was calculated by subtracting the actual departure and arrival times from the scheduled times given in the timetable. The total delay for the trains excludes the 20-minute headway time.

Table 7: Train delays experienced in real-time.

Train number	Origin	Destination	Average delay (minutes)
T1 8813	Rietvallei	Durban	23
T2 8815	Rietvallei	Durban	7
T3 5901	Rietvallei	Durban	31

T4 5949	Rietvallei	Durban	62
T5 6461	Rietvallei	Durban	15
T6 8901	Rietvallei	Durban	22
T7 5997	Rietvallei	Durban	33
T8 8909	Rietvallei	Durban	66
T9 8945	Rietvallei	Durban	17
T10 6455	Rietvallei	Durban	23
T11 8981	Rietvallei	Durban	75
T12 8903	Rietvallei	Durban	12
T13 8905	Rietvallei	Durban	74
T14 8973	Rietvallei	Durban	15
T15 8911	Rietvallei	Durban	40
T16 8983	Rietvallei	Durban	90
T17 8917	Rietvallei	Durban	22
T18 8921	Rietvallei	Durban	54
T19 8947	Rietvallei	Newcastle	29

T20 8959	Rietvallei	Durban	18
T21 8939	Rietvallei	Durban	123
T22 8923	Rietvallei	Durban	19
T23 6453	Rietvallei	Newcastle	45
T24 8511	Rietvallei	Newcastle	25
T25 8955	Rietvallei	Newcastle	100
T26 5041	Rietvallei	Durban	61
T27 6451	Rietvallei	Durban	90
T28 6431	Rietvallei	Newcastle	55
T29 6455	Rietvallei	Newcastle	30
T30 6453	Rietvallei	Newcastle	99
T31 6467	Rietvallei	Durban	83
T32 6275	Rietvallei	Durban	59
T33 1571	Rietvallei	Durban	55
T34 6431	Rietvallei	Durban	27
T35 6201	Newcastle	Durban	11

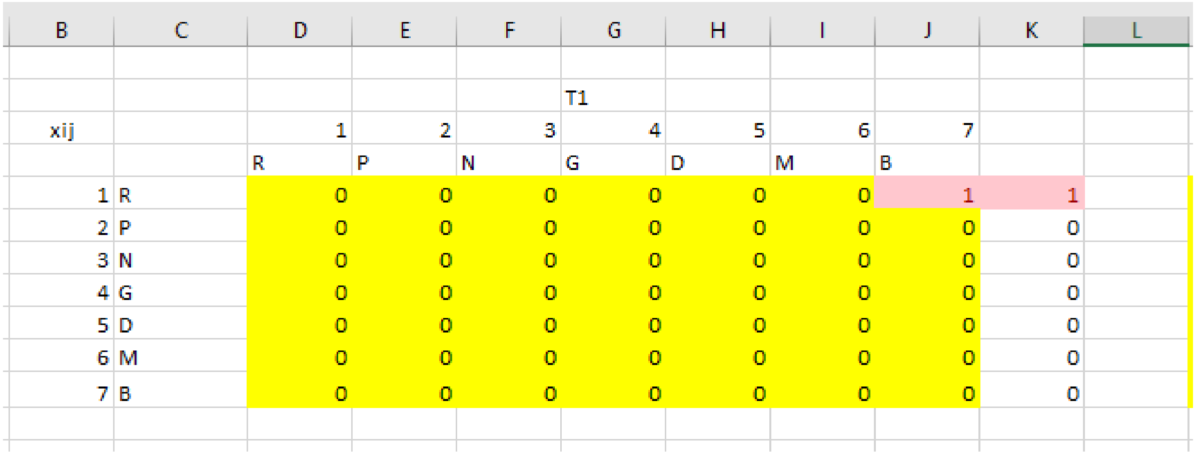
T36 5519	Newcastle	Durban	23
T37 1451	Newcastle	Durban	58
T38 6203	Newcastle	Durban	18
T39 5955	Danskraal	Durban	100
T40 5537	Danskraal	Durban	76
T41 5201	Danskraal	Durban	20
T42 2611	Danskraal	Durban	12
T43 6401	Danskraal	Durban	58

After the delays and the routes were calculated, the different variables and parameters were inserted into the developed MILP models.

5.3 MILP models' results

5.3.1 Results from the model using Table 6

On an MS Excel spreadsheet, each train/vehicle is assigned a path table, which shows a "1" representing the path where the train has traversed, and zero elsewhere. These tables represent the decision variable x_p in the TTP and x_{ijk} in the VRPTW.



B	C	D	E	F	G	H	I	J	K	L
					T1					
xij		1	2	3	4	5	6	7		
		R	P	N	G	D	M	B		
1	R	0	0	0	0	0	0	1	1	
2	P	0	0	0	0	0	0	0	0	
3	N	0	0	0	0	0	0	0	0	
4	G	0	0	0	0	0	0	0	0	
5	D	0	0	0	0	0	0	0	0	
6	M	0	0	0	0	0	0	0	0	
7	B	0	0	0	0	0	0	0	0	

Figure 28: A screenshot of the $x_{i,j}$ matrix for train T1

The next step was to calculate the average travel time for the trains. This was done using five trains at a time because the software crashed when trying to calculate many variables at once. The average travel time was calculated using the results in Table 6 and the delay times in Table 7 were used as the waiting times. These values were used for the calculation of ET_{ij} as illustrated in Figure 29.

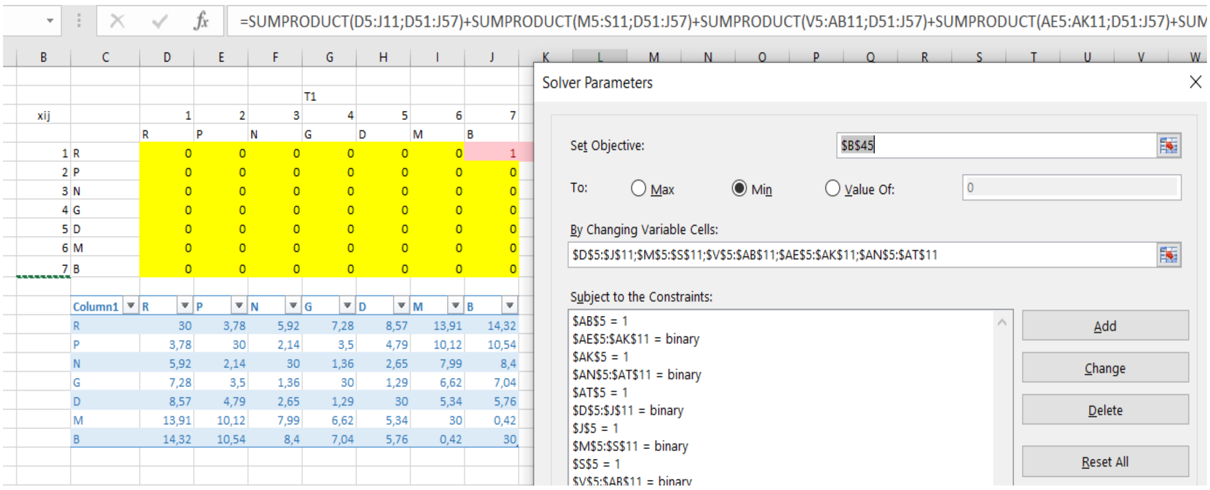


Figure 29: A screenshot of the variables, constraints and objective function.

The initial feasible solution for the VRPTW model in Chapter 5 shows that the value of the objective function in equation (28) is 523.397 hours. This is calculated using the sumproduct of the train path tables and the calculation for ET_{ij}, which includes the delay matrices.

These results represent a theoretical travel time usng the speed at which the trains travel and distance between the origin and departure station. The results do not take into account the delay incurred by the trains at signal points as well as the extra time trains use for accelerating and slowing down to stop. This is addressed in the validation process in section 5.4

5.3.2 Results from clean timetable data

Using the timetable in Addendum B, the average travel time (summarised in Table 4) and the delay time in Table 7 provided an estimation for the value of the objective function as 576.2237 hours. This was calculated using the same method as the one provided in section 5.3.1 and just replacing the different parameter values with the clean timetable.

These results mean that the current timetable has a maximum total time elapsed of 576.2367.22 hours for all the trains. This is 52.839 hours more than the initial feasible solution obtained in section 5.3.1. This means that the NatCor is currently running at a total delay value of 2.20165 days compared to the initial feasible solution.

It is important to note that with the current travel time windows, it would be impossible for the longer trains to be run because of the added delay caused by these time windows compared to the improved model. A suggestion would be for the company to study the timetable and to make some adjustments to run more optimal schedules using the initial feasible solution.

Running longer trains and/or increasing headway times is also highly dependent on the trains running on time and therefore sticking to the allowed delays and using that as a buffer time. From the Raw data provided by the company, the trains have not been departing at scheduled times and because the timetables used by the company are not robust, the delays accumulate.

5.3.3 Results using the old timetable

The old timetable referred to in this section is the raw data timetable using the trains listed in Addendum A. The timetable was provided by the company and has been used by the company for past 10 years only adding trains as the years have gone. This timetable has 67 trains scheduled to travel on the NatCor. Using the same travel time matrix on Table 4 and the same method explained in section 5.3.1 the estimated total travel time for trains on the old timetable is 892.68 hours.

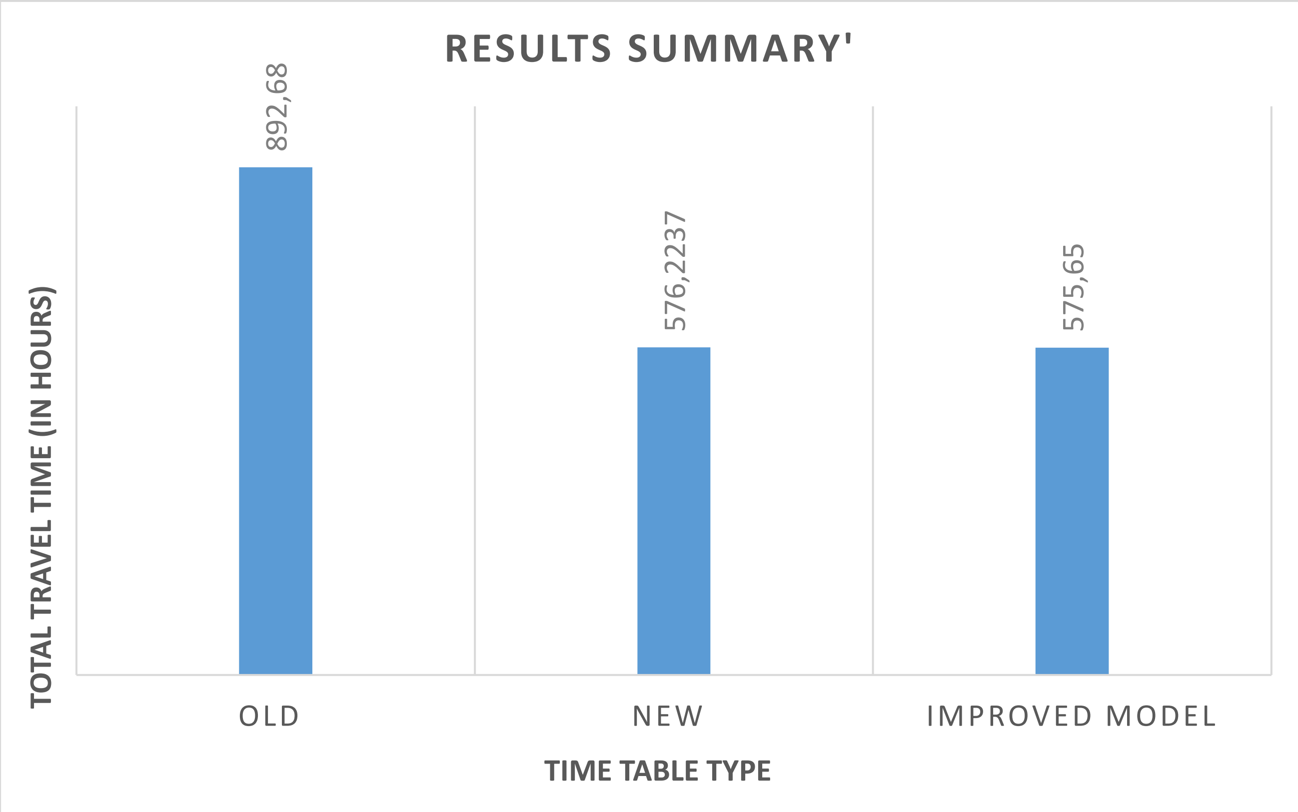


Figure 30: A summary of the results

Figure 30 shows a significant difference in the total travel times between the old timetable, the clean timetable and the model travel times. The model has the lowest total travel time. The old timetable is congested and contains discontinued and duplicated trains. The company has been using this schedule not realising that capacity slots have been taken up by trains that are no longer in service. Cleaning the timetable made a significant difference.

5.3.4 Model verification

The first step in model verification was to ensure that the purpose of the model and problem definition align. The model's objective function is to minimise the total elapsed time of trains, which is the sum of the total travel time and delays recorded. The elapsed time for trains informs how long the trains take up capacity on the line. Minimising the total elapsed time will help unlock more capacity slots to accommodate more or longer trains.

The model was then checked to confirm that none of the assumptions listed in section 4.1 were violated. The verified input data and the constraints ensure this.

The model was programmed in MS Excel Open Solver, which is able to handle more variables than the normal Excel Solver, and was tested to check if the infeasibility caused by changing some constraints would be detected.

5.3.5 Model validation

When comparing the results obtained from the model with those obtained from the current verified timetable in Addendum B, as expected the improved model yields a lower elapsed time for trains on the NatCor. This is because the model travel times exclude time that should be scheduled for braking, acceleration, deceleration at temporary speed restriction points and stopping at signals, as well as excluding stations where drivers are changed for book-off. Figure 22 in section 3.4.3 recorded 11 slots lost to temporary speed restrictions. This equates to 11*20 = 220 minutes (3.67 hours) added delay not accounted for in the model.

The accounting for the delays caused by sharing part of the line with commuter trains run by PRASA was also not considered. The Transnet LTPF (2019) states that passenger trains move faster than freight trains and, because of this, it is estimated that a passenger train uses 1.3 times freight train operational slots when sharing the network. The number of PRASA trains that run on the NatCor vary between peak and off-peak times.

The Transnet Scheduling as well as the Control and Signalling departments were consulted and it was found that there is a total of 26 signal points on the line which help control headway times. There are six main signals placed at the end of the sections identified in Table 3, where trains may be required to stop. Three to five minutes stopping time is allocated for each train at these signal points. The average braking and acceleration time for trains on the NatCor is 3.5 minutes. This means that for the trains currently running, an average of 7.5 minutes (0.125 hours) must be added for every end of section signal the train passes.

With all the above taken into consideration the new estimated travel time matrix for the model is illustrated in Table 8.

Table 8: A matrix of new estimated travel times.

	Rietvallei	**Perdekop**	**Newcastle**	**Glencoe**	**Danskraal**	**Marianhill**	**Durban**
Rietvallei	M	3.91	6.17	7.66	9.07	15.21	15.62
Perdekop	3.91	M	2.26	3.75	5.29	11.42	11.84
Newcastle	6.17	2.26	M	1.485	2.90	9.29	9.71
Glencoe	7.66	3.75	1.485	M	1.415	7.92	8.34
Danskraal	9.07	5.29	2.90	1.415	M	6.64	7.06
Marianhill	15.21	11.42	9.29	7.92	6.64	M	0.42
Durban	15.62	11.84	9.71	8.34	7.06	0.42	M

The estimated maximum total elapsed time (ET_{ij}) using Table 8 yields a value of 571.9867. This new value is a better estimate of the objective value and differs to the real-time value by 4.25 hours. When the capacity slots lost to temporary speed restrictions is added to this then the final ET_{ij} is 571.9867 + 3.67 = 575. 6567 hours. The model provides a better estimate of the total elapsed time for trains on the NatCor and from the results it is evident that the model does what it is intended to do and behaves as intended when comparing it to the results obtained by Akgül (2020). A comparison of the timetables and the improved model with the times added is given illustrated in Figure 30.

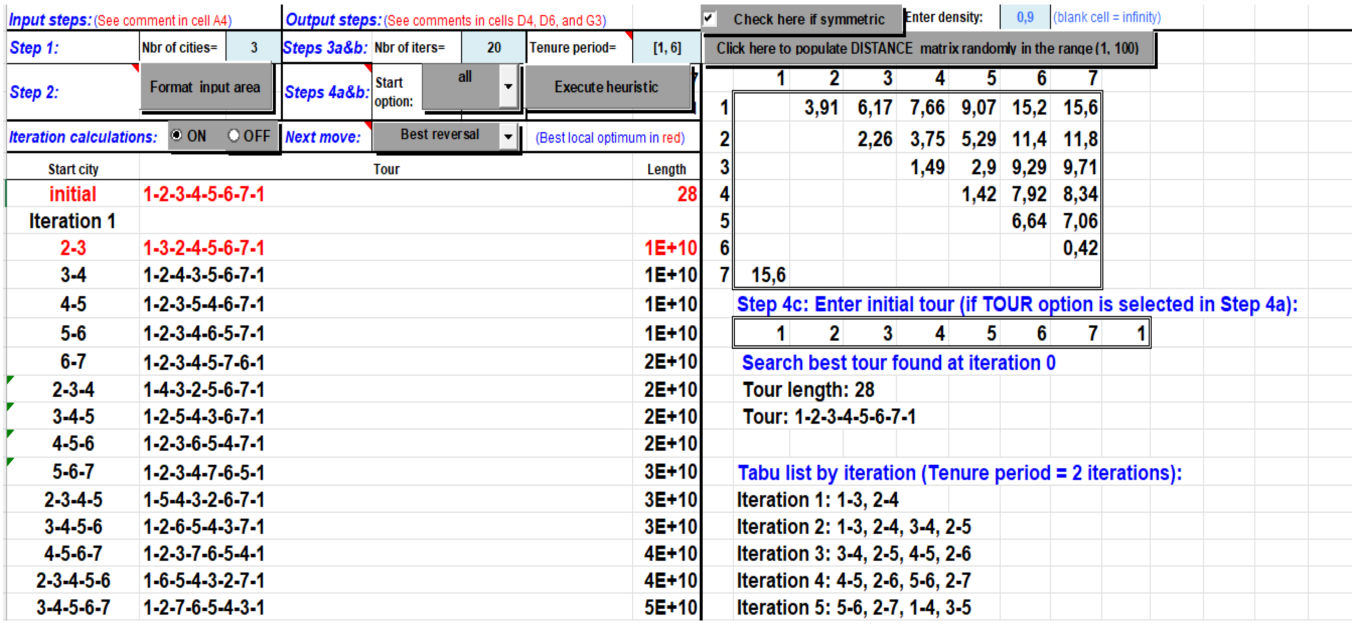
5.4 Results using the Tabu Search metaheuristic

Section 5.3 provided an initial feasible solution to the MILP model. In this section the TS is used to improve the initial feasible solution. The TS is an algorithm usually used to calculate a minimum distance that can be travelled between cities. In this research the algorithm was implemented using a TS algorithm for the Traveling Salesman Problem programmed on MS Excel and it calculates the minimum estimated travel time for all the different routes traversed by the trains on the NatCor. A step-by-step guide of how the algorithm is executed is shown in Figure 31.

1. **Starting tour.** This is a specific starting city for a tour constructed by the nearest-neighbor heuristic
2. **Subtour reversal.** Two added tour legs replace two deleted ones to produce a new tour
3. **Neighborhood at iteration *i*.** All tours (including infeasible ones with infinite length) generated by applying subtour reversals to tour *i*.
4. **Tabu move.** A reversal tour is tabu if *both* of its deleted legs are on the tabu list.
5. **Next move at iteration *i*.** Identify the shortest tour in neighborhood *i* and select it as the next move if it is non-tabu, or if it is tabu but yields a better solution. Else, exclude the shortest (tabu) tour and repeat the test with the next shortest neighborhood tour.
6. **tabu tenure period T at iteration *i*.** The tenure period is the (in this case random) number of successive iterations a tabu element stays on the tabu list.
7. **Changes in tabu list at iteration *i*.** Reversal legs defining tour *i* from tour *i* − 1 are added to the list. Tour legs completing tenure (those that entered the list at iteration *i* - t + 1) are deleted from the list.

Figure 31: TS algorithm steps

The algorithm is initialised by setting the different parameters and entering the relevant path as the initial tour. The travel time matrix in Table 8 was used as the "distance matrix" with the big M values replaced by blank spaces. The algorithm calculates the length of the tours using the Nearest Neighbour Heuristic (NNH) and searches for a better solution after 20 iterations. A screen shot of the first iterations of algorithm for the Rietvallei to Durban path is provided in Figure 32.


Input steps: (See comment in cell A4)
Output steps: (See comments in cells D4, D6, and G3)
Check here if symmetric
Enter density: 0,9 (blank cell = infinity)
Step 1: Nbr of cities= 3
Steps 3a&b: Nbr of iters= 20 Tenure period= [1, 6]
Click here to populate DISTANCE matrix randomly in the range (1, 100)
Step 2: Format input area
Steps 4a&b: Start option: all
Execute heuristic
Iteration calculations: ON OFF
Next move: Best reversal (Best local optimum in red)

Start city	Tour	Length
initial	1-2-3-4-5-6-7-1	28
Iteration 1		
2-3	1-3-2-4-5-6-7-1	1E+10
3-4	1-2-4-3-5-6-7-1	1E+10
4-5	1-2-3-5-4-6-7-1	1E+10
5-6	1-2-3-4-6-5-7-1	1E+10
6-7	1-2-3-4-5-7-6-1	2E+10
2-3-4	1-4-3-2-5-6-7-1	2E+10
3-4-5	1-2-5-4-3-6-7-1	2E+10
4-5-6	1-2-3-6-5-4-7-1	2E+10
5-6-7	1-2-3-4-7-6-5-1	3E+10
2-3-4-5	1-5-4-3-2-6-7-1	3E+10
3-4-5-6	1-2-6-5-4-3-7-1	3E+10
4-5-6-7	1-2-3-7-6-5-4-1	4E+10
2-3-4-5-6	1-6-5-4-3-2-7-1	4E+10
3-4-5-6-7	1-2-7-6-5-4-3-1	5E+10

	1	2	3	4	5	6	7
1		3,91	6,17	7,66	9,07	15,2	15,6
2			2,26	3,75	5,29	11,4	11,8
3				1,49	2,9	9,29	9,71
4					1,42	7,92	8,34
5						6,64	7,06
6							0,42
7	15,6						

Step 4c: Enter initial tour (if TOUR option is selected in Step 4a):

1	2	3	4	5	6	7	1

Search best tour found at iteration 0
Tour length: 28
Tour: 1-2-3-4-5-6-7-1

Tabu list by iteration (Tenure period = 2 iterations):
Iteration 1: 1-3, 2-4
Iteration 2: 1-3, 2-4, 3-4, 2-5
Iteration 3: 3-4, 2-5, 4-5, 2-6
Iteration 4: 4-5, 2-6, 5-6, 2-7
Iteration 5: 5-6, 2-7, 1-4, 3-5

Figure 32: Iterations for the Rietvallei to Durban path

It is important to note that the calculation of the path includes an extra 15.62 hours that was added to close off the tour. With this extra travel time removed the total average travel time from Rietvallei to Durban is calculated as :

$$28 - 15.62 = 12.38 \text{ hours} \tag{35}$$

This is a 3.24 hour travel time difference improvement.

The minimum travel time for trains travelling from Rietvallei to Newcastle was calculated only using the NNH because the TS requires the path to have more than 3 cities visited. The travel time matrix for this path is found in Table 9. The results from the NNH were 6.17 hours.

Table 9: Travel time matrix for the path from Rietvallei to Newcastle

City	Rietvallei	Perdekop	Newcastle
Rietvallei		3.91	6.17
Perdekop			2.26

Newcastle	6.17		

Next the minimum travel time for the Newcastle to Durban path was calculated using the matrix in Table 10.

Table 10: Travel time matrix for the path from Newcastle to Durban

City	Newcastle	Glencoe	Danskraal	Marianhill	Durban
Newcastle		1.485	2.90	9.29	9.71
Glencoe			1.415	7.92	8.34
Danskraal				6.64	7.06
Marianhill					0.42
Durban	9.71				

A screen shot of the calculation for the path on MS Excel is shown in Figure 33

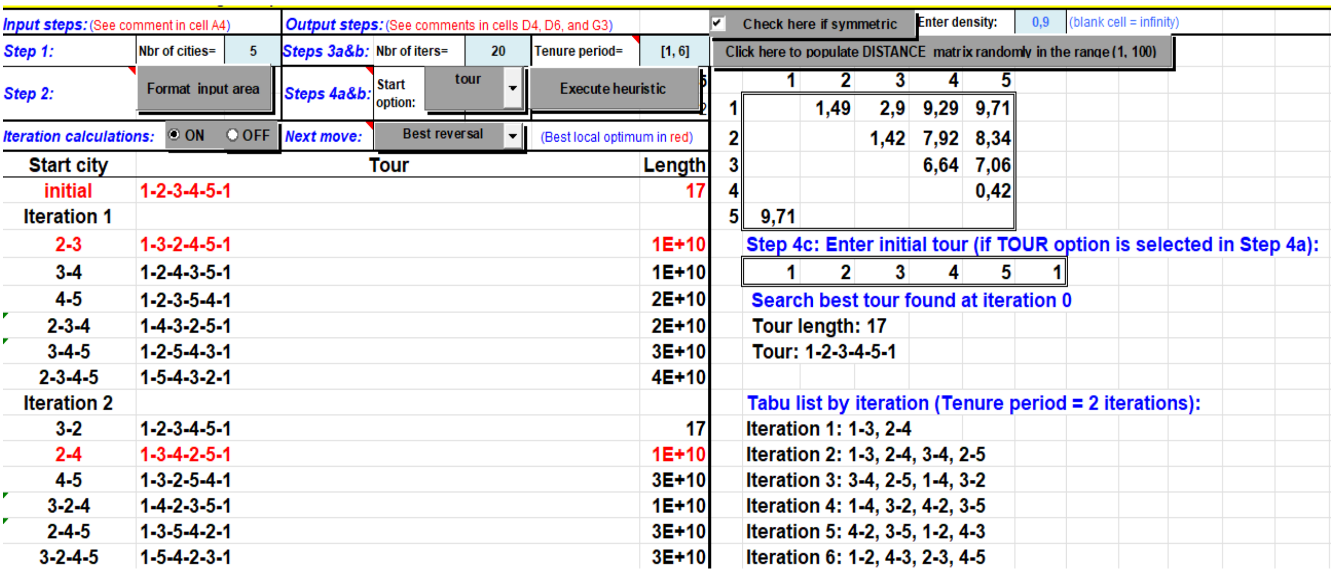


Figure 33: Iterations for the Newcastle to Durban path

The travel time calculated was 17 hours which includes a tour from Durban to Newcastle to close of the path for the algorithm. Removing this tour resulted in a minimum travel time of 7.29 hours, which is a 2.42 hour improvement.

The final path traversed by trains on the Natcor runs from Danskraal to Durban. The travel time matrix for the path is given in Table 11.

Table 11: Travel time matrix for the path from Danskraal to Durban

City	Danskraal	Marianhill	Durban
Danskraal		6.64	7.06
Marianhill			0.42
Durban	7.06		

Similar to the calculation for the minimum travel time from Rietvallei to Newcastle, the minimum travel time for trains travelling from Danskraal to Durban was calculated only using the NNH because the TS requires the path to have more than 3 cities visited. The resulting minimum travel time was 7.06 hours which is the same as the results in section 5.3.5

To obtain the final solution from the TS it is important to recognize that there is more than one train traversing each of the identified paths. Thus, to calculate the total travel time for all the trains, the total number of trains traversing each path need to be summed up and multiplied by the travel time corresponding to the particular path. There are 22 trains scheduled to travel from Rietvallei to Durban, 7 trains scheduled to travel from Rietvallei to Newcastle, 4 trains scheduled to travel from Newcastle to Durban and 5 trains schedules to travel from Danskraal to Durban. The resulting the minimum travel time for all the trains is:

$$27 * 12.38 + 7 * 6.17 + 4 * 7.29 + 5 * 7.06 = 441.91 \text{ hours} \tag{35}$$

This result is a further improvement of the initial solution found in section 5.3.5. The result is a 134.3137 hours improvement on the results from the clean timetable and a final comparison is shown in Figure 34.

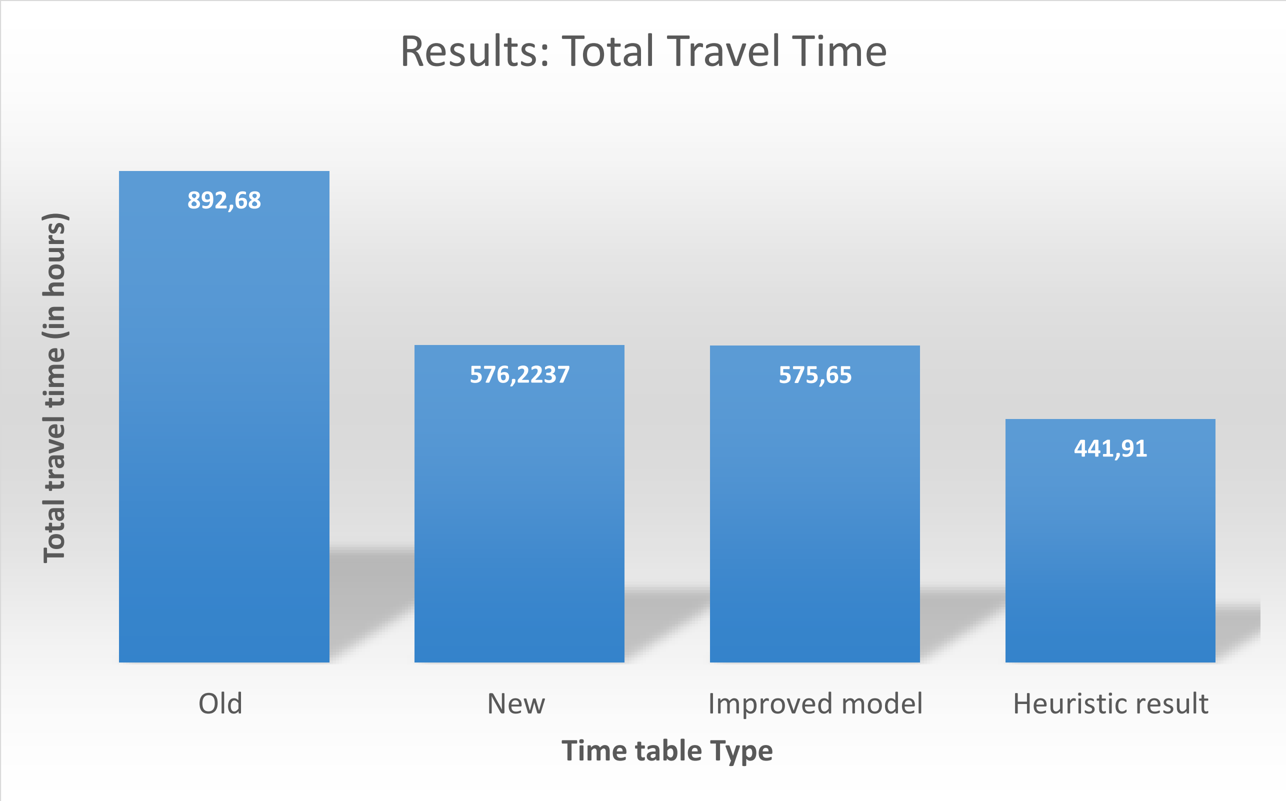


Figure 34: A comparison of all results obtained for the minimum total travel time

5.5 Results interpretation

From the results it is evident that the reason the company has been unable to unlock more capacity slots on the NatCor is because of the lack of updates made to the schedules and timetables over the past 10 years. The company had deemed it almost impossible to meet the demands forecast in the LTPF for the next 5-35 years and had investigated how adding more tracks on the line would improve the capacity and when the improvement could be implemented. Table 12 shows that the improvement in the total travel time for the trains achieved by cleaning up the current schedule resulted in a 35.45% reduction in the total travel time. This opened up multiple capacity slots taken up by trains no longer running on the line as well as slots taken up by duplicated trains.

Table 12: Total travel time improvement

Timetable/model type	Minimum travel time in hours	Improvement in hours	Improvement percentage
Old timetable	892.680	-	-
Clean timetable	576.224	316.456	35.45%
Improved model	575.560	317.030	35.51%
TS model	441.910	450.77	50.50%

Further improvement was achieved using the VRP model and the resulting travel times for each path were used as an initial feasible solution to the TS. The algorithm further improved the travel times and reduced the delays incurred by the trains. The TS result was a 50.50% improvement on the current congested timetable, reducing the total travel time by a total 450.77 hours.

A further reduction to the travel time can be achieved through recovery of the capacity slots lost to sharing part of the corridor with PRASA passenger commuter trains. These trains take up a significant amount of capacity especially during peak travel times. The company has started investigating ways to recover the slots with the main ones being constructing a new track and removing the freight trains from that track and negotiating with PRASA to use the track exclusively.

From the results it is evident that the model has fulfilled what the research was trying to achieve. The capacity was optimised through minimising the delays incurred as trains travel on the NatCor.

5.6 The theoretical effect of the improved timetable on the DCT

The DCT currently experiences long train and truck dwell times. The dwell times are estimated to be five days long. A contribution to these long dwell times is the lack of available infrastructure capacity at the DCT.

Pier 1 at the DCT currently has three tracks, which can each accommodate trains up to 50 wagons long. These trains are currently being loaded and unloaded with one crane. This creates a turnaround time of 5 hours and 4.8 trains are loaded and unloaded per day.

Pier 2 has six tracks, which also accommodate trains up to 50 wagons long per track. Three of the tracks at Pier 2 are used for unloading, with three cranes unloading the trains. The other three tracks are used for loading the trains and this is done using three reach stackers. The turnaround time for trains at Pier 2 is also estimated at 5 hours and only 4.8 trains are loaded and unloaded at Pier 2.

There are 43 trains scheduled to run on the NatCor line and 70% end up at the DCT. The DCT has no space or infrastructure to accommodate these trains, which causes congestion at the yard and leads to delays and long dwell times.

Improving on the capacity of the NatCor by minimising the delays in total travel times for the trains may result in running longer trains. This means that the non-optimal operations currently experienced at the DCT will be affected even more negatively. The DCT currently has no infrastructure to accommodate trains longer than 50 wagons for loading and unloading.

Continual improvement in the efficiency of the operations on the NatCor line and decreasing delays, and therefore adding more trains to the line, implies a much worse effect on the DCT because the DCT cannot currently handle an efficient and optimal NatCor schedule. This would force the company to invest in more infrastructure or to consider other options, such as increasing resources for DCT operations.

On the other hand, as proven in Chapter 5, the NatCor timetables are currently scheduled with large delay times. This delay leads to further delays in operations at the DCT. The effects of delays of any kind at the DCT also in turn affect the NatCor because the longer trains dwell at the DCT the longer the delays, so causing a vicious cycle of delays modelled using a reinforcing causal loop diagram as follows:

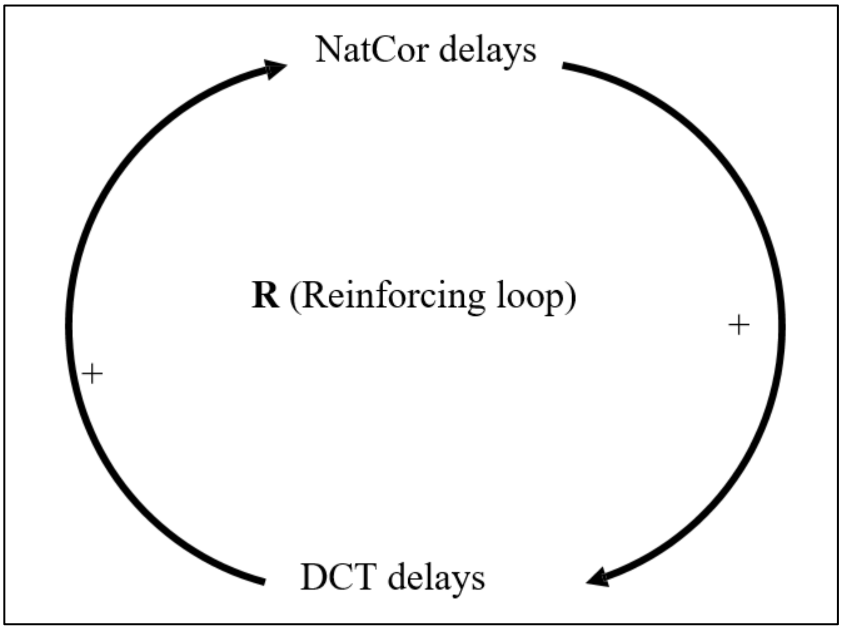


Figure 35: A causal loop diagram of the NatCor and DCT delays.

An intervention to optimise the capacity at the terminal is needed for the DCT. This can be explored using current infrastructure, like the NatCor line. The intervention can be done using the information obtained from the NatCor line optimisation results as inputs.

6. Conclusion and recommendations

6.1 Conclusion

There is currently a capacity shortage problem on the Natal Corridor, and with Transnet's annual demand increases in the commodities running on the line, the company has considered expanding existing infrastructure by possibly adding another track on the corridor. However, the company has been somewhat reluctant to implement the expansion because of the elevated costs. This research was conducted in order to optimise capacity on the line with the existing infrastructure and looked mainly at the schedules encompassing infrastructure and train types. The research followed the guidelines of the seven-step operations research framework highlighted in Chapter 3. The framework is considered a reliable method and has often been used.

The first step was to analyse the orientation of the problem. The primary objective of this step was to introduce the problem to the reader, so that they have a clear picture of the relevant issues. This step is covered in detail in Chapter 1.

The second step was to define the problem. The objective here was to refine the deliberations further, from the orientation phase to the point where there is a clear definition of the problem, scope and objectives. This step was covered in Chapter 1 under the objectives and purpose of the research. This part of the research identified what the study was trying to achieve and the parts of the NatCor line that would be analysed. Section 3.1 highlighted the research approach used in the study.

The third step was data collection. At this stage, the data was collected with the objective of translating the problem defined in the second phase into a model that can then be analysed objectively. This section was covered in Chapter 3. The process of how the data was collected was also explained in more detail. The NatCor line raw data contained a schedule, which was heavily congested, and with 67 trains running on an average delay of 45.93 minutes per train. This made it impossible for the company to consider running longer trains on the line and hence the suggestion to expand. However, from the cleaning and sorting of the data, it was found that many trains were no longer running on the corridor – but were still shown as taking up capacity on the line. These trains were removed and an updated timetable of trains currently running on the line was constructed. The new updated timetable produced a diagram showing less congestion, even before minimising the delays using the developed MILP hybrid TTP and adapted VRPTW models.

Models were then formulated in step 4 of the framework. This step also served as an achievement of objectives 1 and 2. The first objective listed in section 1.4 was to formulate the NatCor capacity problem as MILP mathematical models. The MILP formulated in this research was a TTP, which was a hybrid model formulated from using models by Liu and Han (2016) and Stanojević et al. (2010). The general TTP is a complex problem and is considered an NP-hard problem to solve as-is. The TTP was thus converted into an equivalent VRPTW to be able to employ solution methods that are not complex.

Objective 2 was to estimate the optimal capacity of the NatCor by using the branch-and-cut method through the COIN-OR branch and cut (CBC) algorithm to solve the formulated MILP models. This was achieved in step 5 of the framework, where a solution for the models is obtained. The solution method employed by this research was a technique developed by Akgül (2020). The author solved the VRPTW with homogeneous fleet by programming the model on Excel Open Solver using a CBC method. It is important to note that an adaptation of the model was used because some of the constraints from the model were not relevant for the converted TTP. A new schedule was constructed using the travel times in Table 6, while keeping the scheduled departure times the same. The new schedule diagram was less congested than the diagram in Figure 23. To achieve objective 3, the results obtained using the VRPTW were further improved by using the TS metaheuristic. The algorithm used the results from the VRPTW as an initial feasible solution. The initial feasible solution was improved in iteration zero of the algorithm where the NNH was used to calculate the minimum length, in hours, of the different paths traversed by trains on the line.

In step 6, the model was validated and an analysis was done on the output. The first part of this analysis was to verify that the solution itself makes sense. The feasible solution followed the same trend as the model developed by Akgül (2020), which validated the model. The validation process also included adding back delays that were not part of the original model. The results of the improved model estimated the sum of average travel times was a minimum total travel time of 575.65 hours for the trains. The results were close to the real-time results calculated using the clean timetable. This almost similar results validated that the correct model was implemented.

The results from the clean timetable were higher than the improved model results. This could be part of the cause of current insufficient capacity being experienced on the NatCor line. Reducing the travel time intervals to the improved times could possibly free up some capacity, making it more possible to either increase headway time and run longer trains or add more trains of the current length to the line.

Objective 4 was to compare the results from the model and the clean timetable to results obtained using past timetable. This was achieved in section 5.3.3 by using the old timetable in Addendum A as an input to the VRPTW and solving it to obtain the minimum estimated travel time. The results from the old timetable were compared to results obtained from the clean timetable and from the ideal model. The old timetable that has been used by the company for the past ten years yielded a result of 892.69 hours as the minimum travel time for the trains on the schedule. This was significantly higher than the other three results as evident in Figure 36

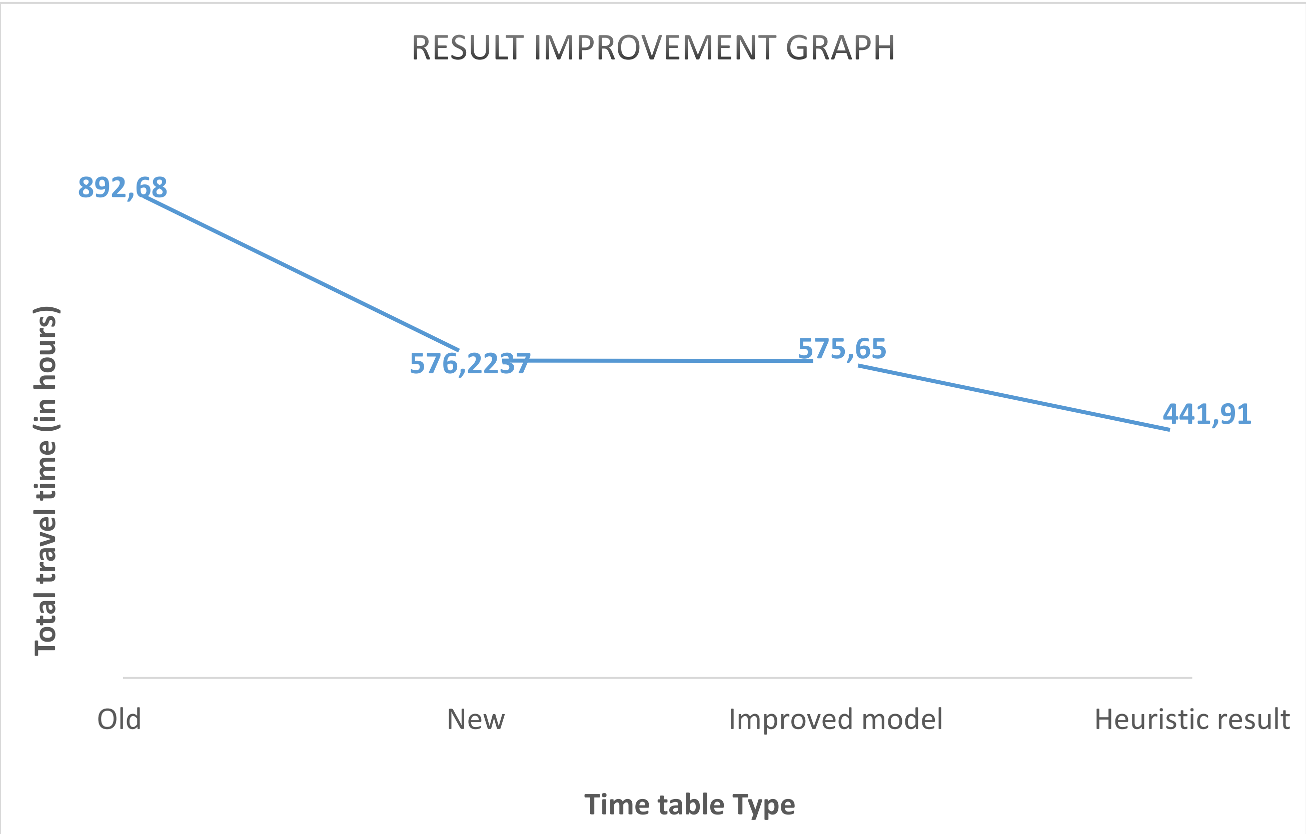


Figure 36: A graph summarising the final results

A comparison of Table 4 and Table 6 showed that the longest delay difference on the line is experienced between Glencoe and Marianhill. This is exacerbated by the PRASA-operated passenger trains, which join the NatCor line at Cato Ridge. The company is currently exploring the possibility of constructing a new track in that region to recover slots lost from sharing part of the line with PRASA.

The final step of the OR framework is the implementation and monitoring phase. This will be completed by the company should they wish to use the case study. This can be done by considering the effects the optimisation of the NatCor schedules will have on the DCT.

In section 5.6 it was shown that with the above results it can be expected that the DCT will not be able to handle any capacity increases. This section also shows how objective 5 was achieved. Objective 5 was to determine the theoretical effects that the results can have on the DCT to assist in recommending future research that can be done on the NatCor line and the DCT. In section 5.6 it was determined that as it stands, the DCT is lacking on infrastructure and the infrastructure that it currently has needs to be updated. Running more or longer trains will only add to congestion currently happening at the yard, which could lead to further delays on the Corridor, so causing a long cycle of delays.

The company has investigated expanding the infrastructure at Pier 1 and Pier 2. Other options have been to build a dry port at another location not too far from the connecting rail network. This would seem to be a more optimal approach than the current situation at the DCT.

6.2 Recommendations

This research has created an opportunity for future research into how operations at the DCT can be improved simultaneously with those of the NatCor. This would mean that the negative results from the NatCor capacity optimisation could be managed before spiralling out of control. In doing this, the delays in shunting, loading and offloading at the DCT could be managed and train turnaround times could be monitored and improved.

The models formulated and used in this research can further be expanded into the networks and not just focus on a single line. This would help the company reconcile most of its operations on its rail network, because optimisation of one line will affect others where the lines connect at certain points. Optimising multiple lines and corridors simultaneously will help save the company time and money.

Research can also be further developed to create timetables that are more robust. From the literature, it can be shown that developing a more robust timetable ensures that even with delays there will be no knock-on effect and trains that are not delayed can operate as normal without incurring any delays.

7. References

Abbas-Turki, A., Zaremba, E., Grunder, O. and El-moudni, A. (2011) ‘Perfect homogeneous rail traffic: A quick efficient genetic algorithm for high frequency train timetabling.’ *14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. Washington DC, USA, 5-7 October. New York: IEEE, pp. 1495-1500.

Abdullah, S., Burke, E.K. and McCollum, B., 2005, July. ‘An investigation of variable neighbourhood search for university course timetabling.’ In *The 2nd Multidisciplinary Conference on Scheduling: Theory and Applications, NY, USA* (pp. 413-427).

Abril, M., Barber, F., Ingolotti, L., Salido, M.A., Tormos, P. and Lova, A. (2008). ‘An assessment of railway capacity.’ *Transportation Research Part E: Logistics and Transportation Review*, *44*(5), pp. 774-806. Available at: https://www.sciencedirect.com/science/article/abs/pii/S1366554507000579 (Accessed: August 2019)

Akgül, A (2020) *Vehicle Routing Problem with Spreadsheet- optimisation of Vehicle Routing Problem variants*. Available at: https://www.udemy.com/course/vehicle-routing-problem-with-spreadsheet/ (Accessed: December 2021).

Andersson, E., Peterson, A. and Törnquist Krasemann, J. (2015) ‘Improved railway timetable robustness for reduced traffic delays–a MILP approach.’ *6th international conference on railway operations modelling and analysis-rail.* Tokyo, Japan, 23-26 March. Available at: https://www.diva-portal.org/smash/record.jsf?pid=diva2%3A799362&dswid=-2471 (Accessed: 21 May 2020)

Arenas, D., Chevrier, R., Hanafi, S. and Rodriguez, J. (2015) ‘Solving the Train Timetabling Problem, a mathematical model and a genetic algorithm solution approach.’ In 6th international conference on railway operations modelling and analysis (RailTokyo2015).

Bard, J.F., Huang, L., Dror, M. and Jaillet, P., 1998. A branch and cut algorithm for the VRP with satellite facilities. *IIE transactions*, *30*(9), pp.821-834.

Bellman, R. and Kalaba, R., (1957) ‘On the role of dynamic programming in statistical communication theory.’ *IRE Transactions on Information Theory*, *3*(3), pp.197-203.doi: 10.1109/TIT.1957.1057416. (Accessed: 31 January 2022)

Belottti, P. Kinches, C. and Mahajan, A. (2012). *Mixed-Integer Nonlinear Optimization*. Argonne, Illinois, Springer.

Brännlund, U., Lindberg, P.O., Nou, A. and Nilsson, J.E. (1998) ‘Railway timetabling using Lagrangian relaxation.’ *Transportation science*, *32*(4), pp.358-369.doi: https://doi.org/10.1287/trsc.32.4.358 (Accessed: 15 January 2020)

Brucker, P. (1979) ‘NP-Complete operations research problems and approximation algorithms.’ *Zeitschrift für Operations Research,* 23, pp.73–94.doi: https://doi.org/10.1007/BF01951543 (Accessed: 20 March 2020)

Burdett, R. and Kozan, E. (2006). Techniques for absolute capacity determination in railways. Transportation Research Part B: Methodological, 40(8), pp. 616-632. Available at https://eprints.qut.edu.au/10385/ (Accessed: 23 January 2020)

Burdett, R.L. (2016). Optimisation models for expanding a railway's theoretical capacity. *European Journal of Operational Research*, *251*(3), pp.783-797.doi: https://doi.org/10.1016/j.ejor.2015.12.033 (Accessed: 23 January 2021)

Bussieck, M.R. and Pruessner, A. (2003) ‘Mixed-integer nonlinear programming.’ *SIAG/OPT Newsletter: Views & News*, 14(1), pp.19-22. Available at: https://www.mat.univie.ac.at/~neum/glopt/mss/BusP03.pdf (Accessed: 22 January 2022)

Cacchiani, V. and Toth, P. (2011) ‘Nominal and robust train timetabling problems.’ *European Journal of Operational Research*, *219*(3), pp.727-737.doi: https://doi.org/10.1016/j.ejor.2011.11.003 (Accessed: 15 January 2022)

Caprara, A., Fischetti, M. and Toth, P. (2002) ‘Modeling and solving the train timetabling problem.’ *Operations research*, *50*(5), pp.851-861.doi: https://doi.org/10.1287/opre.50.5.851.362 (Accessed: 23 January 2022)

Carr, J. (2014).’ An introduction to genetic algorithms.’ *Senior Project*, 1(40), pp.7. Available from:

https://www.whitman.edu/documents/Academics/Mathematics/2014/carrjk.pdf (Accessed: 31 January 2022)

Chinneck, J. W. (2015).' Practical optimization: a gentle introduction.' *Systems and Computer Engineering'* Carleton University. Ottawa. Available from: http://www.sce.carleton.ca/faculty/chinneck/po. html, pp 11. (Accessed: 16 Jan 2022)

Christofides, N. (1976) 'The vehicle routing problem' *Revue française d'automatique, informatique, recherche opérationnelle*. Operations Research, 10(6), pp.55-70. Available from: https://www.rairo-ro.org/articles/ro/pdf/1976/01/ro197610V100551.pdf (Accessed: 31 January 2022)

Corana, A., Marchesi, M., Martini, C. and Ridella, S. (1987) 'Minimizing multimodal functions of continuous variables with the "simulated annealing" algorithm' *ACM Transactions on Mathematical Software (TOMS)*, *13*(3), pp.262-280.doi: https://dl.acm.org/doi/abs/10.1145/29380.29864 (Accessed: 31 January 2022)

Dai, H., Khalil, E., Zhang, Y., Dilkina, B., and Song, L. (2017). 'Learning combinatorial optimization algorithms over graphs'. *Advances in neural information processing systems 30*. Long Beach, CA. USA. 4-9 December. New York, Curran Associates Inc.

Dale, S. (2015) 'Heuristics and biases: The science of decision-making.' *Business Information Review*, 32(2), pp.93-99.doi: https://doi.org/10.1177/0266382115592536 (Accessed: 20 February 2021)

Dantzig G.B and Thapa, M.N, (1997). *Linear Programming*, New York, Springer

Deb, K., (1999) 'Solving goal programming problems using multi-objective genetic algorithms'. In *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99.*Washington DC. USA. 6-9 July. New York: IEEE. Vol. 1, pp. 77-84

Desrochers, M., Desrosiers, J., & Solomon, M. (1991). A New Optimization Algorithm for the Vehicle Routing Problem with Time Windows. *Operations Research*, *40*(2), 342–354. Available at: http://www.jstor.org/stable/171457 (Accessed: 4 February 2022)

Dhoruri, A.I., Sari, E.R. and Lestari, D. (2013) 'Solving Capacitated Vehicle Routing Problems with Time Windows by Goal Programming Approach' *IndoMS International*

Conference on Mathemathics and Its Applications. Sleman. 6-7 November.Indonesia: Indonesia Mathematical Society. pp. 155-162

Dick, C.T. and Mussanov, D. (2016) Operational schedule flexibility and infrastructure investment: capacity trade-off on single-track railways. *Transportation Research Record*, *2546*(1), pp.1-8.doi: https://doi.org/10.3141/2546-01 (Accessed: 20 February 2022)

Dingler, M. H., Lai, Y. C., & Barkan, C. P. (2009). Impact of train type heterogeneity on single-track railway capacity. *Transportation Research Record*, *2117*(1), 41-49.doi: https://doi.org/10.3141/2117-06 (Accessed: 22 February 2022)

Eck, R.D. (1979). QM an Introduction to Quantitative Methods for Business Application. Cincinnati. O.H. South Western Publishing House.

Eddy, S.R. (2004) 'What is dynamic programming?' *Nature biotechnology*, *22*(7), pp.909-910.doi: https://doi.org/10.1038/nbt0704-909 (Accessed: 20 February 2022)

Elf, M., Gutwenger, C., Jünger, M., & Rinaldi, G. (2001). 'Branch-and-cut algorithms for combinatorial optimization and their implementation in ABACUS.' *Computational Combinatorial Optimization*. 2241 (1). pp. 157-222.doi: 10.1007/3-540-45586-8_5. (Accessed: 22 February 2022)

Fernandes, F. P., Costa, M. F. P., & Fernandes, E. M. (2009) 'Overview on mixed integer nonlinear programming problems.' *AIP Conference Proceedings*.1168 (1), pp. 1374-1377.doi: https://aip.scitation.org/doi/abs/10.1063/1.3241342 (Accessed: 20 February 2022)

Fleszar, K., Osman, I.H. and Hindi, K.S. (2009) 'A variable neighbourhood search algorithm for the open vehicle routing problem'. *European Journal of Operational Research*, *195*(3), pp.803-809.doi: https://doi.org/10.1016/j.ejor.2007.06.064 (Accessed: 20 February 2022)

Forrest, S. (1996) 'Genetic algorithms.' *ACM Computing Surveys (CSUR)*, *28*(1), pp.77-80.

Furini, F. and Kidd, M.P. (2013) 'A fast heuristic approach for train timetabling in a railway node.' *Electronic Notes in Discrete Mathematics*, *41*, pp.205-212.doi: https://doi.org/10.1016/j.endm.2013.05.094 (Accessed: 22 February 2022)

Garrisi G and Pastor, C. (2019). 'Train-Scheduling Optimization Model for Railway Networks with Multiplatform Stations,' *Sustainable High Volume Road and Rail Transport in Developed and Developing Countries.* Sustainability 2020, 12(1), 257; available from www.doi.org/10.3390/su12010257. (Accessed: 01 January 2022)

Gendreau, M. (2002) An introduction to tabu search. *Handbook of metaheuristics*.pp. 37-54. Springer, Boston, MA.

Giegerich, R., Meyer, C. and Steffen, P. (2004) 'A discipline of dynamic programming over sequence data.' *Science of Computer Programming*, *51*(3), pp.215-263.doi: https://doi.org/10.1016/j.scico.2003.12.005 (Accessed: 23 February 2022)

Gigerenzer, G. and Gaissmaier, W. (2011) 'Heuristic decision making.' *Annual review of psychology*, 62, pp.451-482.doi: https://doi.org/10.1146/annurev-psych-120709-145346 (Accessed: 21 February 2022)

Glover, F. and Laguna, M. (1998) 'Tabu search' *Handbook of combinatorial optimization* (pp. 2093-2229). Boston, MA. Springer.

Hansen, I.A.(2004) 'Increase of capacity through optimized timetabling'. *WIT Transactions on The Built Environment*, 74(1).pp.10.doi:10.2495/CR040541 (Accessed: 22 February 2022)

Hansen, I.A. (2010) 'Railway network timetabling and dynamic traffic management.' *International Journal of Civil Engineering*, 8 (1): 19-32. Available at: http://ijce.iust.ac.ir/files/site1/user_files_6k93w6/ijce-A-10-3-244-152f23a.pdf (Accessed: 20 February 2022)

Hansen, P., Mladenović, N. and Pérez, J.A.M.(2010) 'Variable neighbourhood search: methods and applications' *Annals of Operations Research*, *175*(1), pp.367-407.doi: https://doi.org/10.1007/s10479-009-0657-6 (Accessed: 21 February 2022)

He, H., Daume III, H. and Eisner, J.M.(2014) 'Learning to search in branch and bound algorithms.' *Advances in neural information processing systems*, *27*, pp.3293-3301.doi: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.701.4574&rep=rep1&type=pdf (Accessed: 21 February 2022)

Henderson, D., Jacobson, S. H., and Johnson, A. W. (2003). The theory and practice of simulated annealing. Handbook of metaheuristics (pp. 287-319). Boston, MA. Springer

Higgins, A., Kozan, E. and Ferreira, L.(1997) Heuristic techniques for single line train scheduling. *Journal of heuristics*, *3*(1), pp.43-62.doi: https://doi.org/10.1023/A:1009672832658 (Accessed: 21 February 2022)

Hillier, F.S and Lieberman, G.J. (2010) Introduction to Operations Research. 9th edn. New York. N.Y. McGraw Hill

Iyengar, J. and Srivatsava, M. (2000) 'An artificial intelligence approach to goal programming goal structure development using AHP'. *Journal of International Information management*, Vol. 9(1), p.3. Available at: https://scholarworks.lib.csusb.edu/jiim/vol9/iss1/3/ (Accessed: 19 February 2022)

Jolai, F. and Aghdaghi, M .(2008) 'A goal programming model for single vehicle routing problem with multiple routes' *Journal of industrial and system engineering*, 2(2), pp 154-163.doi: 20.1001.1.17358272.2008.2.2.6.1 (Accessed: 20 January 2022)

Kallehauge, B. (2006). *On the vehicle routing problem with time windows*. Doctoral Thesis. Technical University of Denmark.

Khadem Sameni, M. (2012) *Railway track capacity: Measuring and managing*. Doctoral thesis, University of Southampton.

Kokash, N. (2005). An introduction to heuristic algorithms. Department of Informatics and Telecommunications. University of Trento. pp.1-8.

Landex, A. (2008) 'Evaluation of railway networks with single track operation using the UIC 406 capacity method.' *Networks and Spatial Economics*, 9(1): 7-23.doi: https://doi.org/10.1007/s11067-008-9090-7 (Accessed: 19 February 2022)

Lee, Y. and Chen, C.Y.(2009) 'Modeling and solving the train pathing problem. *Twelfth World Multi-Conference on Systemics, Cybernetics and Informatics*, Orlando, USA. 29 June-2 July. Florida.

Lindfieldt, O (2011) 'An analysis of double-track railway line capacity' *Transportation Planning and Technology*, 34(4): p.301-322.doi: https://doi.org/10.1080/03081060.2011.577150 (Accessed: 22 February 2022)

Lindfeldt, A., 2015. *Railway capacity analysis: Methods for simulation and evaluation of timetables, delays and infrastructure*. Doctoral thesis. KTH Royal Institute of Technology.

Liu, P. and Han, B (2016) 'Optimizing the train timetable with consideration of different kinds of headway time.' *Journal of Algorithms and Computational Technology,* 11(2). pp.148-16.doi: https://doi.org/10.1177/1748301816689685 (Accessed: 18 February 2022)

Luethi, M., Nash, A., Weidmann, U., Laube, F., & Wuest, R. (2007). Increasing railway capacity and reliability through integrated real-time rescheduling. *11th World Conference on Transport Research. Institute for Transport Planning and System*s. Berkeley, USA. 24-28 June. ETH Zurich. Institute for Transport Planning and Systems

Martinich, J.S. (1997). Production and Operations ManagementL An Applied Modern Approach. Hoboken, NJ. John Wiley & Sons

Mason, A. J. (2012). Opensolver - an open source add-in to solve linear and integer programmes in excel. In Operations research proceedings 2011 (pp. 401-406). Springer, Berlin, Heidelberg

McKinney, W. (2012) Python for data analysis: Data wrangling with Pandas, NumPy, and IPython. Newton, MA. O'Reilly Media, Inc.

Mitchell, J.E. (2002) Branch-and-cut algorithms for combinatorial optimization problems. *Handbook of applied optimization*, *1*, pp.65-77. Available at: http://eaton.math.rpi.edu/faculty/Mitchell/papers/bc_hao.pdf (Accessed: 18 February 2022)

Mitten, L.G., (1970) 'Branch-and-bound methods: General formulation and properties'. *Operations Research*, *18*(1), pp.24-34.doi: https://doi.org/10.1287/opre.18.1.24 (Accessed: 16 February 2022)

Monks, T. (2015) 'Operational research as implementation science: definitions, challenges and research priorities', *Implementation Science*, 11(81), DOI 10.1186/s13012-016-0444-0 (Accessed: 18 February 2022)

Ombuki, B., Ross, B.J. and Hanshar, F.(2004)'Multi-objective genetic algorithms for vehicle routing problem with time windows'. *Applied Intelligence*, *24*(1), pp.17-30.doi: https://doi.org/10.1007/s10489-006-6926-z (Accessed: 22 February 2022)

Osman, I.H. (1993) 'Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem'. *Annals of operations research*, *41*(4), pp.421-451.doi: https://doi.org/10.1007/BF02023004 (Accessed: 18 February 2022)

Osman, I.H. and Laporte, G.(1996) 'Metaheuristics: A bibliography' *Operations Research* 63(1), 511–623.doi: https://doi.org/10.1007/BF02125421 (Accessed: 18 February 2022)

Paszkowicz, W.(2013) 'Genetic algorithms, a nature-inspired tool: a survey of applications in materials science and related fields: part II'. *Materials and Manufacturing Processes*, *28*(7), pp.708-725.doi: https://doi.org/10.1080/10426914.2012.746707 (Accessed: 20 January 2022)

Pirim, H., Eksioglu, B. and Bayraktar, E.(2008) *Tabu Search: a comparative study* (pp. 1-27). London, UK: INTECH Open Access Publisher.

Presidential Infrastructure Coordinating Commision. (2013). A Summary of the South African National Infrastructure Plan. 1 (2-5), 10-38 Available at: https://www.gov.za/sites/default/files/PICC_Final.pdf

Ramamurthy, P. (2007) *Operations Research*. 2nd edition. Kochi. New Age International Publication

Raithatha, Y. (2017). *Understanding the economic impact terrorism has on the destination decision making: Northern Irish tourists*. Doctoral thesis. Dublin Business School.

Reeb, J.E. and Leavengood, S.A. (1998) Using the simplex method to solve linear programming maximization problems. Oregan State University. Available at: https://ir.library.oregonstate.edu/concern/open_educational_resources/xw42n8256 (Accessed: 20 February 2022)

Richards, A. and How, J.(2005), 'Mixed-integer programming for control,' *The 2005 American Control Conference.*, 4(1). pp. 2676-2683 doi: 10.1109/ACC.2005.1470372 (Accessed: 23 January 2022)

Robenek, T., Jiang Hang, C. and Bierlaire, M.(2014) 'The ideal train timetabling problem'. *14th Swiss Transportation Research Conference.* Ascona, Switzerland, 14-16 May. Zurich: Transport and Mobility Laboratory EPFL

Rothlauf, F. (2011) 'Optimization Methods'. Design of Modern Heuristics. Natural Computing Series. Berlin, Heidelberg. Springer. doi: https://doi.org/10.1007/978-3-540-72962-4_3 (Accessed: 28 February 2022)

Saunders, M., Lewis, P. and Thornhill, A.(2006) Research methods. *Business Students.*4th edition. England. Pearson Education Limited.

Scholtz, C. (2017) *Durban container terminal: capacity analysis and the feasibility of a dry port*. Doctoral thesis. Stellenbosch University.

Sen, N. and Nandi, M.(2012) 'Goal programming, its application in management sectors– special attention into plantation management: a review'. *International Journal of Scientific and Research Publications*, *2*(9), pp.1-6.doi: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.375.4906&rep=rep1&type=pdf (Accessed: 20 January 2022)

Sevkli, M. and Aydin, M.E.(2006) 'A variable neighbourhood search algorithm for job shop scheduling problems.' *European Conference on Evolutionary Computation in Combinatorial Optimization.* Budapest, Hungary.10-12 April. Berlin, Heidelberg. Springer. (pp. 261-271).

Silver, E.A.(2004) 'An overview of heuristic solution methods'. *Journal of the operational research society*, *55*(9), pp.936-956.doi: https://doi.org/10.1057/palgrave.jors.2601758 (Accessed: 22 January 2022)

Singh, G., Sier, D., Ernst, A.T., Gavriliouk, O., Oyston, R., Giles, T. and Welgama, P. (2012). 'A mixed integer programming model for long term capacity: A case study from the Hunter Valley coal chain'. *European Journal of Operations Research,* 22(1), pp. 210-224.doi: https://doi.org/10.1016/j.ejor.2012.01.012 (Accessed: 22 January 2022)

Sklenar, J. (2017), 'Dynamic Programming Theory and Applications'. Malta University. Italy. Available at: https://is.muni.cz/www/jkrejci/Dynamic_Programming.pdf (Accessed: 22 January 2022)

Smith, J.C. and Taskin, Z.C.(2008) 'A tutorial guide to mixed-integer programming models and solution techniques'. *Optimization in Medicine and Biology*, pp.521-548.doi: https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.464.6182&rep=rep1&type=pdf (Accessed: 20 December 2021)

Sorensen, K., Sevaux, M., and Glover, F. (2018) 'A history of metaheuristics' to be published in *Handbook of Heuristics*, pp 1–18.Available at: https://doi.org/10.1007/978-3-319-07124-4 (Accessed: 7 January 2022)

Stanojevi´c, P., Mari´c, M., Kratica, J., Bojovi´c, N., and Milenkovi´c M. (2010). Mathematical Optimization for the Train Timetabling Problem. *Mathematica Balkanica.* Vol 24(3-4), pp 303-312. Available at: http://eudml.org/doc/11351 (Accessed: 23 February 2022)

Swietanowski, A. (1995) 'A Penalty Based Simplex Method for Linear Programming', *International Institute for Applied Systems Analysis*. Available at: http://pure.iiasa.ac.at/4587 (Accessed: 20 January 2022)

Taha, H.A. (2017) *Operations Research: An Introduction*.10th edition. Fayetteville. Peasrson

Tahernejad, S., Ralphs, T.K. and DeNegre, S.T.(2020) 'A branch-and-cut algorithm for mixed integer bilevel linear optimization problems and its implementation.' *Mathematical Programming Computation*, 12(4), pp.529-568.doi: https://doi.org/10.1007/s12532-020-00183-6 (accessed: 22 January 2022)

Tindell, K.W., Burns, A. and Wellings, A.J. (1992) 'Allocating hard real-time tasks: an NP-hard problem made easy'. *Real-Time Systems*, *4*(2), pp.145-165.doi: https://doi.org/10.1007/BF00365407 (Accessed: 23 January 2022)

Todd, M.J. (2002.) The many facets of linear programming. *Mathematical programming*, *91*(3), pp.417-436. Available from: https://www.researchgate.net/publication/2837588_The_Many_Facets_Of_Linear_Programming (Accessed: Jan 31 2022).

Toth, P. and Vigo, D. (2002). *The vehicle routing problem*. 1st edition. Philadelphia. Society for Industrial and Applied Mathematics.

Transnet SOC Ltd. (2016) *30-year Long-Term Planning Framework.* Available at: https://www.transnet.net/BusinessWithUs/LTPF%202017/LTPF%20Chapter%201%20Introduction.pdf (Accessed: 11 June 2018).

Transnet Freight Rail (2021). Annual report 2021. Available at: https://www.transnet.net/InvestorRelations/AR2021/Transnet%20Freight%20Rail.pdf

Transnet Limited (2015). Annual Report 2015 (1), 75-100.

Transnet Limited (2019). Annual NatCor Report 2019 (1)

Westerlund, T. and Pettersson, F.(1995) "An extended cutting plane method for solving convex MINLP problems," Computers and Chemical Engineering, 19 (1), pp. S131–S136. doi: 10.1016/0098-1354(95)87027-X , (Accessed: 22 February 2022)

Wolsey, L.A. (1998). *Integer Programming*. New York. Wiley

Xinchao, Z. (2011) 'Simulated annealing algorithm with adaptive neighborhood.' *Applied Soft Computing*, *11*(2), pp.1827-1836.doi: https://doi.org/10.1016/j.asoc.2010.05.029 (Accessed: 22 February 2022)

Yaghini, M., Nikoo, N. and Ahadi, H.R. (2014). 'An integer programming model for analysing impacts of different train types on railway line capacity'. *Transport* [Electronic], 29(1): 28-35. Available: https://doi.org/10.3846/16484142.2014.894398 (Accessed: 23 May 2018)

Yang, X.S. (2011) 'Metaheuristic optimization: algorithm analysis and open problems'. *International Symposium on Experimental Algorithms*. Naples, Italy. 5-7 May. Berlin, Heidelberg: Springer, pp. 21-32.

Zandin, K. and Maynard, H. (2000) *Maynard's Industrial Engineering Handbook*. 5th edn. New York, NY. McGraw Hill

Zhang, M., Wang, Y., Su, S., Tang, T. and Ning, B. (2018) 'A short turning strategy for train scheduling optimization in an urban rail transit line: the case of Beijing subway line 4.' *Journal of Advanced Transportation*. 2018(1).doi: https://doi.org/10.1155/2018/5367295 (Accessed: 22 January 20222)

Žilinskas, A. (2006) ‘Operations research: Theory and applications.’ *Technological and Economic Development of Economy*, *12*(4), pp.261-261.doi: https://doi.org/10.1080/13928619.2006.9637751 (Accessed: 20 January 2022)

Addendum A – Catalogue

Train number	Origin	Destination	Departure time	Arrival time	Commodity
2611 (Flexi) T67	GUN	BYD	00:15	19:21	Maize
8921 (Mega) T27	KAZ	KGX	00:40	19:03	Container 50 wagon
5901 (Mega) T4	ISO	FYN	01:25	20:49	Jet fuel empty 37 wagons
8981 (Flexi) T19	CDP	KGX	02:10	20:33	Container 50 wagon
6451 (Mega) T40	HZL	WST	03:00	00:59	Manganese 65 wagon
6459 (Mega) T41	HZL	COR	03:00	22:03	Manganese 65 wagon
8923 (Mega) T32	CPK	KGX	03:10	00:02	Container 50 wagon
8815 (Mega) T2	WED	WST	03:30	22:26	Coal 50 wagon
8561 (Flexi) T3	WED	MWF	03:30	21:52	Steel 50 wagon
1673 (Flexi) T33	PRZ	PNT	03:55	23:59	Conditional
8903 (Mega) T20	KAZ	KGX	04:54	23:30	Container 50 wagon
6461 (Mega) T10	Prz	WST	04:55	01:28	Chrome 37 wagons
6469/16469/26469 (Mega) T11, T12,T13	Prz	Byd	04:55	00:51	Chrome 37 wagons

26467 (Mega) T39	NEC	MWF	04:55	15:00	Manganese 50 wagon
8201/5201 (Flexi) T65/T66	GUN	BYD	04:55	22:46	Maize 32 wagon
25537 (Flexi) T60	DSL	FYN	05:20	13:01	Maize 32 wagon
8551 (Flexi) T5	BIJ	MWF	06:08	03:30	Steel 40 wagon
8501 (Flexi) T6	BIJ	MWF	06:08	03:30	Steel 40 wagon
18551 (Flexi) T7	BIJ	BYD	06:08	03:30	Steel 40 wagon
8503 (Flexi) T8	BIJ	BYD	06:08	03:30	Steel 40 wagon
5949 (Access) T9	SBG	FYN	06:16	02:34	Access 28 wagon
5519 (Access) T49	NCS	BYD	07:00	17:44	Steel 40 wagon
8997 (Flexi) T18	WAO	PNT	08:25	05:51	Motor cars 40 wagon
8905 (Mega) T21	CDP	KGX	08:39	03:02	Container 50 wagon
8931 (Flexi) T54	WKM	WST	09:06	21:56	Coal 40 wagon
8947 (Mega) T28	KAZ	KGX	10:01	04:38	Container 50 wagon
8973 (Flexi) T22	CDP	KGX	10:39	05:02	Container 50 wagon
6203 (Flexi) T53	BLG	BYD	10:46	21:16	Coal 40 wagon
1571 (Flexi) T48	STQ	BYD	10:50	06:14	Granite and General 40 wagon
6425 (Mega) T37	PRZ	NCY	11:50	22:15	Coal
8939 (Mega) T31	CPK	KGX	12:05	09:04	Container 50 wagon

6453 (Flexi) T44	BIJ	Wykom	12:11	20:25	Empties 50 wagon
5955 (Flexi) T55	GUN	BYD	12:15	05:55	General and fuel 32 wagon
58955 (Flexi) T56	GUN	MWF	12:15	05:55	General and fuel 32 wagon
55955 (Flexi) T57	GUN	MWF	12:15	05:55	General and fuel 32 wagon
45955 (Flexi) T58	GUN	KGX	12:15	06:10	General and fuel 32 wagon
6223 (Mega) T35	MTN	NCY	13:00	20:44	Lime 40 wagon
8511 (Flexi) T36	ISO	NCY	13:15	22:23	40 wagons
6401/1711(Mega)T 50, T51	TLN	BYD	13:20	01:51	Coal wagon
8813 (Flexi) T1	WED	WST	13:55	08:46	Coal 40 wagon
8983 (Mega)) T24	CDP	KGX	14:55	09:39	Container 50 wagon
8959 (Flexi) T29	CPK	KGX	15:10	11:58	Container 50 wagon
8985 (Flexi) T30	CPK	WST	15:10	12:15	Container 50 wagon
6455 (Mega) T43	YIO	NCY	15:40	18:49	Empty 105 wagons
8901 (Mega) T14	KAF	IPG	16:50	11:12	Motor cars 25 wagon
65537 (Flexi) T59	GUN	BYD	18:13	12:46	Maize 32 wagon
68937 (Flexi) T61	GUN	BYD	18:13	12:46	Maize 32 wagon

58937 (Flexi) T62	GUN	MWF	18:13	12:46	Maize 32 wagon
48937 (Flexi) T63	GUN	KGX	18:13	13:01	Maize 32 wagon
58937 (Flexi) T64	GUN	MWF	18:13	12:46	Maize 32 wagon
76009 (Mega) T49	JHB	DNP	18:40	09:10	Passenger
5997 (Flexi) T15	KAF	PNT	19:50	14:03	Conditional
8911 (Access) T23	CDP	KGX	20:00	14:23	Container 50 wagon
6201 (Mega) T48	NCS	MTV	20:30	06:36	40 wagons
6431 (Mega) T42	YIO	NCY	21:00	02:09	Ore 105 wagon
5041 (Flexi) T45	STQ	BYD	21:50	15:40	chrome 40 wagon
6467 (Mega) T38	HZL	WST	22:10	15:57	Manganese 65 wagon
1451 (Flexi) T52	NCY	MWF	22:21	09:32	Steel
8945 (Flexi) T17	WAO	PNT	22:30	20:10	Motorcars 45 wagon
8917 (Flexi)) T26	KAZ	KGX	22:30	16:53	Container 50 wagon
6275 (Access) T47	STQ	BYD	22:35	18:06	General 40 wagon
6275 (Access) T46	ISO	BYD	22:37	18:22	Sugar 40 wagon
6223 (Mega) T34	LAC	NCY	23:15	20:44	Lime 40 wagon
8917 (Flexi) T25	NAS	KGX	23:17	16:53	Container 50 wagon
8909 (Mega) T16	ROS	PNT	23:20	21:03	Motorcars 30 wagon

Addendum B – Clean timetable

Train number	Origin	Destination	Departure time	Arrival time
T32 6275	Rietvallei	Durban	00:40:00	18:05:30
T18 8921	Rietvallei	Durban	02:01:00	18:29:00
T9 8945	Rietvallei	Durban	02:25:00	19:25:30
T11 8981	Rietvallei	Durban	03:31:00	19:59:00
T8 8909	Rietvallei	Durban	04:00:00	20:28:00
T2 8815	Rietvallei	Durban	04:50:00	21:36:00
T40 5537	Danskraal	Durban	05:20:00	12:30:30
T12 8903	Rietvallei	Durban	06:15:00	22:56:00
T22 8923	Rietvallei	Durban	06:40:00	23:28:00
T27 6451	Rietvallei	Durban	07:14:00	23:52:00
T5 6461	Rietvallei	Durban	08:10:00	01:28:00
T36 5519	Newcastle	Durban	08:20:00	18:48:00
T4 5949	Rietvallei	Durban	09:30:00	03:13:30
T25 8955	Rietvallei	Newcastle	09:55:00	16:18:00
T13 8905	Rietvallei	Durban	10:00:00	02:28:00
T38 6203	Newcastle	Durban	10:46:00	21:16:00
T3 5901	Rietvallei	Durban	11:01:30	03:57:30
T19 8947	Rietvallei	Newcastle	11:22:00	21:24:00
T14 8973	Rietvallei	Durban	12:00:00	04:28:00
T42 2611	Danskraal	Durban	12:00:00	19:04:30
T29 6455	Rietvallei	Newcastle	12:25:00	18:44:00
T10 6455	Rietvallei	Durban	12:29:00	05:07:00
T33 1571	Rietvallei	Durban	13:00:00	05:58:00

T23 6453	Rietvallei	Newcastle	14:20:00	20:39:00
T30 6453	Rietvallei	Newcastle	14:20:00	20:39:00
T1 8813	Rietvallei	Durban	15:15:00	07:56:00
T21 8939	Rietvallei	Durban	15:30:00	08:30:00
T41 5201	Danskraal	Durban	15:30:00	22:31:30
T24 8511	Rietvallei	Newcastle	15:43:00	22:02:00
T16 8983	Rietvallei	Durban	16:16:00	09:04:30
T6 8901	Rietvallei	Durban	16:34:00	10:15:30
T20 8959	Rietvallei	Durban	18:30:00	11:24:00
T43 6401	Danskraal	Durban	18:30:00	01:35:00
T34 6431	Rietvallei	Durban	19:40:00	08:50:30
T28 6431	Rietvallei	Newcastle	19:45:00	02:04:00
T35 6201	Newcastle	Durban	20:30:00	06:44:00
T7 5997	Rietvallei	Durban	21:00:00	13:28:00
T15 8911	Rietvallei	Durban	21:21:00	13:49:00
T31 6467	Rietvallei	Durban	22:03:30	16:01:00
T37 1451	Newcastle	Durban	22:30:00	09:15:30
T39 5955	Danskraal	Durban	22:30:00	06:02:00
T26 5041	Rietvallei	Durban	22:48:30	16:42:00
T17 8917	Rietvallei	Durban	23:51:00	16:19:00

Addendum C – All Possible Paths on the NatCor

	Route Code	Origin	Destination	Distance (km)
0	R0	Rietvallei	Perdekop	189.00
1	R1	Rietvallei	Newcastle	296.00
2	R2	Rietvallei	Glencoe	364.00
3	R3	Rietvallei	Danskraal	428.31
4	R4	Rietvallei	Marianhill	695.31
5	R5	Rietvallei	Booth	716.06
6	R6	Perdekop	Newcastle	107.00
7	R7	Perdekop	Glencoe	175.00
8	R8	Perdekop	Danskraal	239.31
9	R9	Perdekop	Marianhill	506.31
10	R10	Perdekop	Booth	527.06
11	R11	Newcastle	Perdekop	107.00

12	R12	Newcastle	Glencoe	68.00
13	R13	Newcastle	Danskraal	132.31
14	R14	Newcastle	Marianhill	399.31
15	R15	Newcastle	Booth	420.06
16	R16	Glencoe	Perdekop	175.00
17	R17	Glencoe	Newcastle	68.00
18	R18	Glencoe	Danskraal	64.31
19	R19	Glencoe	Marianhill	331.31
20	R20	Glencoe	Booth	352.06
21	R21	Danskraal	Perdekop	239.31
22	R22	Danskraal	Newcastle	132.31
23	R23	Danskraal	Glencoe	64.31
24	R24	Danskraal	Marianhill	267.00
25	R25	Danskraal	Booth	287.75

26	R26	Marianhill	Perdekop	506.31
27	R27	Marianhill	Newcastle	399.31
28	R28	Marianhill	Glencoe	331.31
29	R29	Marianhill	Danskraal	267.00
30	R30	Marianhill	Booth	20.75

Addendum D – Train allocation to a path

Train number	Path	Origin	Destination
T1 8813	R5	Rietvallei	Durban
T2 8815	R5	Rietvallei	Durban
T3 5901	R5	Rietvallei	Durban
T4 5949	R5	Rietvallei	Durban
T5 6461	R5	Rietvallei	Durban
T6 8901	R5	Rietvallei	Durban
T7 5997	R5	Rietvallei	Durban
T8 8909	R5	Rietvallei	Durban
T9 8945	R5	Rietvallei	Durban
T10 6455	R5	Rietvallei	Durban
T11 8981	R5	Rietvallei	Durban

T12 8903	R5	Rietvallei	Durban
T13 8905	R5	Rietvallei	Durban
T14 8973	R5	Rietvallei	Durban
T15 8911	R5	Rietvallei	Durban
T16 8983	R5	Rietvallei	Durban
T17 8917	R5	Rietvallei	Durban
T18 8921	R5	Rietvallei	Durban
T19 8947	R5	Rietvallei	Newcastle
T20 8959	R5	Rietvallei	Durban
T21 8939	R5	Rietvallei	Durban
T22 8923	R5	Rietvallei	Durban
T23 6453	R1	Rietvallei	Newcastle
T24 8511	R1	Rietvallei	Newcastle
T25 8955	R1	Rietvallei	Newcastle

T26 5041	R5	Rietvallei	Durban
T27 6451	R5	Rietvallei	Durban
T28 6431	R1	Rietvallei	Newcastle
T29 6455	R1	Rietvallei	Newcastle
T30 6453	R1	Rietvallei	Newcastle
T31 6467	R5	Rietvallei	Durban
T32 6275	R5	Rietvallei	Durban
T33 1571	R5	Rietvallei	Durban
T34 6431	R5	Rietvallei	Durban
T35 6201	R15	Newcastle	Durban
T36 5519	R15	Newcastle	Durban
T37 1451	R15	Newcastle	Durban
T38 6203	R15	Newcastle	Durban
T39 5955	R25	Danskraal	Durban

T40 5537	R25	Danskraal	Durban
T41 5201	R25	Danskraal	Durban
T42 2611	R25	Danskraal	Durban
T43 6401	R25	Danskraal	Durban