Scheduling Algorithms for Routine Maintenance of Roads in Maintenance Wards of a Gravel Road Network

A.F. Burger

12243701

Thesis presented in partial fulfilment of the requirements for the degree of Master of Civil Engineering at the University of Stellenbosch

Study Leader: Dr. G.C. van Rooyen

March 2005

Opsomming

Die studie beskryf in hierdie tesis fokus op die ontwikkeling van algoritmes vir die skedulering van roetine instandhoudingswerk in instandhoudingswyke van 'n gruispadnetwerk. Dit is onderneem tydens die ontwikkeling van die "Blading Optimisation Module" van die "Gravel Management System" van die Provinsiale Regering: Wes Kaap.

Twee algoritmes is ontwikkel en afgebeeld op 'n loods-objekmodel op die rekenaar. Die algoritmes en toepassingskoppelvlak maak voorsiening vir die beperkings en veranderlikes wat geïdentifiseer is tydens onderhoude met personeel van die distrikmunisipaliteite in die Wes Kaap. Die algoritmes is getoets en geëvalueer met behulp van die loodstoepassing. Gevolgtrekkings en aanbevelings word gemaak op grond van die resultate van die evaluasie van die algoritmes.

Declaration

I, the undersigned, hereby declare that the work contained in this thesis is my own original work and has not previously in its entirety or in part been submitted at any university for a degree.

Ek, die ondergetekende, verklaar hiermee dat die werk gedoen in hierdie tesis my eie oorspronklike werk is wat nog nie voorheen gedeeltelik of volledig by enige universiteit vir 'n graad aangebied is nie.

Signature: Date:

Summary

The study reported in this thesis focuses on the development of algorithms that can be used to schedule routine maintenance work in maintenance wards of a gravel road network. This was undertaken as part of the development of the Blading Optimisation Module of the Gravel Management System of the Provincial Government: Western Cape.

Two scheduling algorithms were developed and mapped to a pilot object model on the computer. The algorithms and application interface takes account of the constraints and variables of routine maintenance that were identified through interviews with personnel of the five District Municipalities contained in the Western Cape. The algorithms are tested and evaluated using the pilot application. Based on the evaluation of the algorithms conclusions are drawn and recommendations are made.

This work is dedicated to my wife, Anelle.

"Transport is a basic necessity for sustainable social and economic development and an enabler in addressing poverty and development needs, particularly in rural areas."

Dr. Abdulah M Omar

Foreword to the Department of Transport's business plan for the financial year 2002/2003

Table of Contents

List of Tables

List of Figures

Glossary of Terms

PGWC	Provincial Government: Western Cape
DM	District Municipality
GMS	Gravel Management System
BOM	Blading Optimisation Module
Blading	Term given to the routine maintenance of unsealed roads where a road is graded by a grader
TTC	Total Transportation Cost – typically the sum of Agency Cost and Vehicle Operating Cost
Agency Cost	Cost incurred by a road agency to perform maintenance on a road network
VOC	Vehicle Operating Cost
HDM4	Highway Development and Management System version 4
TRH20	Technical Recommendations for Highways, Manual 20: The Structural design, construction and maintenance of unpaved roads

List of Symbols

$\vee$	logical OR operator
$\wedge$	logical AND operator
$\circ$	"concatenation" operator
$\sqcup$, $\cup$	"union" operator
$\in$	"element of" operator
$\mid$	"for which / where" operator
$\{\ldots\}$	set

1 Introduction

Investment in road infrastructure is expensive. In general, the citizens of a country pay for such investment by way of taxes and levies. Therefore, government agencies responsible for the management of such infrastructure bear a heavy responsibility to ensure that available funds are used efficiently and that the investment is protected by way of cost-effective maintenance. In addition to this, the principle of transparent governance requires that government is able to justify expenditure to its constituency. Such justification is aided enormously if decision-making processes are shown to be based on sound theoretical principles.

The problem of efficiently managing a gravel road network is a difficult one. This is due to the large number of factors that have to be accounted for in the management process. The responsible Roads Authority, for example, routinely has to deal with two key problems, namely the allocation/requirement of personnel and equipment in the various maintenance wards, and the optimal scheduling of routes followed by blading teams (i.e. people and equipment that are scraping or grading a gravel road) during routine maintenance. The maintenance schedule depends on the availability of blading teams and on the deterioration of the road segments, which, in turn, are influenced by factors like the road usage pattern, the weather and the surface material of the road. Planned blading schedules may be disrupted by extraordinary events like a severe rainstorm or a special agricultural activity like the harvesting of a crop which depends on a degree of quality during transport, e.g. deciduous fruit. The aim of maintaining the gravel network is to provide the best possible service to the communities that depend on it for their livelihood. These communities also have their special requirements and priorities which have to be taken into account.

It is clear that the complexity of a system such as the one described above is beyond what human beings are capable of solving by relying on experience and intuition, although it is true that they may over time develop techniques that give acceptable results. Consistent solutions can only be generated if they are based on a sound mathematical basis. Recent research has proved that problems in the field of engineering management can be modelled on the basis of a branch of mathematics known as graph theory. Although the mathematics of graph theory was initiated 300 years ago, practical problems could not be solved due to a lack of computational capacity.

In the closing decades of the 20th century, however, the development of powerful computers, accompanied by the development of object-oriented programming languages that can be used to map objects of a problem domain to the computer, has created the potential for modelling and solving real management problems within a reasonable amount of time.

To fulfil the potential offered by graph theory and computational capacity requires fundamental research aimed at developing the mathematical techniques that can be used to solve the problems of the domain under consideration. In addition to this, such techniques have to be efficiently mapped to the computer in order to be verified and improved.

1.1 Provincial Government: Western Cape Perspective

The Provincial Government: Western Cape (PGWC), Department of Transport and Public Works is currently in the process of developing a management system, known as the Gravel Management System (GMS). This system forms part of other data management systems currently used by the PGWC. The GMS can be classified as an operational level system assisting PGWC staff, District Municipality managers and Consulting Engineers with the scheduling and management of activities related to the gravel road network under the jurisdiction of the Provincial Government. Figure 1-1 shows the relation of the GMS to the other systems at PGWC.

During the compilation of the User Requirement Specification for the GMS, the need was identified to incorporate a tool that would assist the District Municipalities (DM's) with the optimisation of their blading program/schedule.

The study reported in this document was undertaken as part of the development of the Blading Optimisation Module (BOM) of the GMS.

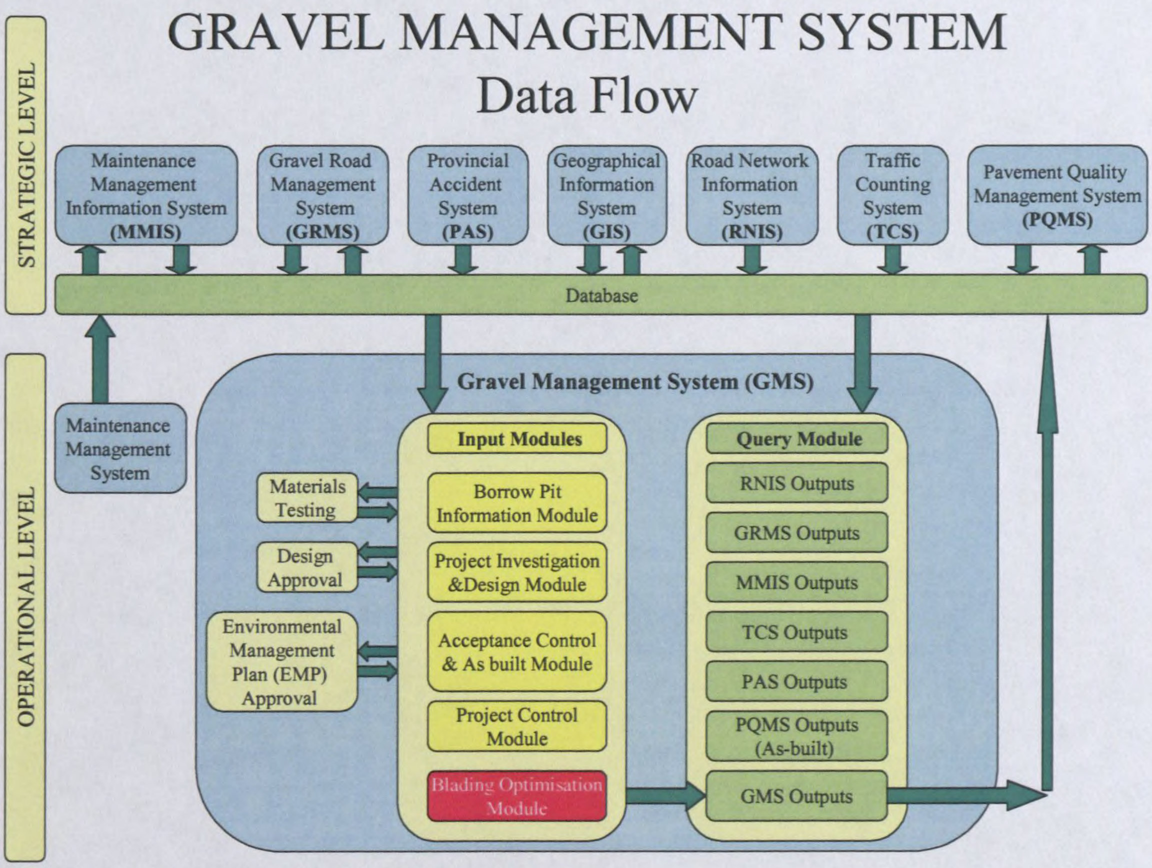


Figure 1-1: Relation of GMS to PGWC's other systems.

1.2 Study Objectives

The purpose of the BOM is to assist operational level managers to optimise routine maintenance work on the gravel road network. This research project was conducted in order to **develop, implement and test scheduling algorithms** that may be used **for the optimisation of routine maintenance** of the Gravel Road Network of the PGWC.

Due to the fact that a real-world problem was investigated, it was necessary to consider the constraints and variables under which the different district municipalities operate. Thus, the investigation of the constraints and variables forms an important part of this study and is reported in this document. The facts that crystallise from the investigation of the constraints and variables form the basis of the development of the scheduling algorithms.

1.3 Criteria for success

The main criteria for success are the provision of scheduling algorithms that:

1. are based on sound theory – both mathematical and engineering, and
2. incorporate the applicable constraints and variables under which gravel road maintenance are performed in the Western Cape.

Other important success criteria are:

- The algorithms have to be mapped efficiently to the computer so that performance may be evaluated and adjustments can be made if necessary.
- The use of the algorithms should require an acceptable amount of time, i.e. should be time efficient.
- The algorithms can be employed on computers with processing capabilities generally available at present, e.g. Pentium 4 micro-processor (or similar) with 1 gigabyte of RAM.

1.4 Document Overview

This document contains seven chapters. The first chapter introduces the research topic that was investigated. In order to determine the constraints and variables that influence maintenance of the gravel network, interviews were conducted with the personnel in the DM's – the interviews are reported in the second chapter. Based on the DM interviews, algorithm requirements were identified and the required algorithms were developed – this is reported in chapter three. Chapter four contains some background to Graph Theory and aspects thereof used in this study. In the fifth chapter the implementation of the required algorithms is discussed. In chapter six the scheduling algorithms that were developed are evaluated and compared. Chapter seven contains conclusions and recommendations.

2 Determining Constraints and Variables

The Province of the Western Cape is divided into 5 District Municipalities (DM's) whose combined geographical areas compose the geographical area of the Western Cape. Each of the DM's is responsible for the maintenance of the road network in its area. The road network is comprised of sealed roads and gravel (unsealed) roads[1].

The five DM's are: Eden, Overberg, Boland, West Coast and Central Karoo. The DM personnel in each of the DM's were interviewed to determine the constraints and variables applicable to the project [Personal Communication, 2004]. This chapter contains the interviews with each of the DM's. At the end of the chapter a synthesis of the interviews are included. The information obtained through the interviews formed the basis of the scheduling algorithm development.

2.1 Eden District Municipality

2.1.1 General Information

The Eden DM is responsible for the maintenance of 3 066 km of Gravel and Earth roads. This figure excludes the Minor Roads that have to be maintained. The Eden DM is sub-divided into 15 Maintenance Wards. According to [Personal Communication, 2004], approximately 80% of the roads have gravel thickness of less than 50 mm. (This could pose a problem for routine maintenance since not enough material is available to "protect" the sub-grade.)

At present, the Eden DM has 19 graders available for the regravel activities on and routine maintenance of the unsealed road network in the DM. Four graders work on the two regravel teams, while the remaining 15 graders work in the DM's maintenance wards.

The boundaries of the maintenance wards were determined before 1991 (apparently based on the resources [people] available). During 1991 the number of personnel was reduced by 100 and the question whether the boundaries still apply is relevant. It would be a worthwhile exercise to again determine the maintenance ward boundaries. However, it is worth noting that the boundaries are sometimes changed to bring a better "balance" between the different maintenance wards.

---

[1] The term gravel road is used generically, although not technically correct. In fact approximately 15% of the total gravel road network of the PGWC comprise earth graded roads, i.e. a gravel wearing course was not imported onto these roads.

Eden DM has received an amount of money from the PGWC for routine maintenance on the unsealed road network for the first time in 2003-2004 financial year. The amount of money is enough to pay for 5,5 hours per working day of active grading for each of the graders. This money was divided between the different roads in the DM according to an empirical prioritization model. The factors that affect the prioritisation model are:

- Traffic (AADT)
- Gravel Thickness and Quality
- Tourism
- Link Road
- Climate (Weinert's N-value)
- Type of Agricultural activities (Sensitive or Non-sensitive activities)

Each of the factors for a road section is evaluated and a score is assigned. The individual scores are added to obtain a score for the road section under consideration. The funds available for the maintenance of a specific road section are then determined based on the score for that section. After 6 months, the funds remaining for each road section were evaluated to see if the model could be further refined.

Currently the maintenance personnel have a monthly meeting where the activities for the forthcoming month are discussed. Grader activities in each of the maintenance wards are scheduled at these meetings. Based on their experience it has been found that the roads with the highest blading frequencies had frequencies in the region of 11 to 13 times per year. This corresponds well to the blading frequencies that are calculated by the GRMS and published in the GRMS reports.

The award of funds specifically for routine maintenance from the PGWC has resulted in a shift from the "on-demand" approach to a programmed approach. As was mentioned an empirical model was developed to determine blading frequencies based on the score and subsequent funds available per road. It is the opinion of Mr. Ottervanger that the program approach is superior to the previous "on-demand" approach – it seems that they can exercise better control over the network condition with the program approach.

Emergencies (e.g. resulting from heavy rain) take first priority for the blading teams. If reports are received of emergency situations, the relevant team is dispatched to attend to the situation and after completion of repairs, the team returns to the normal maintenance program.

2.1.2 Technical Information

The trigger that determines whether a road needs to be graded is the condition. Mr. Ottervanger referred to "volstruisnessies" being formed and that it is important to grade a road very soon after the first occurrences of volstruisnessies. These volstruisnessies may be described as the onset of the formation of corrugations – it would be a ridge forming in the road that induces a vibration in a vehicle's suspension passing over the ridge.

The following paragraphs contain anecdotal information [Personal Communication, 2004], i.e. the facts are not based on scientific studies, but rather on the experience of the DM personnel:

- It is the experience of the maintenance personnel that roads with lower quality material require more attention. The lower the material quality, the higher the blading frequency – this is because roads that have lower quality material deteriorate faster than roads with better quality material.

- It has been experienced that roads that have been regravelled under strict quality control procedures only needed to be bladed up to 14 months after construction. These roads had material quality conforming to the TRH20 specifications and the compaction effort was monitored closely.

- In Mr. Ottervanger's opinion more, lighter bladings are better than less frequent, heavier bladings.

The graders type used is Galion 120 or similar and a blading team consists of one operator and two labourers ("agterryers").

The blading teams aim for a road width of 7 meters with 4% camber. The road width results in 4 blade passes per section and this aids in preserving the road shape. An even number of passes means that the camber is achieved on both sides of the road centreline – if an odd number of passes is made it means that the centreline is graded horizontally (no camber) and this could result in the formation of centreline potholes due to insufficient drainage.

The blading teams have to open up side drains every time a road is graded. However, this policy may be ignored if the side drains are in good condition. Mitre drains are opened up every time the road is graded.

Two types of blading are defined: "dry" blading and "wet" blading. Dry blading is the blading operation after rain on a road (it would seem the term "dry" or wet refers to the absence (dry) or presence (wet) of a water truck and pneumatic tyre roller during blading). The rate of production achieved for dry blading is approximately 8 km per day. For the wet blading the rate of production achieved is approximately 1 km per day. In the case of wet blading a heavy pneumatic tyre roller is also present to compact the finished surface.

Mr. Ottervanger does not like the use of windrows due to problems with water (drainage) that could be encountered.

2.1.3 Eden DM Weighting Method for Fund Distribution

A further meeting was held with Mr Ottervanger. The objective of the meeting was to document the method used by the Eden DM to "weight" roads in order to distribute funds fairly between roads.

The methodology used is to calculate an index per road. The index is based on five individual parameters to which a weight – between zero and five (0 – 5) – is assigned. The parameters are:

- Traffic
- Gravel thickness (as reported by consultants when performing visual assessments)
- Climate (as indicated by Weinert's N-value [from GRMS annual report])
- Agriculture
- Tourism

The Traffic parameter's weight is assigned based on a number of traffic categories. These are reported in Table 2-1.

In the case of the Gravel thickness parameter, a number of thickness categories are defined and the weight is assigned based on these. (Refer to Table 2-1.)

The Climate parameter's weight is calculated as follows:

$$Weight = 5 - \frac{N}{2}$$

where

Weight	the weight assigned to the climate parameter
N	Weinert's N-value (in the range 0 to 10) [as reported in the GRMS annual report per road]

Values for Weinert's N-value higher than 10 is rare in the Eden DM area.

The weight for Agriculture is assigned based on the sensitivity of the agricultural products produced to riding quality. For instance, a road where a number of vegetable farms are located, producing produce such as tomatoes or peppers, will have a higher weight than a road where wheat or stock farming is predominant.

In the case of the Tourism parameter, the weight is assigned based on the number of guest houses and places of interest located on the road. Another factor considered is whether tourists use a road as a link road to places of interest or accommodation located along an adjacent road.

The Eden DM is divided into three "cost centres" (i.e. Langeberg, Klein Karoo and Outeniqua) that together comprise the 15 maintenance wards. Each of the cost centres has a superintendent that is responsible (among other duties) for the maintenance of the gravel road network. The indices of all the roads in the Eden DM are calculated and the total is then calculated for the DM. The total for each of the cost centres is calculated concurrently. The total budget allocated to routine gravel road maintenance is then apportioned to each cost centre based on the ratio of the cost centre's total to the total for the DM. The superintendent from each cost centre then has the opportunity to allocate funds for special maintenance to roads identified. These funds are subtracted from the maintenance portion and the remaining funds are allocated to each road based on the ratio of the road's index to the total for the cost centre. The funds allocated to a road can then be used to determine the blading frequency for the road. This is summarised in the flow diagram in Figure 2-1.

Table 2-1: Summary of the weights assigned to each parameter for Eden DM weighting method.

Parameter	Category	Weight
Traffic [ADT]	0 – 30	0
	31 – 80	1
	81 – 150	2
	151 – 350	3
	351 – 500	4
	500 +	5
Gravel Thickness [mm]	100 +	0
	81 – 100	1
	61 – 80	2
	41 – 60	3
	21 – 40	4
	0 – 20	5
Climate [Weinert's N-value]	0 to 5	
Agriculture	0 to 5	
Tourism	0 to 5	

The weighting method described here holds promise for use as a method of quantifying socio-economic factors when, for instance, benefits of maintenance are calculated. It may be worth noting that the method described here is a well-known tool from decision theory, called utility analysis. The utility analysis will be incorporated at a later stage for the calculation of Total Transportation Cost (TTC), thus enhancing the optimisation function based on the TTC[2].

2.1.4 Constraints

The following constraints were mentioned as the biggest factors affecting the effective maintenance of the unsealed road network in the Eden DM:

- Current condition of the network (both material quality and material quantity)
- Funding in terms of the "bigger picture" – this refers specifically to the quality of the machines that are provided by PGWC for the maintenance of the network. Currently the aim is to achieve an average of 90 to 95 % availability of machines. This is achieved on average, but there are machines for which the

[2] Refer to sections 3.1 and 3.3.

average availability is much less. These machines need major attention (e.g. engine rebuilding) but the funds are not available for this.

- Communication between the relevant parties – PGWC, DRE and DM
- Attitude of the public – would accept much worse riding quality on a sealed road than on an unsealed road [Personal Communication, 2004]. They complain more easily about unsealed roads than about sealed roads.

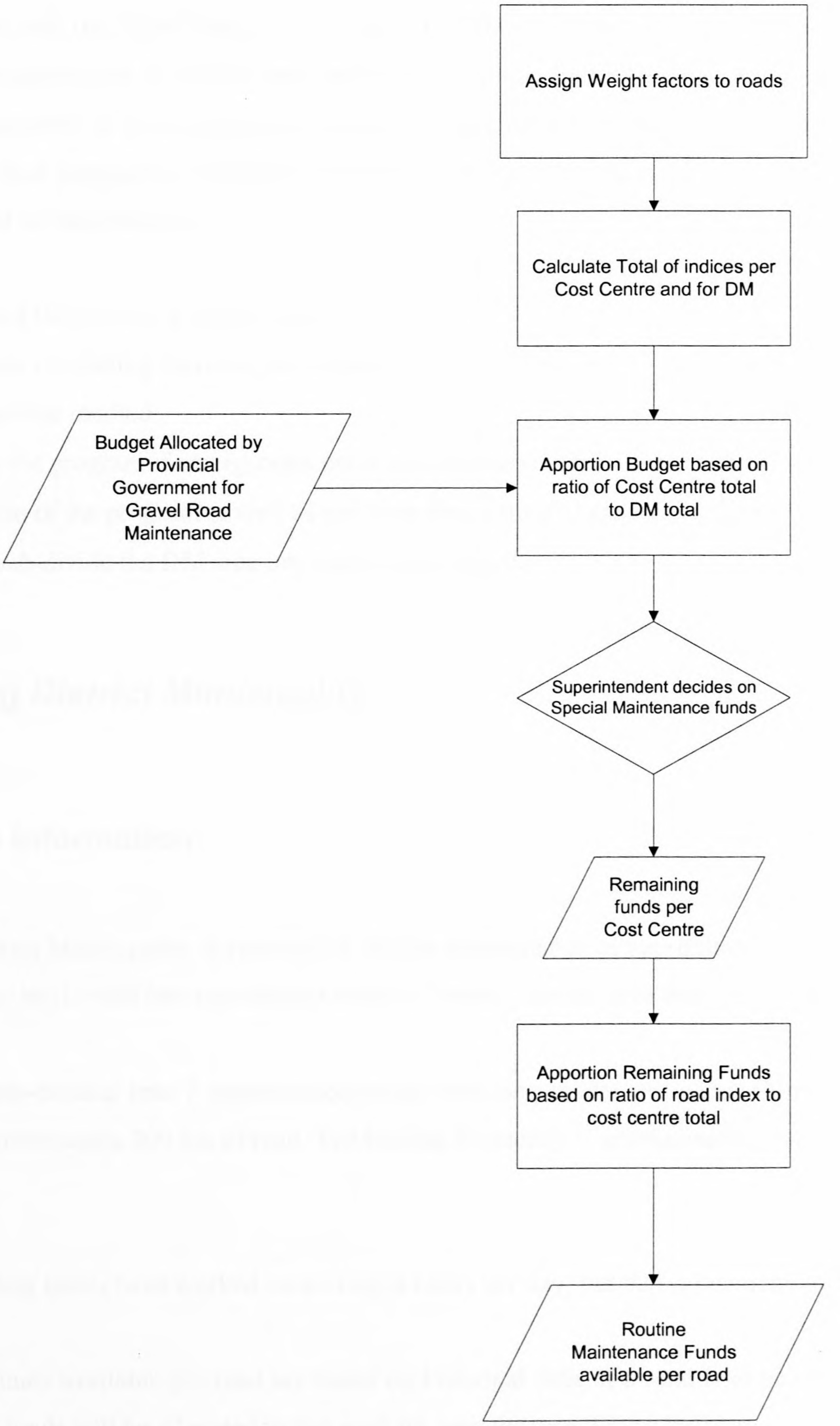


Figure 2-1: Flow diagram for Eden DM fund distribution.

2.1.5 System Requirements

The system needs to allow using the current method of determining blading frequency. (I.e. the empirical method based on the scoring system.) Programming has to be based on this blading frequency together with the type of blading required. Allowance has to be made for overriding the monthly program due to emergencies.

A system where a GPS transponder is attached to every grader for real time positioning would be appreciated if it could be coupled with the PAWCMap utility. This will enable the superintendents to see where every grader is relative to the maintenance ward/DM area and might result in better response time to emergencies. If an emergency is encountered in a maintenance ward and the position of all the graders is known on the network, the grader closest (not necessarily the grader normally working in that specific ward) to the problem may be dispatched to attend to the problem.

The system will need the following functionality:

- Optimisation of blading activities on a monthly basis, based on the blading frequencies as determined by the empirical method.
- Overriding the program if emergencies occur and reprogramming of the current month.
- Visualisation of the program as well as real time positioning of graders on the network.
- Ability to sub-divide the DM area into maintenance wards.

2.2 Overberg District Municipality

2.2.1 General Information

The Overberg District Municipality is responsible for the maintenance of approximately 1 400 km of unsealed roads. The area may be divided into two distinct areas: a "wetter" coastal area and "dryer" inland area.

The DM area is sub-divided into 7 maintenance wards with one grader per ward. Currently every grader is responsible for approximately 200 km of road. The blading frequency is approximately 3 or 4 months (i.e. 3 or 4 times per year).

Up to date the blading teams have worked on average 6 hours per day, but that is changing to 5 hours this year.

The maintenance funds available per road are based on historical data. If a road is to be regravelled during the financial year, less funds will be allocated to that road for maintenance during the year.

The different blading teams are free to work in each other's wards. They are not very "protective" of their areas.

They do not use the GRMS blading frequencies.

Emergencies take first priority above normal maintenance blading.

Currently, it is the responsibility of the grader operator to determine the route that will be followed.

In [Personal Communication, 2004] it was said that a structured blading program would be ideal, but at present they do not have enough graders to work according to a structured program.

2.2.2 Technical Information

The production that is achieved on average is 7 km/day for dry blading (without the addition of water by a watertruck). The graders work on 4 blade passes per road width. It is reasonable to assume that the production will reduce by 1/6 due to the change in working hours this year. If wet blading is applied the production rate may be halved (due to the addition of water by the watertruck – the grader operator has to stop and wait when the watertruck is being filled).

The type of blading applied may be classified as "heavy" blading, i.e. the road surface is broken down during blading and re-shaped by the grader.

Sometimes it is necessary to blade a section twice. This is usually the case when "ridges" have formed on the sides of the road. The material has to be spread over the surface first and then the normal blading activity can commence.

It is policy that potholes are first filled before a road is bladed. However, this is not always possible. When it is not possible, the road is graded as well as possible and then left to be compacted by the traffic. The road is then bladed again on the following day.

A blading team exists of an operator and two labourers ("agterryers"). There is a watertruck for every blading team, but it is not always used.

Side drains have to be maintained every time a road is bladed. However, it is not always necessary due to the good condition of the drains – if this is the case, the side drains are not attended to. Mitre drains have to be opened up every time a road is bladed.

Leaving windrows next to the road (as a material "store") is not favoured due to drainage problems that might be encountered [Personal Communication, 2004]. However, some of the older operators do leave windrows next to the road.

The roads are always graded to a 4% camber on both sides of the centreline. This is achieved by using an oil level that is set up in the grader's cab.

The opinion was expressed that using heavy pneumatic tyre rollers with the blading teams would improve the network condition [Personal Communication, 2004]. At present the blading teams are applying "crisis management" as is evident from the low blading frequencies. This is because the current number of graders is not sufficient for the network need. Mr. Van Eeden said that an additional 2 graders would improve the situation.

2.2.3 Constraints

The biggest constraints affecting effective routine maintenance on the unsealed network of the Overberg DM is:

- Funds – this relates to the quality (and quantity) of the available machines. The DM aims for a 90 % availability on the machines, but due to the quality of the machines, this is not easily achieved.

2.2.4 System Requirements

A system that will allow prioritisation of funds will be appreciated. At present the prioritisation is done based on historic data and a "gut feel".

Another aspect that will be very useful is one that will allow the superintendents to do route planning for the blading teams. The ideal route will be the one that is the shortest and that will optimise the network condition for the minimum funds.

2.3 Boland District Municipality

2.3.1 General Information

The Boland District Municipality is responsible for the maintenance of approximately 1 250 km of gravel and earths roads. The DM is divided into two distinct areas due to the topography. The areas are called the West area and East area. The East area is further sub-divided into the East-North area and the East-South area. The East-

South area encompasses the Worcester-Robertson area, while the East-North area encompasses the Ceres-Wolseley area. The East-North area has two distinct areas (based on climate) that may be identified as the Ceres-inner area (Ceres-Wolseley) and the Ceres-outer area (Ceres-Karoo).

In the West there are two graders available for the routine maintenance work. East-South has 3 graders, while East-North has 2 graders available.

The routine maintenance is not programmed at present, the methodology is rather to blade the roads selectively. However, they do work in cycles where they blade road by road. The current consensus is that they blade roads a maximum of 6 times per year (2 month interval) but the norm is 4 times per year. The blading teams work on average 5 hours per day.

The teams for East-South work from a base (Worcester or Robertson). The teams leave the grader and other equipment on the road and travel to the work site by bakkie. The East-North team for Ceres-inner also works from Ceres, while the Ceres-outer team sleeps out (due to the large distances that have to be covered).

The Boland DM has the full range of circumstances in the area. There are regions where there is an abundance of water, then there are regions with little water and then there are areas where there is practically no water available.

The type of blading applied is mostly wet blading, with the exception of the Ceres-outer area (due to the lack of water in the area). Wet blading is referred to when a water truck is part of the blading team. The teams do not have Pneumatic Tyre Rollers (PTR's) that work with them because these rollers are not available. The method of blading may be defined as "heavy" – the existing road surface is broken down and then shaped again by the grader. Because such large regions in the area are dry, the teams will move to road sections if rain has fallen on them. This is similar to responding to crisis situations. This means that the order of sections to be bladed have to be re-shuffled.

The Minor Road network under Boland DM's maintenance responsibility is graded once a year.

Factors that influence the blading frequency include whether the region is wet or dry, climate, topography, traffic (including type, e.g. agriculture), natural drainage, through traffic (due to weigh bridge working, better condition than other roads), seasonal effects (such as grape harvest time ("parstyd"), onion harvest time, etc.). Roads that are of significance to, for instance, the wine producers will be graded just before harvest time.

The operators are allowed to leave sections out if they are in good condition.

Based on the foregoing it is clear that they do not make use of the blading frequencies published in the GRMS reports.

The personnel talk to the operators every morning so they know where every team is working at any given time.

2.3.2 Technical Information

Production achieved is approximately 5 km per day in ideal conditions. This is for wet blading. For dry blading the production may be doubled.

A blading team comprises a grader, water truck and 2 or 3 labourers. Roads are bladed with four blade passes and on 4% camber.

The grader type used is Galion 120 or similar, although they also use Galion 140 graders (construction type). The experience indicates that the G140 graders give better production than the lighter graders.

In terms of availability, it is difficult to comment, because it is felt that the teams are not filling in the forms correctly (the teams think that they will be penalized) [Personal Communication, 2004]. Therefore, they feel that the statistics for availability is not correct. However, the opinion was expressed that the availability of the graders are not much higher than 60% – indicating poor performance for the machines.

As mentioned earlier the teams do not have a PTR available because the equipment is not available.

When a road is regravelled, a material store is left on the side of the road which the operators may spread over the roads when blading. The operators do not leave windrows on the road sides because of drainage problems associated with the practice.

The experience indicates that roads bladed with water added perform better than the roads bladed without water added.

Drains are opened when necessary and the condition of the drains are checked every time a road is bladed.

The DM personnel are of the opinion that they could maintain the road condition far better if they could just “sweep” the roads 10 times a year. However, with the current resources it is not possible to maintain the road network in this way.

2.3.3 Constraints

The biggest constraint for the Boland DM is funds. Higher funding could be used to appoint more people that could monitor the maintenance effort. More funds would also mean the equipment may be maintained better – e.g. re-building engines when needed. Another important constraint is training. At present the operators working on the graders were trained in the construction teams. However, there are no construction teams any more in PGWC. Therefore, the gentlemen feel that this may pose a problem in the near future.

The consensus is also that there are not enough machines available for the work load.

2.3.4 System Requirements

A system where the programming of the teams may be done will be appreciated.

It will also be useful if the system could show graphically where each team was working at any given moment – attaching a GPS transponder to each of the graders.

2.4 West Coast District Municipality

2.4.1 General Information

The West Coast District Municipality is responsible for the maintenance of 2 509 km of proclaimed (Main and Divisional) gravel and earth roads. A total of 6 866 km of Minor Roads are also under the maintenance authority of the DM.

The DM area is sub-divided into 13 maintenance wards. Thus, there are currently 13 gravel road maintenance teams in the West Coast DM. The Piekenierskloof Pass forms a natural boundary in the DM's area and the area is divided into 8 wards north of the pass (called the Cederberg area) and 5 wards south of the pass (called the Swartland area).

The maintenance teams all have base towns from where they work. However, the teams sleep out and are able to work 7 to 8 hour days.

The teams work in cycles that vary in length from 4 weeks to 6 weeks (in the larger wards). The result is that the average blading frequency on the network is 10 to 11 times per year.

Blading frequency is determined for each road section individually. The frequency is based on a number of factors, including material quality, traffic, tourism, sensitive agricultural products, topography etc. A formalised method for determining blading frequency is not used. The blading frequencies as suggested in the GRMS annual report are not used for determining the blading frequency.

The grader operator has to use his judgment when he passes a Minor Road section. If the section is in bad condition he will grade it, otherwise it will not be bladed.

The DM provides training for internal trainee operators on the construction teams. A policy of internal appointments is followed – this means that operators are trained internally and when posts become available the DM endeavours to fill the post from personnel internally.

Ward boundaries are not "fixed" and teams are allowed to work in each other's ward.

A blading team consists of the grader operator and two labourers. In general no water trucks are used and there are no pneumatic tyre rollers present.

2.4.2 Technical Information

The type of blading applied is a light blading where a "blanket" of material is spread over the road. Material is brought in from the side of the road. Typically, the material would be material that was ridden out of the wheel paths to the sides of the road.

A heavy blading (shaping) is applied at least once a year to every road section and this is done during the rain season (winter). The camber that is aimed for is a 3 % camber. The number of blade passes is always an even number (the number depending on the road width) in order to ensure that the road has a crossfall to both sides (drainage on the road).

As with the Boland teams, roads of strategic value are bladed just before the grape harvest time ("parstyd") – especially in the irrigation areas along the Olifants River.

The policy is that no windrows are left on the road after blading (windrows have to be bladed level with the road). However, this is difficult in certain areas where the road shape is inverted (resembles a canal). This is because the material forms windrows on both sides of the road.

Production rates achieved varies with the road classification (due to differences in width). On the Main Roads 8 km per day is achieved, on Divisional Roads 10 km per day is achieved and on the Minor Roads 12 – 15 km per day is achieved. However, these figures are based on a 45 hour workweek that is changing to a 40 hour workweek. Thus, the production rates should be adjusted accordingly.

Side drains are opened up every year just before the rainy season. During the rest of the year the side drains are opened as and when necessary. Mitre drains are opened every time the road is bladed.

When roads are badly corrugated, a Sandvik blade is used by the grader to break down the tops of the corrugations. The blade has a large number of "chisels" attached to it and the "chisels" break down the corrugations. The road is then bladed normally with a conventional blade attached to the grader.

The opinion was expressed that the 13 graders currently available are not sufficient for the current condition of the network and the rate at which it deteriorates [Personal Communication, 2004]. However, if more funds were made available for regravel work, they feel that the number of graders would be sufficient.

2.4.3 Constraints

The biggest constraint under which the West Coast DM operates is the funding available for regravel work. If these funds were increased, it is felt that the current number of graders would be sufficient to perform the necessary maintenance work to protect the gravel surfacing of the gravel roads.

However, at the current funding level it is the opinion of the personnel that this DM will very soon run into problems to continue with the way that they work (high frequency blading, light blading with the spreading of a "blanket" of material).

2.4.4 System Requirements

A system that will allow the responsible personnel to produce work programs for the blading teams would be appreciated. The system will have to allow the user to manually intervene to allow for e.g. emergencies.

A system that can interface with GPS transponders attached to the graders will be very much appreciated. This will allow the personnel to get real time updates of where the grader is and what it is busy with. This could also aid in better response time to emergencies because it is easier to see which grader is closest to the emergency site.

2.5 Central Karoo District Municipality

2.5.1 General Information

The Central Karoo District Municipality is responsible for the maintenance of 2 499 km of gravel and earth roads.

The DM is sub-divided into four maintenance wards for the purpose of the gravel road maintenance activities. The wards are Laingsburg (1 grader), Prince Albert (town) (1 grader), Murraysburg (1 grader) and Beaufort-West (2 graders). Thus, five blading teams perform the routine maintenance of the gravel roads in this DM.

The DM is located over a very large geographical area and the personnel have to travel large distances to perform their duties. Therefore, the blading teams all sleep out and the maximum number of hours are worked every day.

The type of blading applied to the roads is the light spreading of a "blanket" of material over the roads. This is done to protect the underlying gravel wearing course.

The teams are allowed to work over each other's boundaries. Thus, it might happen that a blading team will respond to a request from the public in the ward of another team.

The DM area is prone to thunderstorms in the summer months. The thunderstorms normally result in emergencies occurring (e.g. where roads wash away) and then the teams have to respond to these.

The funds available are divided between the roads that have to be bladed, and based on this division the average blading frequency in the DM is 4 to 6 times per year (every 2 to 3 months). The decision on when a road will be bladed is left to the roads superintendent or foreman responsible for the ward. The blading frequencies as reported in the GRMS annual report are not used at all.

Generally, there is good communication between the foremen and the blading teams and the roads where the teams are working are normally known very accurately.

The opinion was expressed that the number of graders is not sufficient [Personal Communication, 2004]. If the number was increased, it could result in better maintenance of the gravel roads and then emergencies would not have such a big influence on activities.

The training program for operators is not managed very well at this stage. Persons that want to qualify as grader operators are given a "bursary" to obtain their heavy vehicle license and then they are trained to operate the graders. Presently, the bursaries are allocated to people, but they do not make use of the opportunity to receive training and qualify for their heavy vehicle licenses. Thus, not many persons are getting training as grader operators.

2.5.2 Technical Information

As mentioned earlier, the blading teams spread a "blanket" of material over the roads. The rationale behind this is that more, lighter blades will protect the underlying gravel wearing course better – resulting in lower regravel frequencies (thus, longer time lapse between regravel actions). A blading team consists of a grader operator and two labourers.

The blading operation is a dry blading. Thus, there are no water trucks with the teams when doing the normal maintenance. However, heavy (shaping) blading is also performed on the roads and for this a water truck forms part of the team.

Production rates achieved by the blading teams vary, but the average values are 10 to 13 km per day (for three blade passes) and 8 to 10 km per day (for four blade passes). The number of blade passes is a function of the road width. There is an effort within the DM to widen roads, thereby increasing the number of blade passes.

Generally, the teams aim to achieve a 3.5 % camber on the roads. An oil level inside the grader cab aids the operator in achieving the cross-fall. The roads are bladed to the specified cross-fall when heavy blading is performed. Grading to a specified cross-fall is not always possible. The cases when this is not possible are when the road is too narrow or when the road shape is inverted (road resembles a canal).

Windrows are allowed on the roads and the labourers have to open up Mitre drains when blading is performed. The side drains are opened as and when necessary.

The graders used all conform to the Gallion G120 specification (lighter grader).

2.5.3 Constraints

The biggest constraint that the Central Karoo DM operates under is a lack of personnel to perform supervisory duties. The opinion was expressed that if more personnel were available for supervision that better control could be exercised over the blading teams.

A further constraint is personnel related problems. A symptom of these problems is absenteeism of the grader operators or team members.

2.5.4 System Requirements

A system that can interface with GPS transponders installed on the graders will be of good assistance. If the system is configured to provide real time data and the ability exists to obtain management information in real time, it will result in an improvement in the control over the blading teams. Problems and problem areas could be identified quickly if the ability existed to view the information in graphical format.

This functionality will also aid in providing monthly reports on the activities of the different blading teams.

2.6 Synthesis of Interviews

Based on the interviews conducted with the District Municipalities, the information contained in this section was compiled. Table 2-2 contains a summary of the most pertinent facts that came out of the interviews.

It is clear that the different DM's follow different policies in terms of gravel road routine maintenance. Two approaches to blading are followed. The first entails a "heavy" blading action – this means that the wearing course surface is broken down and shaped again. This is done both with and without the addition of water. The second is the spreading of a thin "blanket" of material over the road width (minimal interference with wearing course surface). The philosophy behind this is that more frequent, "less disturbing" maintenance will result in preserving the wearing course for longer.

Furthermore, there is a wide spread in the blading frequencies applied by the different DM's. The blading frequencies are based on experience and historic data. In the case of the Eden DM, the experience and historic data have been combined with certain environmental variables in an empirical decision function to allocate maintenance funds. It will benefit all the DM's if this method was documented. It could then be adapted for local circumstances to aid the other DM's in the allocation of maintenance funds. This will ensure continuity at the DM's in case the personnel responsible for fund allocation leave and their vast experience and knowledge with them.

Factors that affect blading frequency include traffic, material (quality and quantity), climate, occurrence of sensitive agricultural products along the road, whether the road is used as a tourism route, etc. These factors could be (or are) used to weight the funds allocated for routine maintenance, thereby determining the blading frequency.

Table 2-2: Summary of most pertinent facts.

Aspect	District Municipality				
	Eden	Overberg	Boland	West Coast	Central Karoo
Blading frequency based on a calculation	Yes	No	No	No	No
Blading frequency based on past experience	Yes	Yes	Yes	Yes	Yes
Length of gravel and earth roads (Proclaimed) [km]	3 066	1 400	1 250	2 509	2 499
Number of maintenance wards	15	7	5	13	4
Number of maintenance graders	15	7	5	13	5
Typical Dry Production Rate [km/day]	8	7	10	10	10
Typical Wet Production Rate [km/day]	1	3.5	5	5	5
Approximate average blading frequency [Number per year]		3 - 4	4	10 - 11	4 - 6
Maximum blading frequency [Number per year]	13	4	6	11	6

The interviews show clearly that good communication exists between the DM head office personnel and the blading teams. The point was made at each of the interviews that the site of work of each of the blading teams is known on a day-to-day basis.

However, the possibility of linking graders via GPS technology to a map facility will be of great help. It is a well-known phrase that a picture speaks more than a thousand words. In the case of the blading teams, this is also true. It is felt that it will make the response to emergencies faster, because each grader's position in a DM area will be visible in real time. Furthermore, the possibility also exists to extract management data much easier and faster for each blading team if the grader was fitted with a GPS transponder. This will enable DM head office personnel to pick up problems faster and respond to these problems quicker, thereby improving productivity.

One of the requirements of the BOM is that the scheduling facility must allow for manual intervention. This is necessary when teams have to respond to complaints or emergencies. These events are handled immediately if serious enough. Potentially, the schedule will have to be determined again for the remainder of the cycle.

From the discussions, it was seen that the norm for the DM's is to have a monthly planning meeting where blading activities are discussed and the next month's schedule determined. West Coast DM is the exception to this where one team works on a six-week cycle. This fact shows that the BOM will have to report monthly schedules.

The biggest constraint that the DM's operate under seems to be monetary for every DM. This is either reflected in the response that the equipment is old and that proper maintenance cannot be conducted on the machines due to a lack of funds. Furthermore, it is reflected by the fact that there is a shortage of personnel to control the routine maintenance of gravel and earth roads. Another response was that more funds for regravelling work could improve the network condition to such a level that the number of graders available would be sufficient for maintaining the network at desirable levels.

In terms of system requirements, the DM's need a system that could schedule blading operations efficiently. A result of this could be improved productivity and better use of scarce funds. Furthermore, the scheduling facility must be set up such that emergencies could be handled and the schedule has to be re-determined after responding to emergencies.

All of the DM's feel that a system that allows real time data transfer from a GPS linked system will benefit them. This will enable the personnel to know precisely where every grader is working at a given moment in time. Furthermore, management data may be obtained from such a system (almost immediately) resulting in better management of the blading operations.

3 Proposed Scheduling Algorithms

Based on the interviews with the District Municipality personnel, two algorithms were developed for scheduling routine maintenance on the Gravel Road Network. In this chapter the two algorithms are discussed after the background to the development of the algorithms has been sketched.

3.1 Development of the scheduling algorithms

The blading strategy currently employed by the DM's is not an optimum. This is because a team starts at one point in the ward network and works through the network until all the roads in the ward have been bladed. The team then starts again, at either the same point or a different point, and works through the network until all roads have been bladed. The result of this way of working is that the blading frequency of roads is not based on material, traffic and environmental characteristics, as it should be, but rather on the time it takes the team to work through its maintenance ward. This results in blading frequencies as low as four times a year irrespective of the traffic, material and environmental characteristics of individual roads.

It is because of the above facts that it became clear that the DM's need a tool which could assist in the optimisation of the blading activities in the different maintenance wards. However, before scheduling algorithms can be developed to optimise blading activities, it is necessary to define optimisation criteria. Success of maintenance strategies may be quantified in many different ways, two of which are:

- the network's riding quality (or roughness) is an optimum,
- the network's total transportation cost is an optimum.

In both cases the optimisation function would be the minimisation of the applicable criterion, i.e. network roughness or total transportation cost. E.g. if the ward network roughness is a minimum, the blading activities are considered at an optimum in the specific ward.

<u>Network Roughness Algorithm</u>:

As stated above the present blading strategy is to start at some point in the ward network and work through the ward until all the gravel roads have been bladed once. This is then repeated, sometimes starting at a different location in the ward network. It quickly becomes clear as one thinks about this approach, that factors such as the traffic, material quality and environment does not influence the number of times a road is bladed. Rather, the amount of time needed to work through the ward network dictates the blading frequency. E.g. if it takes two months to work through the ward network the blading frequency of the gravel roads contained in the ward is six times per year.

The objective of the network roughness algorithm is to minimise the network roughness. This may be done based on the condition of the gravel segments at the start of the scheduling period and the deterioration model applied. It is reasonable to assume that if a segment is bladed the required number of times and also each time close to the appropriate instant in time, then the average roughness for the segment will be an optimum, given the blading frequency, material, climate and traffic properties of the segment. This approach to the problem is an intuitive approach, i.e. a heuristic model, and it makes sense to do it this way: A segment has a predefined start condition and required blading frequency – the blading frequency is usually a value that has been arrived at over a relatively long period of time through interaction between the road agency and road users and blading the road the required number of times satisfies the needs of both the agency and road users. Based on the start condition, blading frequency and also the other model parameters, it is possible to calculate the next date at which the segment needs to be bladed. It is bladed the required number of times on or close to the required dates during the scheduling period, e.g. a year. Thus, since it was bladed the required number of times at or close to the appropriate dates, it is reasonable to assume that on average the riding quality was an optimum given the material, climate and traffic characteristics of the segment. If this is true for one segment in the maintenance ward, it will also be true for the remaining segments in the ward. Therefore, use of this scheduling algorithm will optimise the ward riding quality. A blading schedule resulting from the use of the minimise network roughness algorithm is a sequence of segments. The approach used is that a segment on the list is bladed and then the blading team travels to the next segment on the list and blades it. Thus, the team is required to travel between segments in the network.

Considering the above facts, the network roughness algorithm was developed. The aim is to maximise the riding quality of the ward network by employing an optimal blading schedule. The required blading frequency may either be calculated – based on an appropriate deterioration model; traffic, material and environmental variables – or the user may set a value based on experience. It is also necessary to consider the current condition of the road segments at the start of the scheduling window – another factor neglected in the current strategy.

Given that the road segments need to be bladed a different number of times and that they are in different condition states at the start of the schedule, the question still arises when to blade which segments. In order to answer this, it is possible to calculate the date at which each segment needs to be bladed next, i.e. the "next blade date", based on a segment's current condition and the required blading frequency. For a human it is easy to determine a schedule based on the above parameters by intuition. One would simply sort the segments based on their next blade date and follow the resulting sequence. This is in fact the crux of the network roughness algorithm – sorting the list of road segments in a ward network based on their next blade date.

The process for the determination of a schedule is as follows: The user sets the start segment, the time to blade it is added to the calendar and the next blade date for the segment is calculated. The list of segments is sorted according to next blade date and the one at the top is taken as the next segment in the blading sequence. The time to travel to the segment and blade it is added to the calendar and the next blade date for the segment is calculated. The list is sorted again, and the one at the top is set as the next segment in the blading sequence. The sorting and calculation of time to blade and next blade date is repeated until the time used is equal to or greater than the time allowed for the schedule. (Refer to section 3.2, Figure 3-2 and Figure 3-3.)

The approach described above differs significantly from the current practise where a road section, i.e. a sequence of segments between intersections, is bladed completely before moving onto the next road section. The proposed approach leads to travel between segments because the segment bladed after the previous one is not necessarily adjacent to it. Consequently, the algorithm also determines the shortest travel routes between segments.

Network Total Transportation Cost Algorithm:
This algorithm aims to minimise the Total Transportation Cost (TTC) of the network. TTC comprise the agency cost to maintain a road and the road user cost for the road as measured by Vehicle Operating Cost (VOC). For this algorithm the network is divided into sections, where a section is a continuous length of road between two intersections. This approach is closer to the current reality of routine maintenance on the Western Cape network, where the teams first finish blading a road between intersections before moving to the adjacent section. However, in the network cost algorithm, the next section may or may not be adjacent to the previous section. In general the blading team will have to travel to the next section. Another coincidental result of this approach is that the number of possible sequences that may be calculated is reduced for the network because combining the segments in the network to obtain sections may reduce the number of possibilities used in the generation of the blading sequences.

Initially it was thought possible to calculate all the simple routes[3] in a ward network. This set of simple routes could then be reduced to obtain the subset of simple routes for which the total time (or length), i.e. the time to travel and blade, of each route contained in the reduced set is equal to the time allowed, i.e. equal to scheduling window. From the subset the route that minimises the total transportation cost of the ward network could then be determined. This minimum route would then be taken as the schedule, since the time to travel and blade the route is equal to the scheduling window and the route minimises the total transportation cost. Thus, a schedule would be determined that optimises the blading activities in the ward in terms of total transportation cost.

However, two practical problems arise from the above approach:

1. Depending on the size of the network it may not be feasible to calculate all simple routes in it. This is because the number of routes is a function of the number of sections in the network. E.g. for a network containing 30 sections the number of all simple routes is equal to 30!, i.e. 30 factorial, a number of order 10^{32}. It is immediately clear that even with the desktop computers available today, it would be impossible to calculate such a large number of routes.

[3] For an explanation of simple routes (or paths), refer to section 4.3.1.

2. It may happen that when the set of all possible routes is reduced to the subset containing only routes equal in length to the scheduling window, one would end up with an empty set because not one route has a length exactly equal to the allowed time.

The two problems described above are addressed as follows:

1. The ward network is divided into sections, where a section is a continuous length of road between two intersections. The view is taken that all sections are connected to all other sections in the ward network – this is true since it is possible to determine a route from any point in the network to any other point in it. For the scheduling step only sections containing gravel segments are considered. Thus, a subset of the sections is considered, the subset being the set containing sections that have at least one gravel road segment in each section. Sequences of gravel sections of which the sequence time length is equal to the time allowed, subject to certain conditions explained below, are then generated in a random order. If a large number of random sequences are generated, e.g. 1 500 000, it is reasonable to assume that the random sequence for which the network TTC is a minimum is close to the absolute minimum sequence (or route) determined by calculating all possible simple routes in the network. This minimum sequence may be further optimised as described in section 5.2.4.4.
2. Some leniency is built into the time (or length) requirement by allowing sequences with length greater than or equal to 80% of time allowed and sequences with length less than or equal to 110% of time allowed.

The approach as described above is implemented in the network total transportation cost algorithm. From the above discussion it is clear that this algorithm takes a similar approach to the current practise, i.e. a road is bladed between intersections before moving onto the next road, with the important difference that the next road is not necessarily adjacent to the one being bladed.

The detail of the two algorithms is described in the remainder of this chapter.

3.2 Algorithm 1: Minimising Network Roughness

This section describes an algorithm that was developed to address the problem of programming the blading teams in the different maintenance wards to achieve an acceptable level of service, i.e. acceptable roughness level on average. The algorithm is based on the assumption that blading teams blade segments of a road and move between segments of different roads.

<u>Objective</u>: The objective of this algorithm is to **minimise the network roughness**. The cost of the solution does not influence the decision to accept or reject the solution.

As described above, the assumption is made that if a segment is bladed at the appropriate time, i.e. at or near the end of its blading interval, the network roughness will be minimised.

This algorithm may be viewed as a form of "crisis management" since the sequence of segments is determined by searching for the next worst segment.

Worst segment: The worst segment is simply the segment with the earliest next blade date. Consequently it is the first segment in a list resulting from sorting the segments according to the next blade date. Furthermore, it is proposed that each segment in a ward is assigned a weight, similar to Eden DM procedure, and then the weight may be used to determine which segment takes priority when more than one segment has the same next blade date.

Even though cost is not used as an evaluation criterion to accept or reject a solution, the two (opposing) objectives of the DM are addressed in the algorithm. The acceptable schedule will be the one that results in the best average network roughness while at the same time it will minimise the cost associated with rendering the service by minimising the travel time/cost between segments. Travel time/cost is minimised by determining the shortest route between two consecutive segments in the blading sequence.

The deterioration of the segments is used to determine when a segment will be bladed. The function will have to be set up such that the maximum allowable roughness is reached when the time elapsed since last blade is equal to the blading interval (the inverse of the frequency). Initially, it was proposed that a sigmoidal function be used for the modelling of roughness deterioration (refer to Figure 3-1). However, this was later changed to straight line deterioration similar to the HDM4 approach for steady-state deterioration [HDM4 Documentation, 2000].[4] The function will be set up such that it starts at time zero (just after blading) at a roughness value set by the user, i.e. Roughness after Blading. The road then deteriorates until the maximum allowable roughness, i.e. Roughness before Blading, is reached when the time elapsed since last blade is equal to the blading interval. This is an empirical approach to deterioration modelling. Roughness deterioration may, however, also be modelled based on other functions, e.g. TRH20 functions, HDM4 functions. The computer application will be developed such that it is possible to select the method of deterioration modelling.

[4] Refer to section 5.2.4.2.

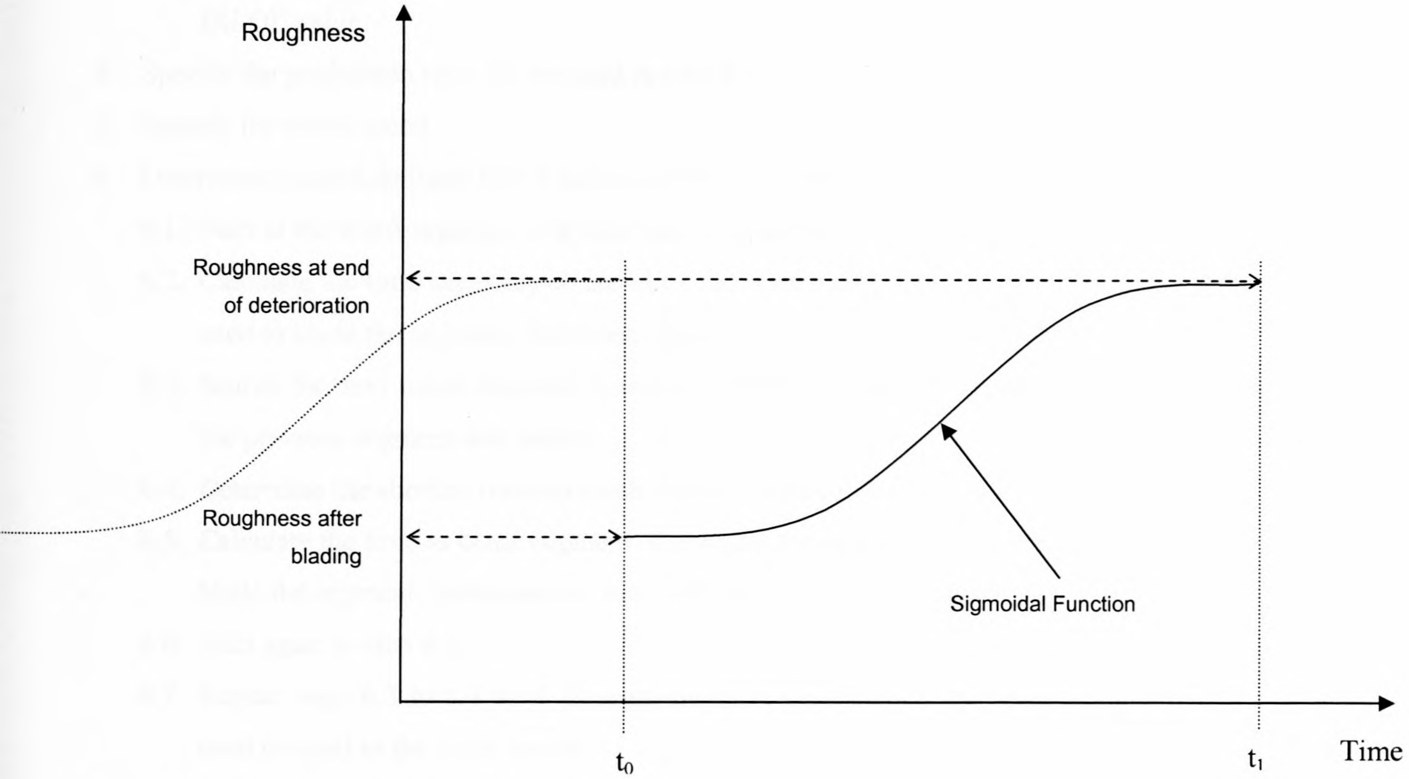


Figure 3-1: Sigmoidal deterioration function.

3.2.1 Variables

The following variables are important for the implementation of the solution algorithm.

1. Type of blading applied: Dry or Wet (including the production rate associated with the blading type).
2. Speed of travel between segments.
3. Blading frequency per segment.
4. *Condition of segments at the start of the analysis cycle.*

3.2.2 Algorithm

The proposed algorithm is summarised in the steps listed below (Refer also to Figure 3-2 and Figure 3-3):

1. Divide the DM road network into its wards.
2. Divide each road into its segments (sealed and gravel).
3. For each segment, assign values for the variables during the analysis cycle.
   3.1. Blading type

3.2. Condition in terms of category: Very Good, Good, Fair, Poor, Very Poor. Converted internally to IRI/QI[5] value.

4. Specify the production rates for wet and dry blading.
5. Specify the travel speed.
6. Determine a schedule (time based sequence) for the analysis cycle as follows:
   6.1. Start at the worst segment. The user may also specify the start segment.
   6.2. Calculate the time necessary to travel to and blade that segment. Increment the time used with the time used to blade the segment. Increment the cost by the cost to blade the segment.
   6.3. Search for next worst segment, keeping in mind that the other roads deteriorated during the time that the previous segment was bladed.
   6.4. Determine the shortest route to reach the next segment (measured in time units).
   6.5. Calculate the time to blade segment. Increment the time used with the time to reach the segment and to blade the segment. Increment the cost with the cost to blade the segment.
   6.6. Start again at step 6.3.
   6.7. Repeat steps 6.3 to 6.6 until all segments have been bladed the required number of times or the time used is equal to the cycle length.
7. Calculate agency cost for schedule determined in step 6.
8. Calculate the VOC for the given roughness per segment
9. Sum VOC for all segments to obtain VOC for network.
10. Calculate the TTC, i.e. Agency Cost + VOC.
11. Decide if schedule is acceptable. If schedule is not acceptable return to step 6 and start at a different segment.

[5] IRI: International Roughness Index; QI: Quartercar Index.

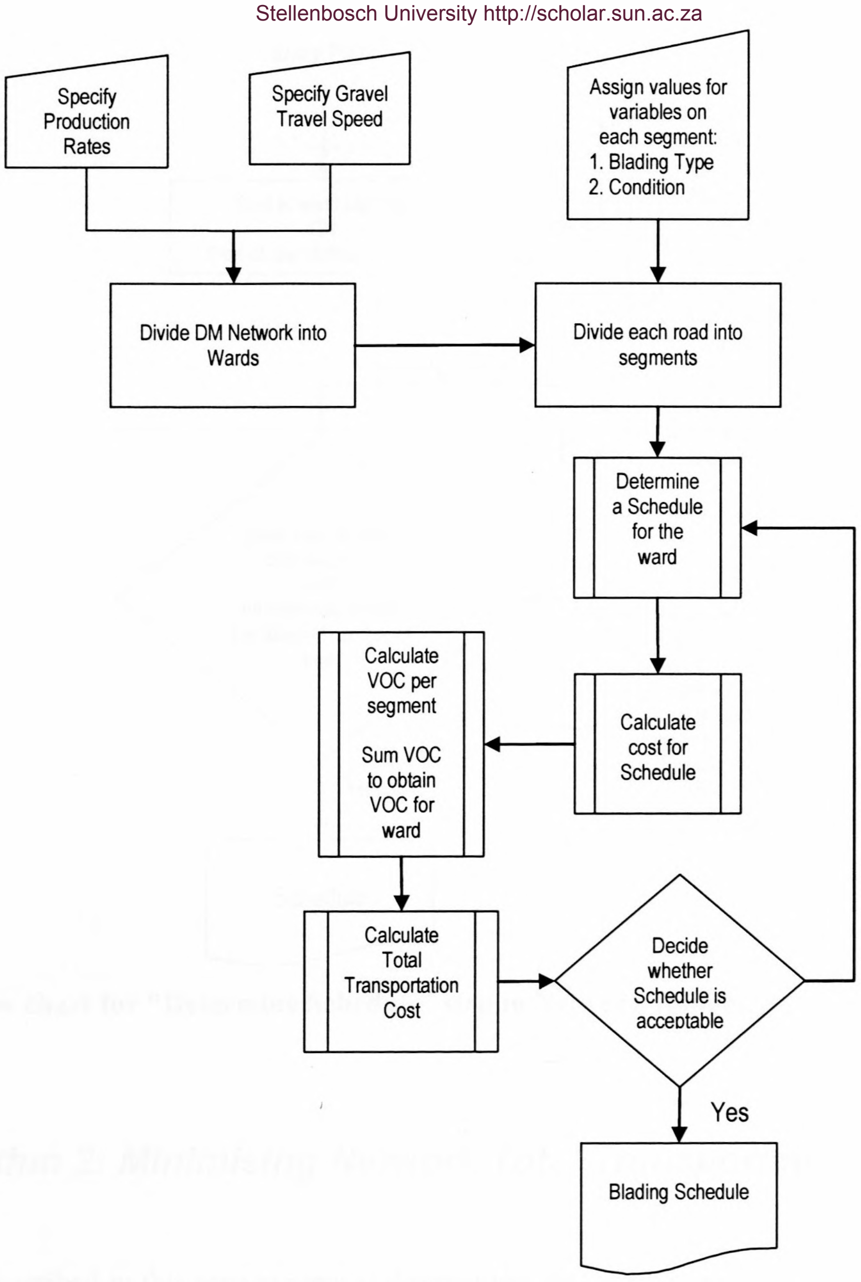


Figure 3-2: Flow chart for Network Roughness Algorithm.

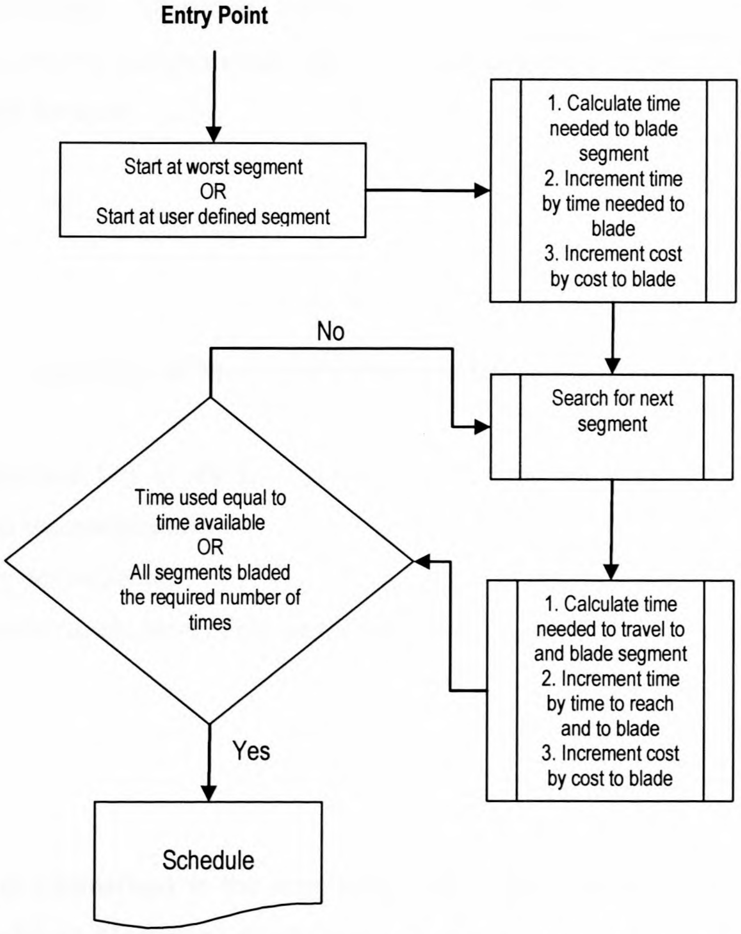


Figure 3-3: Flow chart for "Determine Schedule" step in Network Roughness Algorithm.

3.3 Algorithm 2: Minimising Network Total Transportation Cost

The algorithm described in this section aims at determining the schedule in a ward that will minimise the TTC for the ward network where TTC is defined as the sum of Agency cost and Road User cost.

Agency cost is the cost incurred to provide a certain level of service to the public. Agency cost is in this case defined as the difference between the cost to perform routine maintenance and the cost to do nothing. Routine maintenance cost: salaries, monthly plant rental cost, plant running cost, plant maintenance cost; Do nothing cost: salaries and monthly plant rental cost.

Road User cost is calculated by using Vehicle Operating Cost formulae [Burger, 2003; Appendix A]. VOC for a given road is directly dependent on the roughness of the road.

The algorithm is based on the assumption that blading teams blade road sections, i.e. all the gravel segments between intersections of the ward network, before moving to the next road section.

As with the Network Roughness Algorithm, road deterioration has to be modelled to obtain the average roughness of the network over the analysis cycle. Again the deterioration may be based on an empirical method, or on the TRH20 or HDM4 methods.

3.3.1 Variables

The following variables are important for the implementation of the solution algorithm.

1. Type of blading applied: Dry or Wet (including the production rate associated with the blading type).
2. Speed of travel between segments.
3. Blading frequency per segment.
4. *Condition of segments at the start of the analysis cycle.*

3.3.2 Algorithm

The proposed algorithm is summarised in the steps listed below (the first number of steps are essentially the same as for Network Roughness Algorithm). (Refer also to Figure 3-4.):

1. Divide the DM road network into its wards.
2. Each Road in a ward is further sub-divided into its segments (sealed and gravel).
3. Assign values for the variables during the analysis cycle:
   3.1. Blading type (if required during analysis period).
   3.2. Condition in terms of category: Very Good, Good, Fair, Poor, Very Poor. Converted internally to IRI/QI value.
4. Specify the production rates for wet and dry blading.
5. Specify the travel speed between segments.
6. The application determines all[6] possible simple[7] routes through the network for which the blading time and travel time, thus total time length, for the route is approximately equal to the schedule window. The number of routes is a function of the network structure.
   6.1. For each route, the network condition is calculated as a function of time. The average roughness for the cycle is also calculated.
   6.2. Based on the average roughness, the network TTC for each route is determined.
7. The route with the minimum network TTC is reported to the user.

---

[6] As stated earlier, it might not be possible to calculate all routes in a network, depending on the size of the network. The number of routes is equal to n!, i.e. *n factorial*, where n is equal to the number of vertices (nodes) in the network. This number becomes large very quickly as the number of vertices in the network increases. (Refer to section 5.2.4.4.)

[7] Refer to section 4.3.1.

8. The route is reported as the blading schedule for the ward. Alternatively, if it is not possible to calculate all routes in the ward, the user chooses to accept the route, at which time it becomes the blading schedule, or restarts the process.

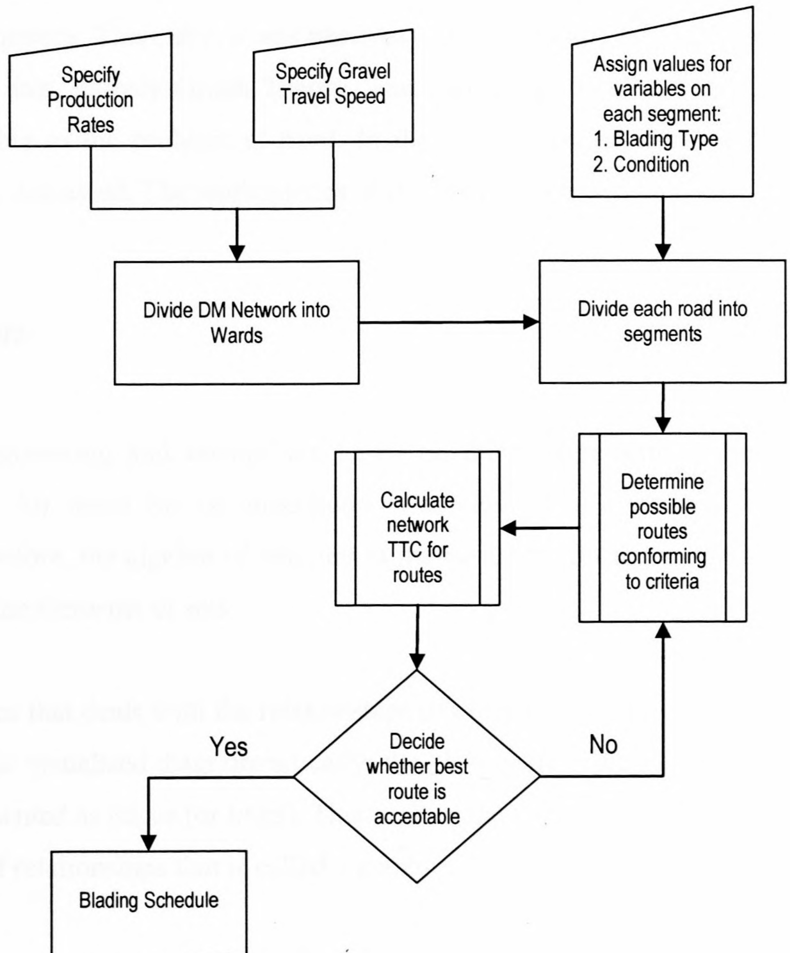


Figure 3-4: Flow chart for Network Cost Algorithm.

4 Aspects of Graph Theory used in this study

It is a requirement of both the scheduling algorithms described in the previous chapter that routes are calculated in the network – either shortest routes between subsequent segments in the blading sequence or routes that may become the blading sequence. Therefore, it was necessary to develop an understanding of the mathematical tool that could be used for this, namely Graph Theory. This chapter gives a brief overview of some concepts of Graph Theory applicable to the problem at hand. In the last section of the chapter the application of these concepts to the study is discussed. The mathematics of this chapter is based on [Pahl, 2001] and [Ruess, 2000].

4.1 Introduction

Many problems in engineering and science are based on the arrangement of objects and the relationships between these objects. An object can be understood as an element of a set or as a relationship between two elements of a set. Therefore, the algebra of relations is the basic tool for the mathematical representation of the relationships between the elements of sets.

The part of mathematics that deals with the relationships of objects is known as graph theory. The relationships between objects may be visualised diagrammatically: the objects are represented as vertices (or nodes) and the relationships are represented as edges (or links). This enables the visualisation of simple relationships in a plane. It is the visualisation of relationships that is called a graph.

The roots of graph theory date back to the 18th century. In 1736 Euler had to solve the famous Koenigsberg Bridge Problem. Koenigsberg was a medieval city in East Prussia that contained an island around which flowed the river Pregel. The different parts of the city were connected by seven bridges. The citizens of Koenigsberg wanted to know if it was possible to define a route to walk through the city so that each of the bridges is crossed only once. (Refer to Figure 4-1.)

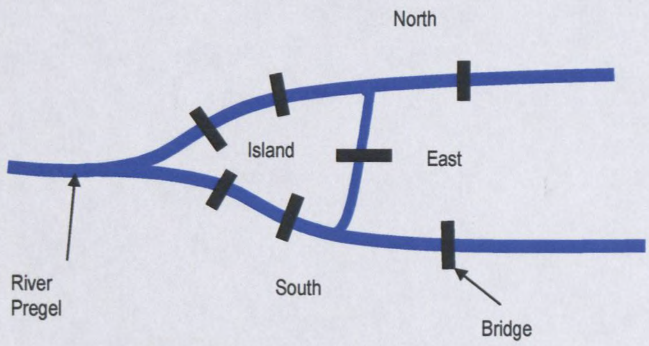
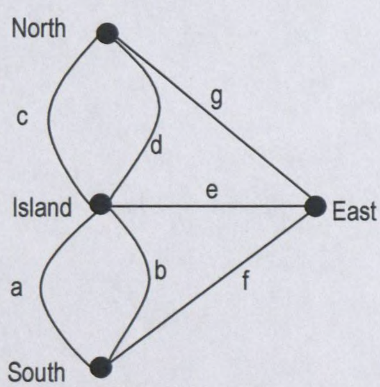


Figure 4-1: Map representing the Koenigsberg Problem.

It is possible to visualise this problem as a graph by taking the four locations (North, East, South and Island) as vertices and the seven bridges as edges. Thus, each edge connects the vertices between which a bridge exists. This is shown below, as well as the matrix form of the problem. The matrix is a "connectivity" matrix indicating that a path exists between two vertices.



	N	S	E	I
N	0	0	1	1
S	0	0	1	1
E	1	1	0	1
I	1	1	1	0

Figure 4-2: Graph representation of the Koenigsberg Problem.

4.2 Definitions

Some definitions are necessary so that the terminology may be better understood. Also refer to Figure 4-3.

Vertex: element of a set that represents some object in real life. These are often referred to as nodes in the Road Management Fraternity.

Edge: object representing a relationship between two vertices, i.e. between two objects of a set. These are often referred to as links in the Road Management Fraternity.

Graph: the visualisation of objects of a set and the relationships between them

Directed Graph: a graph where the direction of the relationships between vertices are given, e.g. a one-way street in an urban road network

Undirected Graph: a graph where the direction of relationships is not prescribed, e.g. a two-way street in an urban road network – this may also be visualised by two directed edges

Network: a graph of which the edges are assigned weights. The weights may be real numbers or literals and the meaning of each depends on the application.

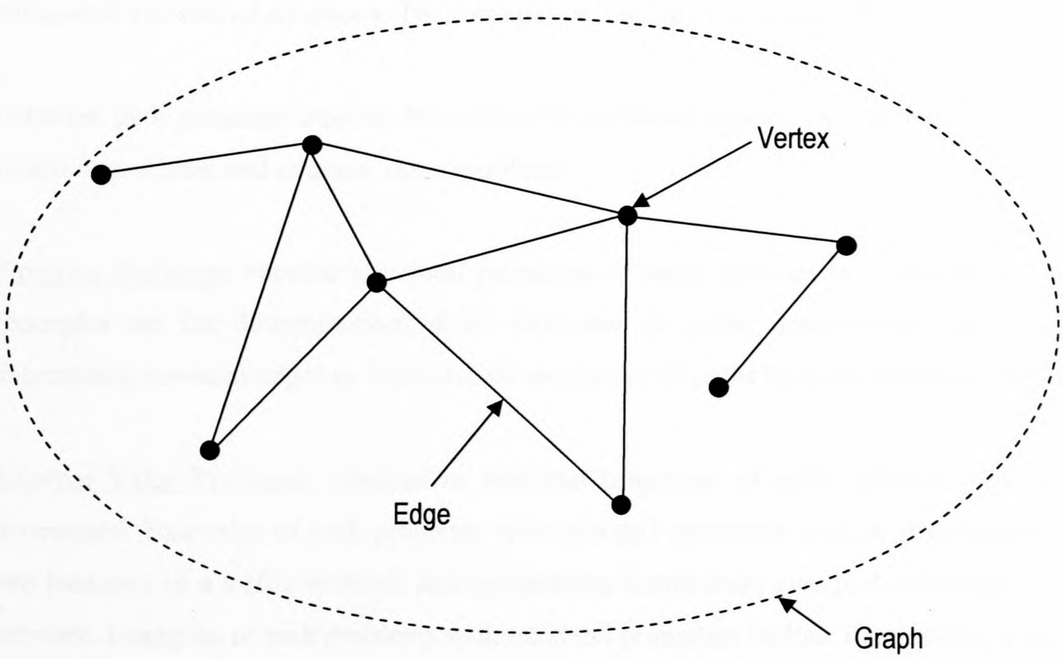


Figure 4-3: Concepts of a Graph.

4.3 Paths in Networks

The determination of a path with specific properties in a network is called a path problem.

Path: a path from i to k is an edge sequence with start vertex i and end vertex k. The path is said to be weighted if it is associated with a weight determined from the weights of its edges according to a given rule. Different path problems involve different rules for assigning weights to paths. For example, the length of a path is determined as the sum of the lengths of its edges, while the label of a path is determined as the concatenation of the labels of its edges.

Path Set: a set of paths with common start vertex i and common end vertex k is called a path set. The path set is said to be weighted if it is associated with a weight determined from the weights of its paths according to a given rule. Different path problems involve different rules for assigning weights to path sets. For example, for a minimum path length problem the length of the shortest path in the set is the weight of the set.

Path Algebra: Path sets may be combined using two operations, namely union and concatenation. The rules for the union and concatenation of weighted path sets can be formulated such that the resulting weights are determined directly without explicitly constructing the path sets. This leads to path algebras for networks. A path algebra is said to be either boolean, real or literal if the weights of the path sets are respectively boolean, real or literal.

The path algebras for different path problems may be generalised by abstraction. They are conveniently formulated in matrix and vector notation. Using path algebras reduces the solution of path problems to the solution of systems of equations. The formulation and solution of path algebras are discussed from section 4.4.

Different path problems may be formulated for different applications. A general distinction is made between structure problems and extreme value problems.

Structure Problems: specific structural properties of paths between two vertices in a network are determined. Examples are the determination of the existence of paths, determining simple or elementary paths and determining common edges or intermediate vertices of all paths between two given vertices.

Extreme Value Problems: minimal or maximal properties of paths between two vertices in a network are determined. Examples of path problems with minimal properties include determining a shortest path between two locations in a traffic network and determining a minimum cost path between two locations in a transport network. Examples of path problems with maximal properties include determining a most reliable path between two vertices in a communication network and determining a maximum capacity path in a road network.

4.3.1 Simple paths

A path is called a simple path if it does not contain any edge more than once.

4.3.2 Elementary paths

A path is called an elementary path if it does not contain any vertex more than once.

4.3.3 Eulerian paths

A simple path in a graph is called an Eulerian path if it contains all vertices of the graph.

4.3.4 Hamiltonian paths

An elementary path in a graph is called a Hamiltonian path if it contains all edges of the graph.

4.4 Path Algebra

Traffic or transportation problems as well as communication problems may be solved with path algebras. Most of these problems have in common the question of finding in the network a shortest or longest path between two vertices. Typically, we are looking for the cheapest, fastest or safest path between two locations. The basic elements of the formulation of path algebras are described below. For a complete treatment of path algebra see [Pahl, 2001].

Path set: If the set of paths between i and k contains all the paths, the set is said to be complete. A complete path set is designated by W_{ik}. If the path set is not complete, the set is a subset of W_{ik} and is designated by a_{ik}. The set of all possible subsets of W_{ik} is the power set of W_{ik} and is designated by $P(W_{ik})$.

The following path sets are special path sets:

zero set	$0_w = \{\}$	the set contains no path
unit set	$1_w = \{\lambda\}$	the set contains only the empty path without edges from vertex i to i
elementary path set	$a_{ik} = \{<i,k>\}$	the set contains exactly one path between vertex i and k, consisting of only the edge from i to k.

Every path set is assigned a unique weight. The binary operations of union ($\sqcup$) and concatenation ($\circ$) are defined for both the path set and the set of weights.

Path set matrix: The matrix representation of path sets is called the path set matrix. The properties of path set matrices are analogous to the properties of path sets. E.g. the matrix representation of the complete path sets is called the complete path set matrix and is designated by **W** if it contains the complete path set W_{ik} for each vertex pair (i,k) of the graph. The elementary path set matrix **A** contains the elementary path a_{ik} for each vertex pair (i,k) of the graph.

Operations on path sets: Path sets are combined using two operators, namely:

1. **Union**: the path set $u_{ik} \in P(W_{ik})$ that contains all paths which are contained in a_{ik} and b_{ik} is called the union of a_{ik} and b_{ik}.

   union: $u_{ik} = a_{ik} \sqcup b_{ik} := \{x \mid x \in a_{ik} \vee x \in b_{ik}\}$

2. **Concatenation**: the path set $u_{im} \in P(W_{im})$ that contains all paths which are formed by concatenating a path $x \in a_{ik}$ and a path $y \in b_{km}$ is called a concatenation of a_{ik} and b_{km}.

   concatenation: $u_{im} = a_{ik} \circ b_{km} := \{x \circ y \mid x \in a_{ik} \wedge y \in b_{km}\}$

The rules for the union and concatenation are also defined for the weights of the path set. Subject to certain conditions, the weights of the union and concatenation of path sets may be determined without explicitly constructing the path sets.

Operations on path set matrices: The operations on path set matrices may be defined analogous to the operations on path sets.

Union: $\mathbf{U} = \mathbf{A} \sqcup \mathbf{B} := [a_{ik} \sqcup b_{ik}]$

Concatenation: $\mathbf{U} = \mathbf{A} \circ \mathbf{C} := [\sqcup (a_{ik} \circ c_{km})]$

Path algebra: The domain $(P(\mathbf{W}); \sqcup, \circ)$ with the power set $P(\mathbf{W})$ of the complete path set matrix and the binary operations $\sqcup$ and $\circ$ is called a path algebra.

4.5 Closure of the elementary path set matrix

In a road network there is typically a very large number of possible paths between any two vertices. The complete path set matrix **W** contains all the complete path sets, while the elementary path set matrix **A** contains only the "direct" paths between adjacent vertices. The closure, **A***, equals the complete path set matrix **W**. For every vertex pair (i,k), it contains the set of all paths from i to k. In column i of closure **A*** we find all paths from the vertices k = 1, 2, 3, …, n to vertex i. Analogously, we may read all paths from a vertex j to each of the vertices k = 1, 2, 3, …, n in row j of matrix **A***. **A*** may, subject to certain conditions, be computed as follows:

$$\mathbf{A}^* := \mathbf{I}_w \sqcup \mathbf{A} \sqcup \mathbf{A}^2 \sqcup \mathbf{A}^3 \sqcup \ldots \sqcup A^s = \mathbf{W}$$

with $\mathbf{A}^n$ the product of **A** to the power n, i.e. $\mathbf{A} \times \mathbf{A} \times \mathbf{A} \times \mathbf{A} \times \ldots \times \mathbf{A}$ (n times); and s the stability index.

The stability index is defined as the least exponent s for which the union $\mathbf{A} \sqcup \mathbf{A}^2 \sqcup \mathbf{A}^3 \sqcup \ldots \sqcup A^s$ is not changed by adding terms A^m with m > s.

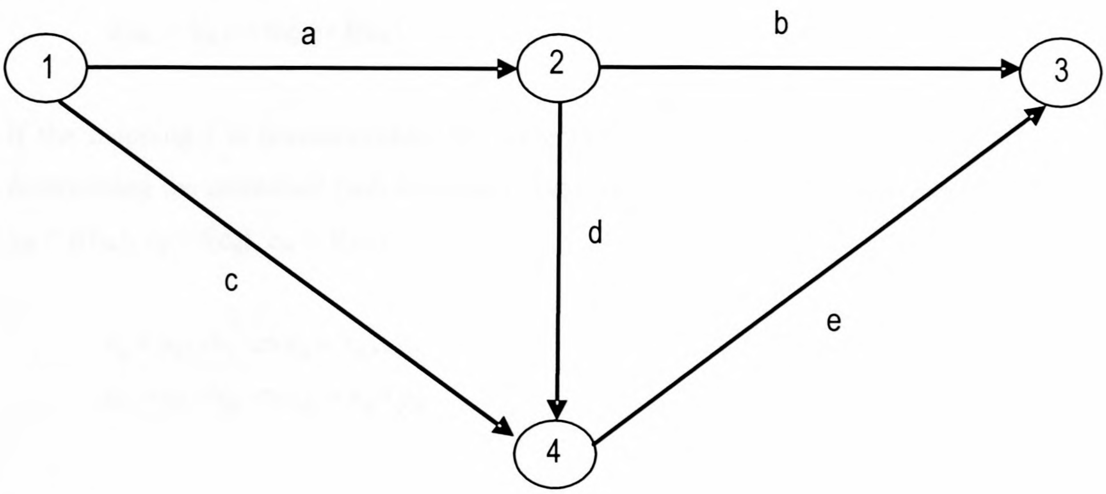
4.6 Systems of equations for path sets

The determination of the closure **A*** may be computationally expensive. In cases where paths to and from a specific vertex are required an alternative method of finding the paths is available.

The set of all paths from each of the vertices i = 1, 2, 3, …, n of a graph to a fixed vertex k is determined by the solution of the system of equations $\mathbf{x}_k = \mathbf{A} \circ \mathbf{x}_k \sqcup \mathbf{e}_k$. The vector **x** designates the k^{th} column of the closure **A***. The unit vector with the unit set $\mathbf{1}_w$ in row k is designated with $\mathbf{e}_k$.

Example:

For the given graph, find all paths from each of the vertices to vertex 3.



Solution:

$$\mathbf{x_3} = \mathbf{A} \circ \mathbf{x_3} \sqcup \mathbf{e_3}$$

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & \{a\} & 0 & \{c\} \\ 2 & 0 & 0 & \{b\} & \{d\} \\ 3 & 0 & 0 & 0 & 0 \\ 4 & 0 & 0 & \{e\} & 0 \end{array} \circ \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \sqcup \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}$$

The equations are solved by back substitution as follows:

$x_3 = 0 \sqcup 1 = 1$ (Paths from 3 to 3)

$x_4 = \{e\} \circ x3 \sqcup 0 = \{e\} \circ 1 \sqcup 0 = e$ (Paths from 4 to 3)

$x_2 = \{b\} \circ x3 \sqcup \{d\} \circ x4 = b \sqcup de = b, de$ (Paths from 2 to 3)

$x_1 = \{a\} \circ x2 \sqcup \{c\} \circ x4 = ab, ade \sqcup ce = ab, ade, ce$ (Paths from 1 to 3)

4.7 Weighted paths

The rules for the determination of the paths' weight are different for different types of path problems. The weight of the path for a shortest distance problem is a length, the weight for a fastest connection problem is a time.

A unique weight $z_{ik} \in Z$ is assigned to every path set $a_{ik} \in P(W)$ from a weight set Z. The unique assignment of $z_{ik} \in Z$ to $a_{ik} \in P(W)$ is a mapping: f: $a_{ik} \rightarrow z_{ik}$, i.e. $f(a_{ik}) = z_{ik}$. The mapping f is said to be homomorphic if the following statements hold:

$$f(a_{ik} \sqcup b_{ik}) = f(a_{ik}) \sqcup f(b_{ik})$$

$$f(a_{ik} \circ b_{ik}) = f(a_{ik}) \circ f(b_{ik})$$

If the mapping f is homomorphic, the weights of combined path sets may be determined without explicitly determining the combined path sets themselves, since the following implications hold for $x_{ij} = f(a_{ij})$, $y_{ij} = f(b_{ij})$, $y_{jk} = f(b_{jk})$, $z_{ij} = f(c_{ij})$, $z_{ik} = f(c_{ik})$:

$$c_{ij} = a_{ij} \sqcup b_{ij} \Rightarrow z_{ij} = x_{ij} \sqcup y_{ij}$$

$$c_{ik} = a_{ij} \circ b_{jk} \Rightarrow z_{ik} = x_{ij} \circ y_{jk}$$

4.8 Weight matrix

A path set matrix **A** may be mapped homomorphically to a weight matrix **Z** if a weight set Z and a homomorphic mapping f with $z_{ik} = f(a_{ik}) \in Z$ are given. Special weight matrices are the zero matrix $\mathbf{0}_z$, the identity matrix $\mathbf{I}_z$ and the elementary weight matrix **Z**.

Closure of the elementary weight matrix: The closure **Z*** may be determined analogously to the closure **A*** from the powers of **Z** without explicitly calculating **A***. The closure **Z*** contains for every vertex pair (i,k) the weight of the set of all paths between i and k.

Systems of equations for weights: Analogous to path sets it is possible to determine the weight of path sets which lead from any vertex i to a fixed vertex k by solving the system of equations $\mathbf{x}_k = \mathbf{Z} \circ \mathbf{x}_k \sqcup \mathbf{e}_k$.

4.9 Literal Path Algebra

The literal labelling of graphs forms the basis for literal path algebras. Literal path algebras for different path problems differ in the definition of the literal weight set and the definitions of the operations. Literal path algebras are particularly important for structure problems in graph theory.

In this study the use of an extreme simple path algebra (specifically: shortest simple path algebra) was necessary in order to determine the length of shortest paths between the vertices of the network, while also keeping track of the edge sequence that results in the shortest path. The remainder of this chapter is dedicated to the definition of the shortest simple path algebra used in this study.

4.9.1 Shortest simple path algebra

4.9.1.1 Problem definition

Let the edges of a graph be uniquely labelled by the characters of an alphabet A. A path is an edge sequence and is designated by a word, which is a sequence of the labels of its edges. A simple path from vertex i to vertex k does not contain any edge more than once.

For this study edges are weighted by real numbers indicating the length of the road segment represented by the edge. The shortest simple path from i to k is the simple path from i to k for which the length, calculated as the sum of the lengths of the edges contained in the sequence, is the minimum.

4.9.1.2 Weights

Let the path set a_{ik} containing paths from vertex i to vertex k be given. The set of simple words that yield the shortest path contained in a_{ik} is chosen as the weight z_{ik} of the path set a_{ik}. If the path set a_{ik} is the zero set 0_w, then $z_{ik} = 0z = \{\}$. If the path set a_{ik} is the unit set 1_w, then $z_{ik} = 1_w = \{\lambda\}$ containing the empty word λ. If the path set a_{ik} is neither the zero set 0_w nor the unit set 1_w, then z_{ik} is a set of simple words. Let the set of all simple words over the alphabet A, including the empty word λ, be S. Then z_{ik} is a subset of S, and hence an element of the power set P(S). Thus the weight mapping is defined as follows:

$$f(0_w) = 0_z = \{\}$$
$$f(1_w) = 1_z = \{\lambda\}$$
$$f(a_{ik}) = z_{ik} \in P(S) \text{ for } a_{ik} \notin \{0_w, 1_w\}$$

In the case of this study, the alphabet used was the set of segment definitions in the form Road Number, km from, km to (e.g. DR01001 0.0 12.84). Since the segment definitions are unique it is possible to form simple words from the set of segment definitions.

4.9.1.3 Operations

For the shortest length algebra, the operations $\sqcup$ and $\circ$ are defined for the weight set Z = P(S). Let the path sets a_{ik}, b_{ik} be weighted by sets x_{ik}, y_{ik} of minimum simple words, i.e. simple words corresponding to the shortest length path from i to k.

The weight $x_{ik} \sqcup y_{ik}$ of the union $a_{ik} \sqcup b_{ik}$ is the reduction **min**$(x_{ik} \cup y_{ik})$ of the union $x_{ik} \cup y_{ik}$.

The weight $x_{ik} \circ y_{km}$ of the concatenation $a_{ik} \circ b_{km}$ is the set of all simple words formed by concatenating a simple word from x_{ik} with a simple word from y_{km}.

union	:	$x_{ik} \sqcup y_{ik} := \min(x_{ik} \cup y_{ik})$
concatenation	:	$x_{ik} \circ y_{km} := \{x \circ y \in S \mid x \in x_{ik} \wedge y \in y_{km}\}$

The domain (P(S); $\sqcup, \circ$) is a literal path algebra with the zero element $0_z = \{\}$ and the unit element $1_z = \{\lambda\}$.

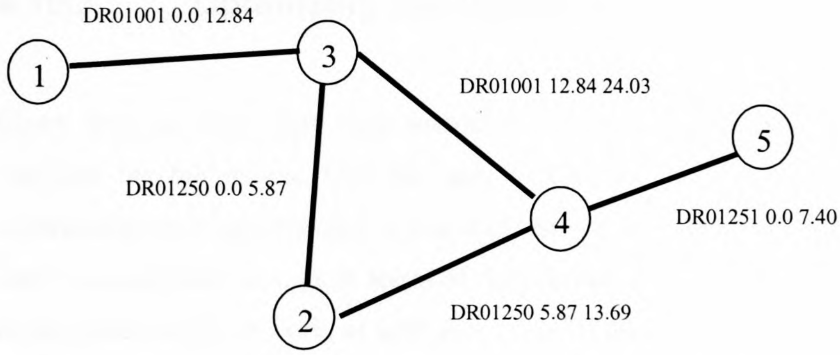
4.10 Calculating all shortest routes in the network

Given the shortest length simple path algebra defined in section 4.9.1, it is possible to calculate all the shortest routes in the network by calculating the closure of the elementary path matrix of the network. The closure is calculated as described in section 4.5.

4.10.1 Constructing the elementary weight matrix

The elementary weight matrix is commonly called the adjacency matrix. This matrix is constructed as a square matrix with the number of rows and columns equal to the number of vertices in the network. The elements of the matrix are the weights of the edges that connect the vertices, i.e. if there is no edge between vertex i and k, the element at positions <i,k> and <k,i> in the matrix is the zero element 0_z; if there is an edge between two vertices i and k, the element at positions <i,k> and <k,i> is a set containing the road segment between the two vertices (e.g {DR01001 0.0 12.84}); the element contained on the diagonal of the matrix is the unit element 1_z. Thus, the matrix has the unique characteristics of being square and symmetric about the diagonal.

Example:



1_z	0_z	*DR*010010.0 12.84	0_z	0_z
0_z	1_z	*DR*012500.0 5.87	*DR*012505.87 13.69	0_z
*DR*010010.0 12.84	*DR*012500.0 5.87	1_z	*DR*0100112.84 24.03	0_z
0_z	*DR*012505.87 13.69	*DR*0100112.84 24.03	1_z	*DR*012510.0 7.40
0_z	0_z	0_z	*DR*012510.0 7.40	1_z

4.10.2 Calculating the closure

The closure is calculated as detailed in the following calculation algorithm. Methods that calculate the union and concatenation of given weight sets, and a method that computes the product of two given weight matrices are required.

1. Repeat the following steps until the number of steps equal the number of vertices
   a. For the first calculation:
      ResultMatrix = product(adjacencyMatrix, adjacencyMatrix)
   b. For subsequent calculations:
      TempMatrix = product(adjacencyMatrix, ResultMatrix)
      if(TempMatrix = ResultMatrix), **break** out of the calculation loop *(equality indicates closure)*
2. Calculate the product as follows:
   a. Two matrices (m1 and m2) for which the product is to be calculated are given to this method
   b. Define a temporary matrix (temp) of the same dimension as the adjacency matrix with zero elements at all positions in the matrix
   c. For (integer i = 1, until i = the number of vertices, increment i by 1)
      For(integer j = 1, until j = the number of vertices, increment j by 1)
         For(integer k = 0, until k = the number of vertices, increment k by 1)
            temp<i,j> = union(temp<i,j>, concatenation(m1<i,k>, m2<k,j>))
   d. The temporary matrix is returned

The union and concatenation are performed as defined for the minimum simple path algebra.

4.10.3 Optimising the closure calculation method

Given that the elementary path matrix is square and symmetric about the diagonal, it is not necessary to calculate the full matrix. One has only to consider either the upper or lower triangle of the matrix. This optimisation was implemented in this study for the lower triangle of the matrix. Calculation of only the triangle leads to significant savings in terms of the number of operations that needs to be performed for the calculation of the routes in the network, as well as in terms of the computer memory requirements.

The savings in calculation operations are demonstrated in the algorithm for the calculation of the product as shown below.

1. Calculate the product as follows:
   a. Two triangles (t1 and t2) for which the product is to be calculated are given to this method
   b. Define a temporary triangle (temp) of the same dimensions as the adjacency triangle with zero elements at all positions in the triangle
   c. Define two temporary elements (element1 and element2)
   d. For (integer i = 1, until i = the number of vertices, increment i by 1)

For(integer j = 1, **until j = i**, increment j by 1)

For(integer k = 0, until k = the number of vertices, increment k by 1)

if(k > i)

element1 = t1<k,i>

else

element1 = t1<i,k>

if(k < j)

element2 = t2<j,k>

else

element2 = t2<k,j>

temp<i,j> = union(temp<i,j>, concatenation(element1, element2))

   e. The temporary triangle is returned

Again, the union and concatenation are performed as defined in the minimum simple path algebra.

5 Computer Implementation

An object model that maps the scheduling problem to the computer was developed. This model incorporates the proposed solution algorithms and it has two main components: a **database** containing network data; and a **runtime model** that allows user interaction, retrieval and storage of data from database, processing of data and reporting to the user. A prototype application was developed using the object-oriented programming language Java.

Aspects of the runtime model (section 5.1) and its implementation (section 5.2) are described below.

5.1 Package Structure

The object model comprises four main packages, i.e. **gui**, **model**, **network**, **schedule** and **utils**. The packages may contain sub-packages. The packages and sub-packages contain classes that define the required properties and implement the necessary methods to provide the application's functionality. The packages and sub-packages are listed in Table 5-1.

Table 5-1: Packages and sub-packages contained in application.

Package	Sub-package(s)	
gui		
model		
network	component	
	graph	*pathfinder*
	gui	
	utils	*detmdl*
		trgBF
schedule	utils	
utils	database	
	set	

The packages of the application are described in more detail below.

5.1.1 Package: gui

This package contains the classes that form the graphical user interface (gui) of the pilot application. The functionality of the classes will not be described in detail. Also contained in this package are three utility classes (SegmentBFData, SegmentCurrentConditionData and ConditionCategoryData) that are used to display blading frequency, current condition and condition category data in tabular form.

WOBSApplication

Class WOBSApplication extends the Java-class JFrame and is displayed when the application is started. When this class is instantiated an object of class MainButtonPanel is also instantiated and displayed. Class WOBSApplication has one attribute, i.e an object of type Model. The Model object is used throughout the application to provide the data sets used during scheduling and this object has to be instantiated by clicking the Initiate Model button.

MainButtonPanel

Class MainButtonPanel contains buttons allowing the user to perform tasks and processes necessary to draw the network, calculate blading frequencies, set the current condition of the segments, etc.

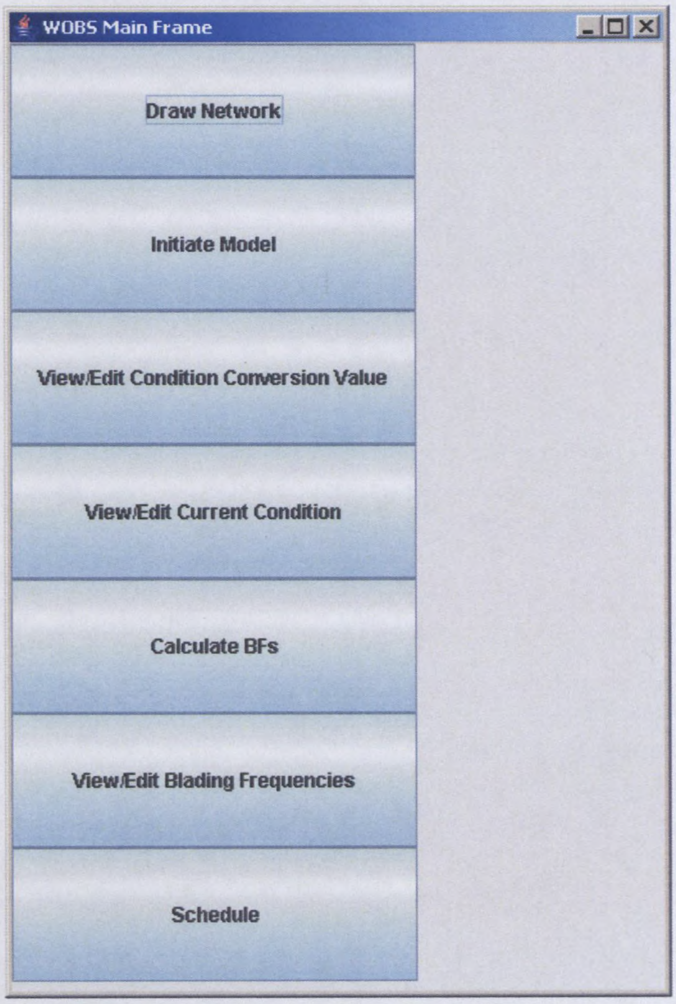


Figure 5-1: Classes WOBSApplication and MainButtonPanel.

EditConditionCatsFrame

Class EditConditionCatsFrame contains an object of type JTable that contains the five condition categories used, i.e.Very Good to Very Poor, with the corresponding QI value to which the category is converted. The user is able to edit and save the conversion values in the data base in table RoughCatLookUp.

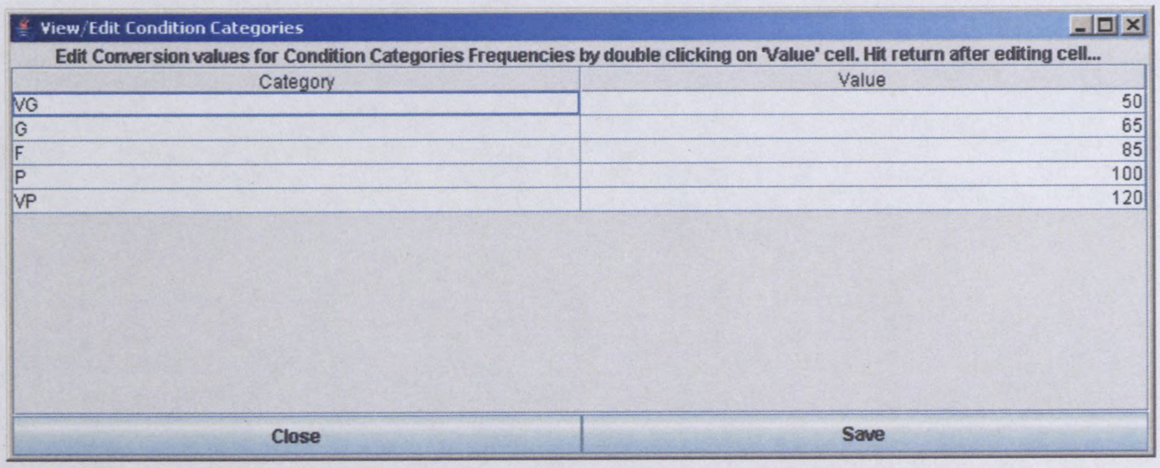
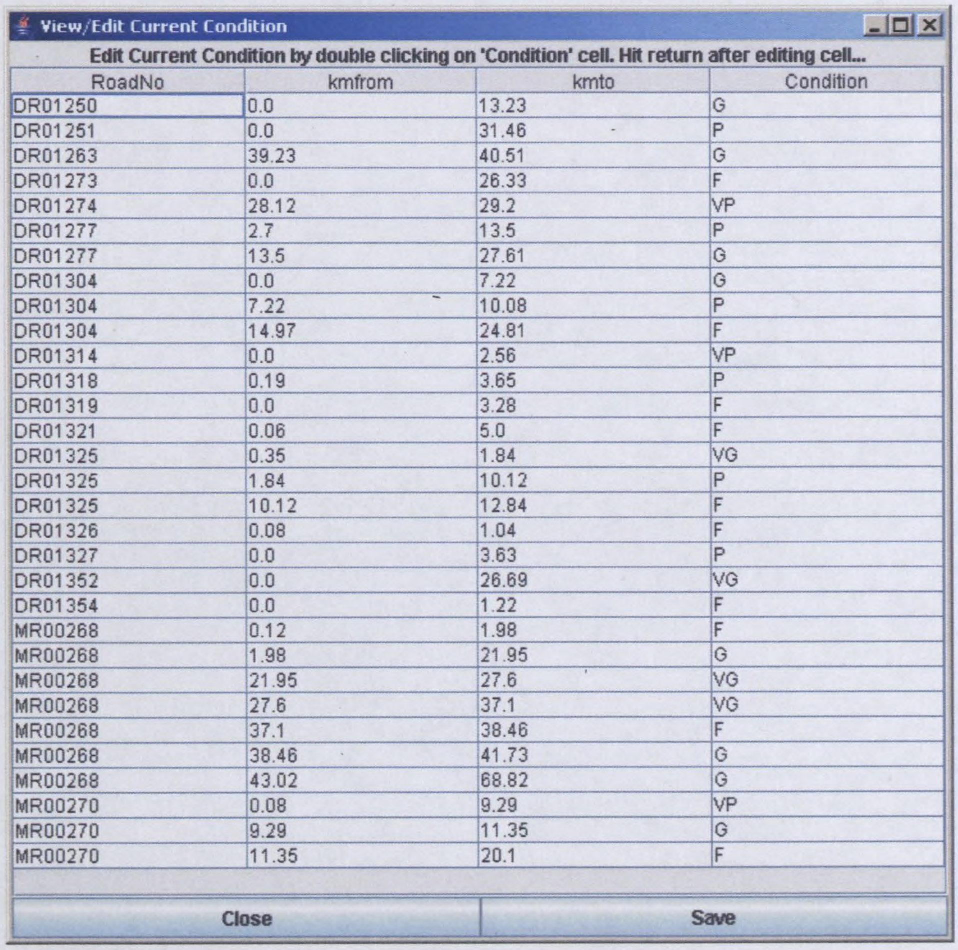


Figure 5-2: Class EditConditionCatsFrame.

EditCurrentConditionFrame

Class EditCurrentConditionFrame contains an object of type JTable that contains the current condition data of the gravel segments in the ward. The JTable allows the user to edit the current condition of the segments by double clicking on the relevant segment's current condition cell. Only the current condition of the segments can be edited, ensuring data integrity.


View/Edit Current Condition

Edit Current Condition by double clicking on 'Condition' cell. Hit return after editing cell...

RoadNo	kmfrom	kmto	Condition
DR01250	0.0	13.23	G
DR01251	0.0	31.46	P
DR01263	39.23	40.51	G
DR01273	0.0	26.33	F
DR01274	28.12	29.2	VP
DR01277	2.7	13.5	P
DR01277	13.5	27.61	G
DR01304	0.0	7.22	G
DR01304	7.22	10.08	P
DR01304	14.97	24.81	F
DR01314	0.0	2.56	VP
DR01318	0.19	3.65	P
DR01319	0.0	3.28	F
DR01321	0.06	5.0	F
DR01325	0.35	1.84	VG
DR01325	1.84	10.12	P
DR01325	10.12	12.84	F
DR01326	0.08	1.04	F
DR01327	0.0	3.63	P
DR01352	0.0	26.69	VG
DR01354	0.0	1.22	F
MR00268	0.12	1.98	F
MR00268	1.98	21.95	G
MR00268	21.95	27.6	VG
MR00268	27.6	37.1	VG
MR00268	37.1	38.46	F
MR00268	38.46	41.73	G
MR00268	43.02	68.82	G
MR00270	0.08	9.29	VP
MR00270	9.29	11.35	G
MR00270	11.35	20.1	F

Close Save

Figure 5-3: Class EditCurrentConditionFrame.

CalcBFsFrame

Class CalcBFsFrame is instantiated when the user clicks the Calculate BFs button in the pilot application. This class contains a textfield where the target roughness is set for the network roughness blading frequencies; as well as two buttons (OK and Cancel). When the OK button is clicked, a new object of type BladingFrequenciesCalculator is instantiated and its calcBFs method is called with the contents of the textfield passed as the target roughness.

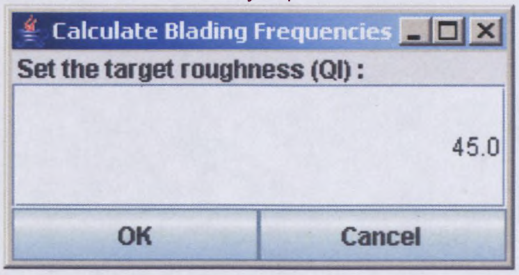
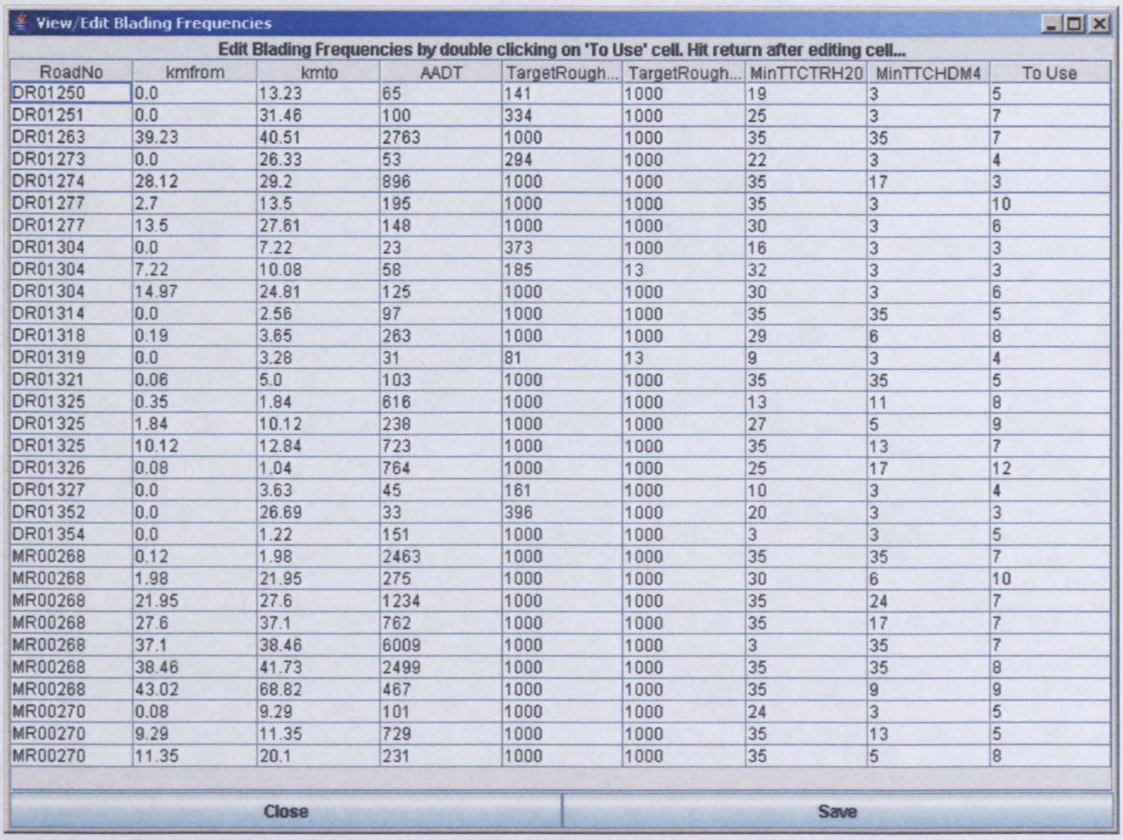


Figure 5-4: Class CalculateBFsFrame.

<u>EditBFsFrame</u>

Class EditBFsFrame is similar class EditCurrentConditionFrame and allows the user to set/edit the blading frequency for the gravel segments in the ward that should be used during scheduling. Again, only the blading frequency can be edited by double clicking on the To Use cell of the relevant segment.



RoadNo	kmfrom	kmto	AADT	TargetRough...	TargetRough...	MinTTCTRH20	MinTTCHDM4	To Use
DR01250	0.0	13.23	65	141	1000	19	3	5
DR01251	0.0	31.46	100	334	1000	25	3	7
DR01263	39.23	40.51	2763	1000	1000	35	35	7
DR01273	0.0	26.33	53	294	1000	22	3	4
DR01274	28.12	29.2	896	1000	1000	35	17	3
DR01277	2.7	13.5	195	1000	1000	35	3	10
DR01277	13.5	27.61	148	1000	1000	30	3	6
DR01304	0.0	7.22	23	373	1000	16	3	3
DR01304	7.22	10.08	58	185	13	32	3	3
DR01304	14.97	24.81	125	1000	1000	30	3	6
DR01314	0.0	2.56	97	1000	1000	35	35	5
DR01318	0.19	3.65	263	1000	1000	29	6	8
DR01319	0.0	3.28	31	81	13	9	3	4
DR01321	0.06	5.0	103	1000	1000	35	35	5
DR01325	0.35	1.84	616	1000	1000	13	11	8
DR01325	1.84	10.12	238	1000	1000	27	5	9
DR01325	10.12	12.84	723	1000	1000	35	13	7
DR01326	0.08	1.04	764	1000	1000	25	17	12
DR01327	0.0	3.63	45	161	1000	10	3	4
DR01352	0.0	26.69	33	396	1000	20	3	3
DR01354	0.0	1.22	151	1000	1000	3	3	5
MR00268	0.12	1.98	2463	1000	1000	35	35	7
MR00268	1.98	21.95	275	1000	1000	30	6	10
MR00268	21.95	27.6	1234	1000	1000	35	24	7
MR00268	27.6	37.1	762	1000	1000	35	17	7
MR00268	37.1	38.46	6009	1000	1000	3	35	7
MR00268	38.46	41.73	2499	1000	1000	35	35	8
MR00268	43.02	68.82	467	1000	1000	35	9	9
MR00270	0.08	9.29	101	1000	1000	24	3	5
MR00270	9.29	11.35	729	1000	1000	35	13	5
MR00270	11.35	20.1	231	1000	1000	35	5	8

Figure 5-5: Class EditBFsFrame.

<u>ScheduleFrame</u>

Class ScheduleFrame contains an object of type ScheduleDetails; buttons allowing the user to set the start and end dates of the schedule; and buttons that allow the user to determine a schedule for the Network Roughness Algorithm or Network Cost Algorithm.

ScheduleDetails

Class ScheduleDetails extends the Java-class JTabbedPane. It contains three tabbed panes containing a number of text fields where the scheduling parameters (e.g. production rate, travel speed, start segment, etc.) may be set. The values of the parameters as set in the tabbed panes are then used in the further scheduling.

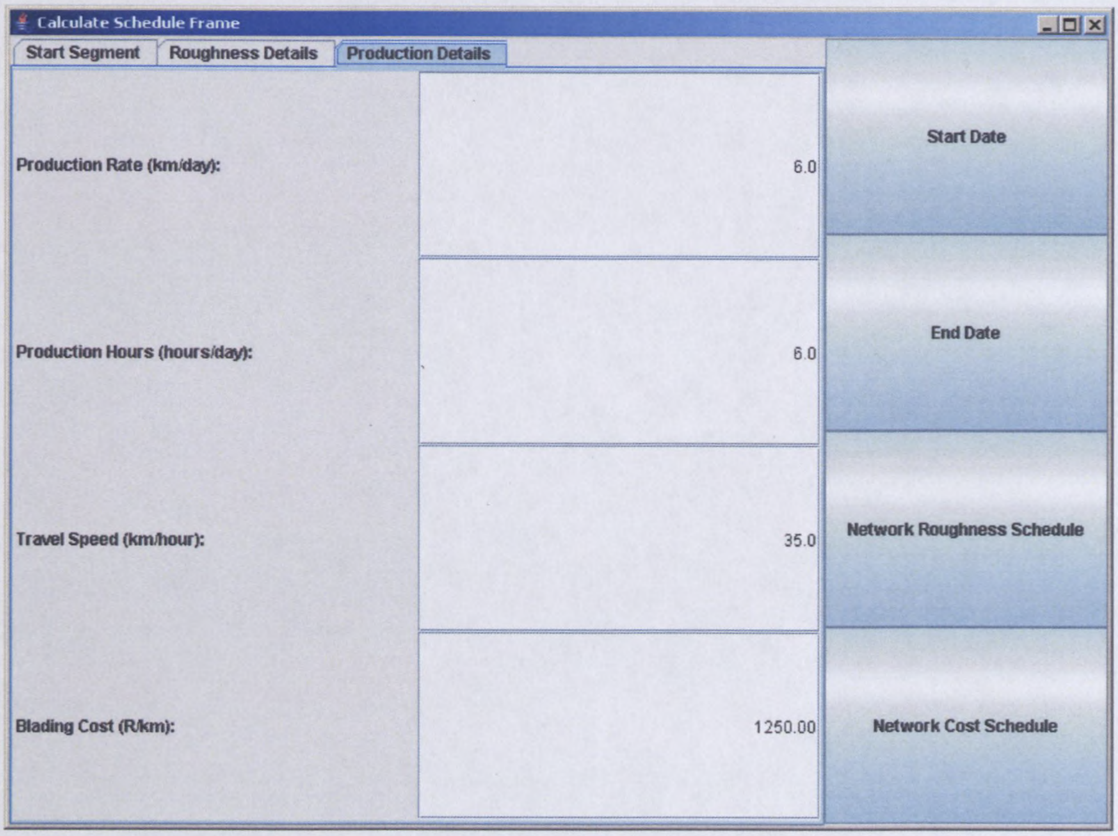


Figure 5-6: Class ScheduleFrame.

CalendarDialog

Class CalendarDialog allows the user to select a date. An object of this class is instantiated when either the Start Date or End Date buttons of class ScheduleFrame are clicked by the user.

Figure 5-7: Class CalendarDialog.

5.1.2 Package: model

This package contains one class: Model. An object of class Model contains sets of the following data:

- the sets of vertices and edges (refer to section 5.1.3.2) of the graph
- a set of the segments comprising the network (i.e. the gravel and sealed segments of the network) (refer to section 5.1.3.1)
- two sets of **sections** (refer to section 5.1.3.2.1): the set of sections contained in the network; coupled with this is the set of sections containing gravel segments

These data sets form the basis of the calculations for the scheduling of routine maintenance on a gravel road network.

Class Model has the following important methods:

- readSegments: the *links* table in the database is accessed and the objects of type Segment (package network.component) is instantiated from the data
- readVertices: the *nodes* table in the database is accessed and objects of type Node (package network.graph) is instantiated from the data
- readEdges: the *links* table in the database is accessed and objects of type Edge (package network.graph) is instantiated from the data
- calcRoutes: an object of type ShortestSimplePathFinder is instatiated (package network.graph.pathfinder) and the shortest routes between all vertices in the graph is calculated
- compileSections: an object of type SectionBuilder is instantiated (package network.graph.pathfinder) and sections are compiled

5.1.3 Package: network

This package does not contain any classes, only sub-packages as listed below.

5.1.3.1 Package: network.component

This package contains the following classes:

<u>Segment</u>

Objects of class Segment represent road segments in the network uniquely. The attributes that define a road segment are *roadNo*, *kmfrom*, *kmto* and *type*.

The important methods of class Segment are:

- calcTargetBF: method to calculate the target Blading Frequency for a segment
- calcAveRough: method to calculate the average roughness of a segment
- calcVOC: method to calculate the Vehicle Operating Cost of a segment for a given average roughness
- compareTo: method to sort a collection of segments based on the BladingSegments' nextBladeDate. If the nextBladeDates are equal, the next attribute is the AADT of the segments' TrafficSegments.

Whenever an object of type Segment is instantiated and the attribute *type* equals "G", indicating a gravel segment, corresponding objects of type BladingSegment, MaterialSegment and TrafficSegment are also instantiated. However, the objects of type BladingSegment, MaterialSegment and TrafficSegment cannot be instantiated if an object of type Segment with attribute type equal to "G" does not already exist. These four classes contain the methods and attributes necessary to define road segments in the network.

<u>BladingSegment</u>

The objects of class BladingSegment represent gravel segments that are bladed in the network. The attributes that are used to describe these objects are *segment*, *lastBladeDate*, *nextBladeDate*, *maxRoughness*, *targetRoughness*, *startRoughness*, *numOfBlades, productionRate and productiveHours*.

The important method of class BladingSegment is:

- calcBladingTime: method to calculate the time (to the nearest quarter hour) needed to blade the segment

<u>MaterialSegment</u>

Objects of type MaterialSegment represent the material and climate properties associated with the gravel segments. The material properties are stored in objects of type Material (*material* and *materialTRH20*), while the climate properties are stored in an object of type Climate. The attribute *segment* is the other important attribute of this class.

The important methods of class MaterialSegment are:

- setMaterialProperties: the material properties for the segment are calculated (based on the principle of weighted averages) from a set of database records that correspond to the *Roadno*, *kmfrom* and *kmto* attributes of the segment
- setMaterialPropertiesTRH20: the material properties are normalised to a maximum sieve size of 37.5 mm

<u>TrafficSegment</u>

Objects of type TrafficSegment represent the traffic properties associated with the gravel segments. The traffic properties are stored in an object of type Traffic (*traffic*), while the other important attribute is *segment*.

The important method of class TrafficSegment is:

- setTrafficProperties: traffic properties for the segment are calculated (based on the principle of weighted averages) from a set of database records that correspond to the *Roadno*, *kmfrom* and *kmto* attributes of the segment

Classes **Material**, **Traffic** and **Climate** extend class DataObject (package network.utils, section 5.1.3.4) and will not be elaborated upon here.

5.1.3.2 Package: network.graph

This package contains two classes (Vertex and Edge) and one sub-package (network.graph.pathfinder).

Classes Vertex and Edge provide the basic functionality needed for modelling an undirected graph.

<u>Vertex</u>
Objects of type Vertex represent vertices in an undirected graph. The important attribute of this class is a set containing the objects of type Edge (*edges*) connected to the vertex.

Class Vertex contains the following important methods:

- edges: this method returns a set containing the edges that are connected to the vertex
- removeEdge: this method removes a given edge from the set of edges attached to the vertex
- addEdge: this method adds an edge to the set of edges attached to the vertex

<u>Edge</u>
Objects of type Edges represent edges in an undirected graph. This class contains three important attributes, two objects of type Vertex (*v1* and *v2*) representing the vertices at the endpoints of the Edge; and an Object called *weight* representing the weight of the edge.

Class Edge contains the following important methods:

- connectedTo: this method returns true if the Edge is connected to the given Vertex, otherwise it returns false
- getVertex: this method returns the first vertex in the vertex set
- getVertex(Vertex v): this method returns the second vertex in the set, given the first vertex
- getWeight: this method returns the Object that are used to define the edge's weight

5.1.3.2.1 Package: network.graph.pathfinder

This package contains the classes used to calculate the routes in the network. The class used to compile the set of road sections in the graph is also contained in this package. The reason for this is that sections describe paths between the intersections in the network. The classes and their important methods are listed below.

Path

This class extends the Java class ArrayList and is used to describe the path (as a list of segments) one would follow to traverse the network from one vertex to another.

The important methods are:

- pathLength: method returns the length (in kilometres) of the path
- gravelLength: method returns the length (in kilometres) of the gravel segments contained in the path
- bladingTime: method returns the time (to the nearest quarter hour) needed to blade the gravel segments contained in the path
- isSimple: method returns true if all the segments in the path are unique (i.e. occur only once)
- isUnitPath: method returns true if the path contains only one element and if that element is equal to the unit element of the path algebra
- isEqualTo: method returns true if the path contains the same elements as the path it is being compared to and if the order in which the elements are contained are the same

Section

The class is used to describe a road section between intersections. A Section is defined as a continuous length of road between intersections in the network. Thus, a section may contain both sealed and unsealed road segments. The section is described by an object of type Path.

The important methods are:

- bladingTime: this method returns the time (to the nearest quarter hour) needed to blade the gravel segments contained in the section
- calcAgencycost: this method returns the cost incurred by the road agency to blade the gravel segments contained in the section
- calcAverageRQ: the average riding quality of the gravel segments contained in the section are calculated and returned
- calcVOC: this method calculates the VOC for the gravel segments in the section
- calcTTC: the TTC of the gravel segments in the section are calculated and returned

Sequence

This class extends the Java class ArrayList and it contains a sequence of sections that are to be bladed.

The important method for this class is:

- calcTTC: this method calculates and returns the TTC of the sequence of sections contained in it

SectionBuilder

This class is used to compile the set of sections contained in the network.

The important methods are:

- buildSections: this method compiles the set of sections contained in the network
- buildPath: this method is used to compile a path between two intersections
- isInterSection: this method returns true if a given vertex is an intersection. A vertex is an intersection if the size of the set of edges connected to it are not equal to two. If the size of the connected edge set is equal to one the vertex is a "dead-end" in the network, if it is more than two the vertex is an intersection.

ShortestSimplePathAlgebra

This class provides the implementation of the union and concatenation operations of the Shortest Simple Path Algebra used to calculate routes in the network. (This class is the implementation of the minimum simple path algebra described in section 4.9.1.)

The important methods are:

- concatenation: this method implements the concatenation operation
- union: this method implements the union operation
- shortestPath: this method is used to determine the shortest path, given two paths

ShortestSimplePathFinder

This class provides the functionality needed to calculate the Shortest Simple Paths from all the vertices in the network to all the other vertices in the network. Both the calculation methods for the full matrix and triangle method are implemented. (This class is the implementation of the calculation method for the closure of the elementary path set matrix described in section 4.10.)

The important methods are:

- compileAdjacencyMatrix: method to compile the adjacency matrix for the graph
- compileAdjacencyTriangle: method to compile the lower triangle of the adjacency matrix of the graph
- multiply: method to multiply two matrices using the ShortestSimplePathAlgebra
- multiplyTraingles: method to multiply to lower triangles using the ShortestSimplePathAlgebra
- equalMatrices: method to ascertain whether two matrices are equal

- equalTriangles: method to ascertain whether two lower triangles are equal

WgtLiteral

This is a utility class used for the calculation of routes in the network. The methods used are not important to the implementation of the scheduling algorithms.

5.1.3.3 Package: network.gui

This package contains one class, DrawPanel.

DrawPanel

This class provides a graphical user interface to the user for the purpose of drawing a line diagram of the network. Refer to section 5.2.1 for further details.

The important methods contained in class DrawPanel are:

- drawNode: this method lets the user draw nodes on the screen
- drawLink: this method lets the user draw links between nodes on the screen
- nameLink: this methods lets the user name the links between nodes on the screen
- toDBase: this method stores the network data in the database in the tables ***nodes*** and ***links***

5.1.3.4 Package: network.utils

This package contains three classes (DataObject, VehicleOperatingCost and BladingFrequenciesCalculator) and two sub-packages (network.utils.detmdl and network.utils.trgBF). The classes contained in this package and sub-packages are utility classes for the network, hence the package name.

DataObject

This class provides a general functionality for the storage of data. It contains a HashMap called *properties* and the methods needed to get the value given the key and set the value given the key and value.

VehicleOperatingCost

This class is an implementation of the Simplified HDM4 method for the calculation of Vehicle Operating Cost. This method has been in use at the PGWC since 2001. (Refer to [Burger, 2003] for more detail.) This class contains an attribute of type Segment for which the VOC is to be calculated.

The method of importance is:

- calcSegmentVOC: this method returns the VOC for the segment, given the average roughness

BladingFrequenciesCalculator

This class is used to calculate the recommended blading frequencies based on two deterioration models (TRH20 and HDM4) and two methods of blading frequency calculation (Target Roughness or Minimising TTC). For each segment in the model, the recommended blading frequencies (four in total) are calculated and stored in the database in the table BladingFreqs.

5.1.3.4.1 Package: network.utils.detmdl

This package contains the interface DeteriorationModel and two implementations of this interface, DeteriorateTRH20 and DeteriorateHDM4.

The implementing classes provide the methods for the calculation of time based deterioration based on the TRH20 and HDM4 deterioration models. The classes and methods will not be described in detail. Refer to section 5.2.2.1 for more detail on the use of the two deterioration models.

5.1.3.4.2 Package: network.utils.trgBF

This package contains the interface TargetBladingFrequency and four classes that implement this interface (TargetRoughnessBFTRH20, TargetRoughnessBFHDM4, MinimumTTCBFTRH20 and MinimumTTCBFHDM4).

The implementing classes provide the methods for the calculation of target blading frequencies based on the two approaches (target roughness, minimum TTC) and two deterioration models (TRH20, HDM4). The important method implemented in each of the implementing classes is:

- calcBF: method used to calculate the blading frequency

5.1.4 Package: schedule

This package contains two classes, NetworkRoughnessSchedule and RandomSectionSequenceSchedule, which are the implementations of the two scheduling algorithms. Also contained in this package is the sub-package schedule.utils.

NetworkRoughnessSchedule

This class implements the minimum network roughness algorithm. The full implementation will be explained in detail in section 5.2 of this chapter.

The important methods of this class are:

- initiateSegmentList: this method gets the set of gravel segments that have to be scheduled from an object of type Model
- setSegmentStartData: this method sets the start data (e.g. current condition, blading frequency) for each of the segments that have to be scheduled
- setStart: method sets the start date of the schedule
- setEnd: method sets the end date of the schedule
- initiateTicker: an object of type CalendarTicker is initiated with the date set to the start date
- setFirstSegmentInSequence: the first segment that has to be bladed is set at the first position in the blading sequence
- scheduleSegments: this method schedules the segments in the set until the date of the CalendarTicker object is equal to the end date of the schedule
- setNextSegmentInSequence: this method uses the compareTo method of class Segment to sort the set of segments in order to determine the next segment that needs to be bladed
- calcRouteToNextSegment: the route from the current segment to the next segment is determined by this method
- calcTravelTimeToNextSegment: this method calculates the travel time (to the nearest quarter hour) to the next segment in the sequence
- calcNetworkTTC: this method calculates the Total Transportation Cost for the network based on the schedule determined

<u>RandomSectionSequenceSchedule</u>

This class is the implementation of the network cost minimisation algorithm. The full implementation will be explained in detail in the section 5.2.

The important methods of this class are:

- initiateTicker: an object of type CalendarTicker is initiated in this method with the date set to the start date of the schedule
- setSegmentStartData: method sets the start data of the set of segments that have to be scheduled
- determineMinBladingInterval: this method determines the shortest blading interval of the segments that have to be scheduled
- buildRandomSequences: this method builds a number of random sequences (elements of the sequence are of type Section) and the sequence resulting in a minimum network TTC is stored
- networkTTC: this method calculates the network TTC
- optimiseMinTTCSequence: this method optimises the minimum TTC sequence by generating random sequences from the set of sections contained in the minimum TTC sequence

- switchOptimisedMinTTCSequence: this method switches the sections in the optimised minimum TTC sequence with sections in the set of sections not present in the sequence, if such a switch will lead to a lower network TTC

5.1.4.1 Package: schedule.utils

This package contains one class, CalendarTicker, which is described in more detail below.

<u>CalendarTicker</u>

This class provides the methods to calculate dates based on total hours worked. The methods are:

- addHours: this method adds the hours worked to the total hours worked. Method tick is then called.
- tick: this method calls method tack while the hours worked is higher than the productive hours in a day.
- tack: this method adds one day to the calendar (excluding Saturdays, Sundays and Public holidays) as long as the hours worked exceed the productive hours. (E.g. say the productive hours are 6.5 hours per day and that it takes 10 hours to blade a given road. Thus, one day will be added to the calendar and the hours worked on that day will equal 3.5 hours.)

5.1.5 Package: utils

This package contains a sub-package (utils.database) with a utility class in it (class DBConnection).

5.1.5.1 Package: utils.database

This package contains the class DBConnection that provides the methods needed to execute SQL queries and updates on the database.

Class DBConnection has the following important methods:

- openConnection: method to open the connection to the database
- closeConnection: method to close the connection to the database
- executeQuery: method to execute a query on the database
- executeUpdate: method to execute an update query on the database

5.2 *Runtime Model Implementation*

In this section the implementation of the runtime model is described. The processes and dependencies are described, as well as the calculation methods used. The purpose of this section is to give the reader an overview of the prototype application in order to understand how the classes described in section 5.1 are related and used to achieve the objectives of the study.

5.2.1 Network Definition

Conceptually, two networks are used during the scheduling of routine maintenance in the Maintenance Wards. These are the **Logical Network** and the **Physical Network**.

The *Logical Network* comprises the gravel segments of roads in the network. These are the sections that are used to determine the schedule sequence.

The *Physical Network* comprises all the road segments present in the network (both sealed and unsealed). This is the network that is used to determine routes between gravel segments/sections.

The user is required to define the physical road network in the maintenance ward under consideration. A graphical user interface is provided on which the user has to draw a line diagram of the network in three steps: In step one the user draws nodes, in step two he connects the nodes with links in step three he names the links. Since the links are uniquely identified by the attributes *roadNo*, *kmfrom*, *kmto* and *type* it follows that both the Logical and Physical network is defined in this process since *type* has value "S" for sealed roads and "G" for unsealed roads.

The links are named by selecting the appropriate segment definition (*roadNo*, *kmfrom*, *kmto* and *type)* from a drop-down list containing segment definitions on the graphical user interface. This drop-down list is populated from the database from a table that contains the segment definitions for the maintenance ward being drawn.

Nodes have to be defined at intersections, change of surface type, municipal boundaries or any other feature that the user deems to be a logical end or start point for segments. For example, if the user decides to split a long segment of gravel road into two segments of approximately equal length, he would define a node at the approximate centre point of the segment under consideration.

For the prototype application, the Swellendam-Infanta ward of the Overberg DM was used for testing the scheduling algorithms. Nodes were defined at the intersection of roads; where surface type changed; and at municipal boundaries. The physical network comprises National Roads and Provincial Trunk, Main and

Divisional Roads. The logical network comprises the gravel segments of the Provincial Trunk, Main and Divisional Roads contained in the physical network. The physical network contains gravel segments that form part of a different maintenance ward and these segments are defined as surface type "S" for this maintenance ward.

Refer to Figure 5-8 for a screen shot of the graphical user interface.

The network definition is stored in the database for later retrieval in two tables: ***nodes*** and ***links***.

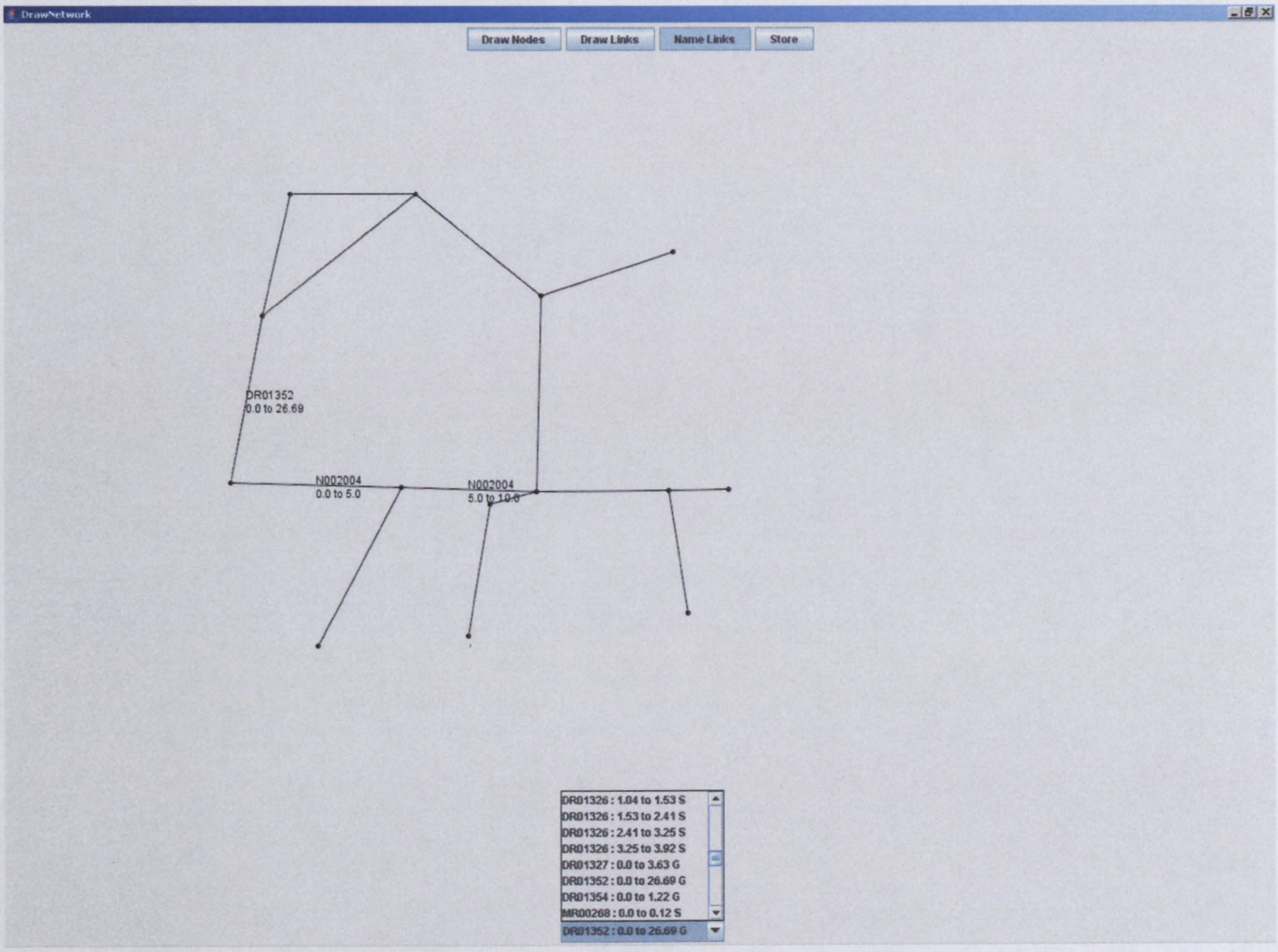


Figure 5-8: Screen shot of DrawPanel graphical user interface.

5.2.2 Calculate and Edit Blading Frequencies

5.2.2.1 Calculate Blading Frequencies

The scheduling algorithms implemented in the prototype application make use of blading frequencies during scheduling. These blading frequencies may be either calculated based on two methodologies as described below, or defined by the user.

For the two methodologies it is necessary to use deterioration models to model roughness deterioration with time. Two deterioration models were implemented in the system – the **TRH20** model and the **HDM4** model. These deterioration models will be described in more detail below.

Methodologies used

Two methodologies were implemented for the calculation of blading frequencies.

The one approach calculates the blading frequency needed to achieve a **target roughness**. This is done based on the material properties of the road, traffic characteristics and climate.

The other approach is based on a **cost optimisation**. The blading frequency that minimises the TTC for a given road is calculated.

For the cost optimisation method it is necessary to calculate the VOC given the average road roughness and traffic on the road. The method of calculation that is used is the **Simplified HDM4 methodology** for the calculation of Vehicle Operating Cost. This methodology has been in use at the PGWC since 2001. [Burger, 2003].

After calculation, the data is stored in the database in table ***BladingFreqs*** for later use.

<u>Target Roughness Blading Frequency</u>

The procedure to calculate the Blading Frequency (BF) based on the target roughness is iterative. The steps of the iteration process are detailed below after two approaches to this calculation method have been discussed.

Two approaches are possible when using this method to calculate the BF and the user of the system must understand the differences clearly when using this method.

Approach 1: The current roughness (riding quality) of the road is taken as the start roughness at the start of the iteration. The BF calculated in this way is the actual BF that will reduce the roughness (on average) to the target roughness. This approach is only used if the actual roughness is greater than the target roughness. This approach will give a higher BF than Approach 2, because it is necessary to bring the average roughness down to the target roughness.

Approach 2: The start roughness at the start of the iteration is set equal to the target roughness. The BF calculated in this way is the BF for "steady-state" maintenance. This gives the required BF, given the material properties, traffic and climate, necessary to obtain the average roughness equal to the target roughness.

Calculation process:

The calculation procedure outlined here is run through for each of the segments in the network.

1. Set the Target Roughness (TR). (E.g. 40 QI / 3.57 IRI)
2. Read the minimum number of blades per road from the segments table.
3. Set **n** = minimum number of blades per year.
4. Set start roughness.
    a. Start Roughness = Converted Roughness* OR (Approach 1)
    b. Start Roughness = Target Roughness (Approach 2)
5. Calculate the blading frequency.

*Converted Roughness: the roughness category of the segment at the start of the analysis cycle is converted to a QI or IRI value.

Loop 1 to Calculate the BF

For h = **n** to 1000

Loop 2 to Calculate Average Roughness for given ***h***

For j = 0 to **h**

Calculate Roughness before blading (RBBL) using chosen deterioration model

Calculate Roughness after blading (RABL) using chosen deterioration model

Calculate average Roughness (ave = (RBBL + RABL)/2)

Set value at position j of average matrix equal to ave. (average[j] = ave)

End of For

Calculate Average Roughness as follows:

Total = sum of values in average matrix

Average Roughness = Total/**h**

End of Loop 2

If (Average Roughness = TR)

Set BF = **h**

Break from Loop 1.

End of Loop1

An upper limit is set for the calculation (typically this may be 1000). If no answer is found, the upper limit is returned as the BF.

Cost Optimisation Blading Frequency

The cost referred to is the Total Transportation Cost (TTC). The Blading Frequency calculated in this method is the BF at which the TTC is a minimum.

Procedure:

A range of BF's is defined, starting at the minimum number of blades up to a maximum number in steps of 2. The maximum number of blades is input by the user.

For each BF in the range defined above, the following is calculated:

1. Average roughness for the year using chosen deterioration model
2. VOC for the average roughness
3. Agency Cost for the BF
4. TTC (sum of 2 and 3)

The TTC at each BF point is now available. It is possible to determine the turning point of the curve. The turning point is taken as the BF for the segment. (Refer to Figure 5-9.)

If the turning point is located between two points, the lower BF is taken as the BF for the segment.

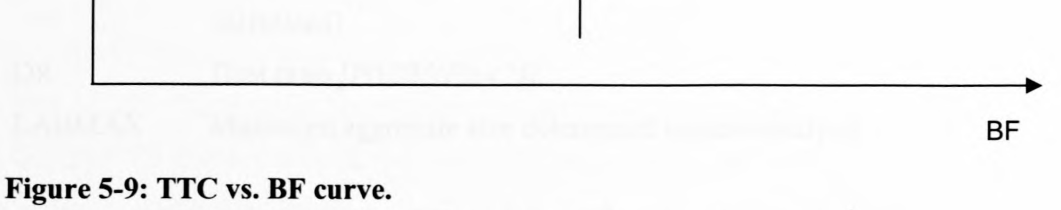


Figure 5-9: TTC vs. BF curve.

If the curve does not turn, determine the slope of the line between the minimum point and the maximum point. If the slope is greater than zero, the BF is set equal to the minimum point. Otherwise, the BF is set equal to the maximum point. (Refer to Figure 5-10.)

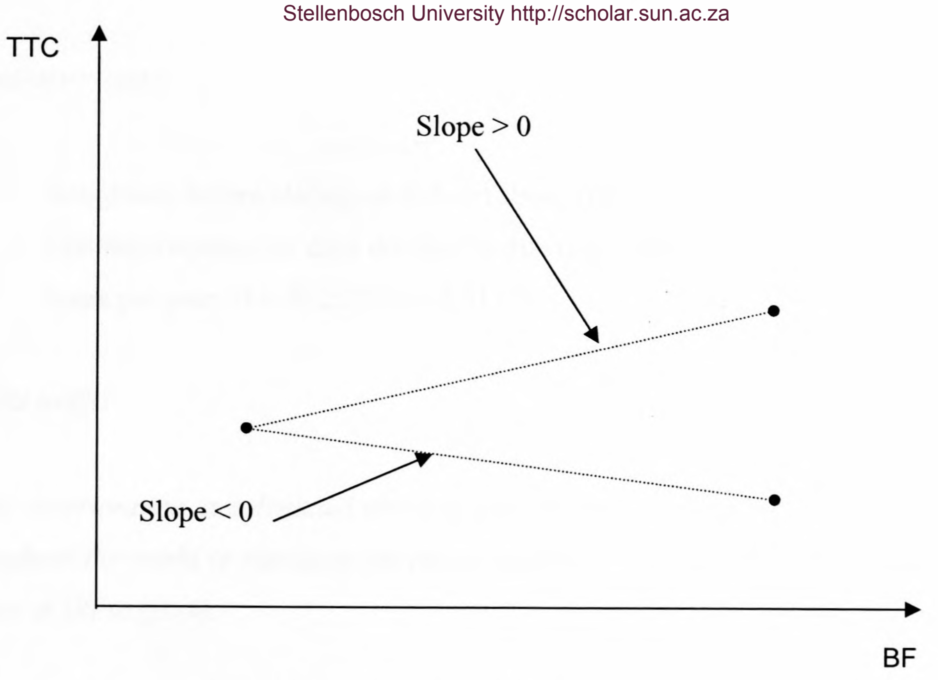


Figure 5-10: Determining BF if there is no turning point.

Deterioration modelling

TRH20 Deterioration Model

The equations for riding quality deterioration are [TRH20, 1990]:

Riding Quality Deterioration:

$$RABL = e^{(1.07+0.699\times\ln(RBBL)+0.0004\times ADT-0.13\times DR+0.0019\times LABMAX)}$$

RABL	Roughness after blading [QI (count/km)]
RBBL	Roughness before blading [QI] (Typical start = 45 QI, thereafter as calculated)
DR	Dust ratio *[P0.075/P0.425]*
LABMAX	Maximum aggregate size determined in sieve analysis

$$\ln(\Delta R) = -13.8188 + 0.00022\times PF + 6.34\times10^{-2}\times S1 + 0.137047\times P26$$
$$+ 0.000345\times ADT\times N + GM\times(6.4194 - 0.0634\times P26)$$

ln(ΔR)	Natural logarithm of change in roughness [ln(QI)]
S1	variable for season (1 for dry season, 0 for wet season)
GM	Grading modulus $\left[\frac{300-(P2.000+P0.425+P0.075)}{100}\right]$

$$RBBL = e^{(\ln(RABL) + D \times \ln(\Delta R))}$$

RBBL	Roughness before blading as defined above [QI]
D	Blading frequency in days divided by 100 (e.g. blading frequency = 4 times per year: D = 91.25/100 = 0.9125)

Converting IRI to QI:

Riding quality deterioration as calculated above is done with QI as input. However, the IRI is the standard that is used throughout the world to represent pavement roughness. For this reason an equation is given below for the conversion of IRI to QI (6).

$$QI = 14 \times IRI - 10$$

IRI	*International Roughness Index [m/km]*

The riding quality deterioration is calculated as follows:

1. The change in roughness [ln(ΔR)] is calculated.
2. The roughness before blading (RBBL) is calculated at the end of the blading cycle.
   a. For first blading cycle, the RABL in the equation is taken as either target roughness or actual roughness (converted to QI value).
   b. For subsequent blading cycles, RABL in the equation is taken as the previously calculated RABL value.
3. The roughness after blading (RABL) is then calculated with the necessary inputs.

HDM4 Deterioration Model

Background

The HDM4 deterioration model for unsealed roads is based on the model used in HDM-III. The model was designed for engineered unpaved roads. These are roads that have controlled alignment, formation width, cross-section profile and drainage.

The roads of the PGWC gravel network conform to the definition of engineered unsealed road. Therefore, the HDM4 unsealed road deterioration model is applicable to the PGWC gravel road network.

Deterioration of unsealed roads is characterised primarily by roughness deterioration and material loss from the surfacing. Furthermore, periodic blading is usually undertaken on a regular basis, either seasonally or frequently enough to keep the roughness within tolerable limits.

In HDM4, average roughness during an analysis year is computed as a function of the roughness at the beginning of the year and of material, traffic, geometry, climate (represented by rainfall parameter) and the specified blading frequency. Over a period of time (the period is dependent on traffic volume and blading frequency) the annual average roughness tends towards a long-term average roughness that may also be computed.

An unpaved road is considered to be a two-layer pavement system, comprised by a gravel surfacing and a subgrade layer. A gravel road has both layers, while an earth road has only the subgrade layer.

Deterioration is predicted based on the properties of the surfacing layer, whether the layer is gravel or subgrade. The user must specify the physical properties of the gravel or subgrade layers.

Model Parameters

The model parameters used are summarised in Table 5-2. The parameters may be differentiated as:

- Material
- Traffic
- Road Geometry
- Environment: Climate and Drainage

Material Properties

The material properties needed as input are listed in Table 5-2.

Traffic

Traffic parameters are ADL, ADH and AADT (refer to Table 5-2).

AADT is related to ADL and ADH as follows:

$$AADT = ADL + ADH$$

Table 5-2: Deterioration model parameters for HDM4 model. [After Table C4.2, p. C4-4, Volume Four, HDM4 documentation]

Type	Variable	Definition
Material	D95	Maximum particle size of the material, defined as the sieve size through which 95% of the material passes [mm]
	MGD	Dust ratio on material gradation
	MG	Slope of mean material gradation
	PI	Plasticity index of the material [%]
	P075	Amount of material passing the 0.075 mm sieve [% by mass]
	P425	Amount of material passing the 0.425 mm sieve [% by mass]
	P02	Amount of material passing the 2.0 mm sieve [% by mass]
	QIMIN	Minimum roughness of material (calculated or specified by user) [QI]
	QIMAX	Maximum roughness of material (calculated or specified by user) [QI]
Traffic	ADH	Average daily heavy vehicle traffic in both directions [veh/day]
	ADL	Average daily light vehicle traffic in both directions [veh/day]
	AADT	Annual average daily traffic in both directions [veh/day]
Geometry	C	Average horizontal curvature of the road [deg/km]
	RF	Average absolute rise plus fall of the road [m/km] *RF may be estimated as:* RF = 10*(average absolute gradient as a percentage)
Climate	MMP	Mean monthly precipitation [mm/month]

Road Geometry

Road geometry is represented by the horizontal curvature, C, and the absolute average rise plus fall, RF.

Roughness progression is influenced by both these parameters.

Environment: Climate and Drainage

Climate

Climate is represented by the parameter MMP – the mean monthly precipitation.

Drainage

Drainage does not have an explicit parameter that represents it in the model. However, it is represented by the fact that the model was derived for unpaved roads with moderate to good cross-sectional geometry. This is in general true for the roads in the PGWC network.

It has to be stressed that the model does not apply to roads with "inverted" geometry, i.e. roads that act as drainage canals for stormwater. (Refer to Figure 5-11.)

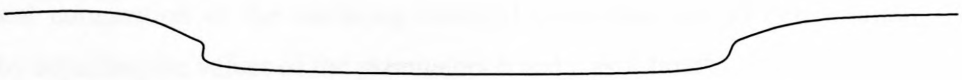


Figure 5-11: "Inverted" road geometry.

Road Roughness

Model Form

Roughness is constrained to a high upper limit by a convex function in which the rate of progression decreases linearly with roughness to zero at QIMAX.

Rate of progression

Rate of progression is given by the equation below:

$$QITG_2 = QIMAX - b*(QIMAX - QITG_1)$$

$QITG_1$, $QITG_2$ roughness at times TG_1, TG_2

QIMAX maximum roughness of the material

b model parameter

The model parameter, b, is given by:

$$b = \exp\left[c*(TG_2 - TG_1)\right] = \exp\left[\frac{365}{n}*c\right]$$

$$c = -\left(0.461 + 0.0174*ADL + 0.0114*ADH - 2.87*AADT*MMP*10^{-5}\right)*10^{-3}$$

n blading frequency

QIMAX is calculated as follows:

$$QIMAX = \max\left[279 - 421*(0.5 - MGD)^2 + 0.22*C - 9.93*RF*MMP*10^{-3}, 150\right]$$

C horizontal curvature

RF absolute average rise plus fall

If P425 = 0

$$MGD = 1$$

Else, if P425 > 0

$$MGD = \frac{P075}{P425}$$

Effect of compaction

Mechanical compaction of the surfacing material slows the rate of deterioration. This effect is taken into account by adjusting the values of the parameters b and c as follows:

$$c' = c * \min\left[1, 0.25 * t * \max\left(1, n^{0.33}\right)\right]$$

$$b' = \exp\left[\frac{365}{n} * c'\right]$$

t time in years since construction/regravelling

Effect of Blading

The roughness after blading is given by the following equation:

$$QI_{ab} = QIMIN + a * \left(QI_{bb} - QIMIN\right)$$

$$a = 0.553 + 0.23 * MGD$$

QIab roughness after blading

QIbb roughness before blading

The minimum roughness of the material is given by the following equation:

$$QIMIN = \max\left\{10, \min\left[100, 4.69 * D95 * \left(1 - 2.78 * MG\right)\right]\right\}$$

D95 maximum particle size

The parameter MG is given by:

$$MG = \min\left(MGM, 1 - MGM, 0.36\right)$$

$$MGM = \frac{MG075 + MG425 + MG02}{3}$$

If D95 > 0.4

$$MG075 = \frac{\ln\left(\frac{P075}{95}\right)}{\ln\left(\frac{0.075}{D95}\right)}$$

else

$$MG075 = 0.3$$

If D95 > 1.0

$$MG425 = \frac{\ln\left(\frac{P425}{95}\right)}{\ln\left(\frac{0.425}{D95}\right)}$$

else

$$MG425 = 0.3$$

If D95 > 4.0

$$MG02 = \frac{\ln\left(\frac{P02}{95}\right)}{\ln\left(\frac{2.0}{D95}\right)}$$

else

$$MG02 = MG425$$

Average roughness during analysis year

The average roughness for an analysis year may be calculated at the end of the year or during the year. The equations for the first case, i.e. end of year, will be given in this document.

If $(t*n)\geq 1$:

$$QI_{avg} = QIMAX*(1-y)+\left(\frac{y*NS}{n}\right)$$

$$y=\left(\frac{n}{365}\right)*\left(\frac{b-1}{c}\right)$$

$$NS=\frac{\left\{n*k+\left[1-(a*b)^{n}\right]*QI_{a}-\frac{k*\left[1-(a*b)^{n}\right]}{(1-a*b)}\right\}}{(1-a*b)}$$

$$k=(1-a)*QIMIN+a*(1-b)*QIMAX$$

QI_a roughness at start of the analysis year

Roughness cycle steady state

When blading is performed at constant time intervals, roughness change eventually leads to a steady state. The steady state is characterised by a saw-tooth pattern of high and low roughness values, with the roughness increasing linearly between the extremes.

The high and low roughness values are given by:

$$QIH=\frac{\left[QIMAX*(1-b)+QIMIN*(1-a)*b\right]}{(1-a*b)}$$

$$QIL=\frac{\left[QIMIN*(1-a)+QIMAX*a*(1-b)\right]}{(1-a*b)}$$

The average roughness value will approach a long term average, QI_{lta}. This is given by:

$$QI_{lta}=QIMAX+(1-a)*(1-b)*\frac{(QIMAX-QIMIN)}{\left[(1-a*b)*\ln b\right]}$$

5.2.2.2 Edit Blading Frequencies

It has been found that the use of the TRH20 and HDM4 deterioration models result in unrealistic blading frequencies. This may be due to a number of factors and sensitivity of the deterioration models to these factors. Previous work [Paige-Green, 1995] has shown that the TRH20 model is suited to the conditions under which it was developed. However, the transferability of this model is not good and it was recommended that this model be calibrated for use in specific areas [Paige-Green, 1991].

It was not within the scope of this study to determine the factors and sensitivities of the models. Therefore, the user has to manually enter the blading frequency that the system should use in subsequent calculations. The user defined blading frequencies are stored in table ***BladingFreqs*** of the database for later use.

5.2.3 Define Current Condition

Current condition data are very important if realistic schedules are to be determined. This data is used in both scheduling algorithms and will be clarified in the next section. This data is stored in the database in table ***CurrentCondition***.

5.2.4 Determine Schedule

5.2.4.1 Initiate data sets

Before scheduling can be commenced, it is necessary to initiate the data sets that will be used in the process. This is explained in more detail below. The data sets that are used are contained in an object of type Model. Thus, a Model object is instantiated at run-time to provide the necessary data sets used during scheduling.

Initiate set of gravel segments

From the network definition the set of roads where the Surface Type equals "G", i.e. the set of gravel road segments, is extracted from the set of roads that comprise the ward road network. This is the set of roads for which a schedule is to be determined.

The start data, i.e. current condition, next blade date, productive hours, production rate, etc., is set/calculated for the segments in the set. This start data form the basis from which the schedule is calculated. The blading frequency used is the one input by the user.

Initiate Network and Calculate Shortest Routes between Vertices

At the same time when the set of gravel roads is compiled, the components needed to define a Network are also initialised. The components that are needed are the vertices and the edges between vertices. The edges are weighted by the segment that it represents.

The algorithms used in the prototype application require **the shortest routes** between all vertices for the calculation of travel time between edges (segments). It is possible to define a minimum simple path algebra for this problem in order to calculate these routes once and use the result matrix multiple times (Refer to Chapter 4, sections 4.10.2 and 4.10.3). The result matrix contains all the shortest routes between every vertex in the graph.

5.2.4.2 Roughness Deterioration Model used during Scheduling

As mentioned in section 5.2.2.2, it was found that the calculated blading frequencies are unrealistic. Because of this, it was decided to use a straight line model for roughness deterioration with time. (Refer to Figure 5-12.)

HDM4 uses a similar approach for "steady state" deterioration (refer to section 0). Steady state deterioration may be defined as the state reached when blading is performed at approximately constant intervals. It is reasonable to assume that steady-state deterioration is present in the network since blading occurs at constant intervals. This type of deterioration is typically visualised by a "saw-tooth" graph where the peaks depict the maximum roughness before blading and the dales depict the roughness after blading.

The blading interval is calculated as the value (rounded to the nearest integer) of:

$$\frac{365}{BF}$$

where BF is the blading frequency.

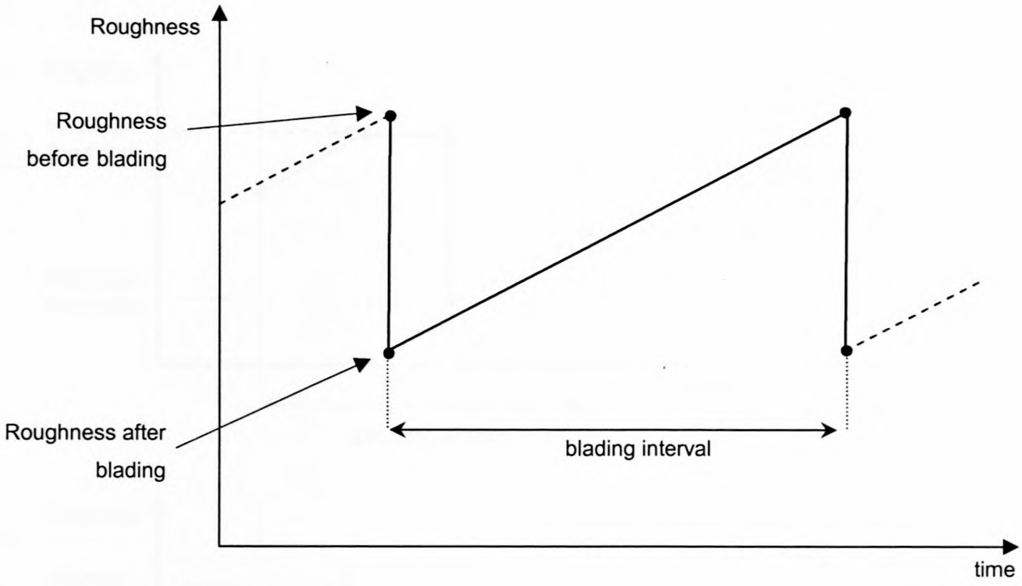


Figure 5-12: Straight line deterioration model.

Average Roughness Values

Segments are either bladed or not bladed during the schedule time window (scheduling window). It is required that the average roughness can be calculated during the scheduling window, if the Total Transportation Cost has to be determined.

For each of the two blading cases (*bladed* or *not bladed*) during the scheduling window, two more cases can be defined. These cases are dependent on the start roughness value, as determined based on the user's assessment of the current condition. The four cases are summarised in Figure 5-13 and Figure 5-14.

For each case, the average roughness is calculated as the weighted average during the scheduling window, i.e. the area under the curve.

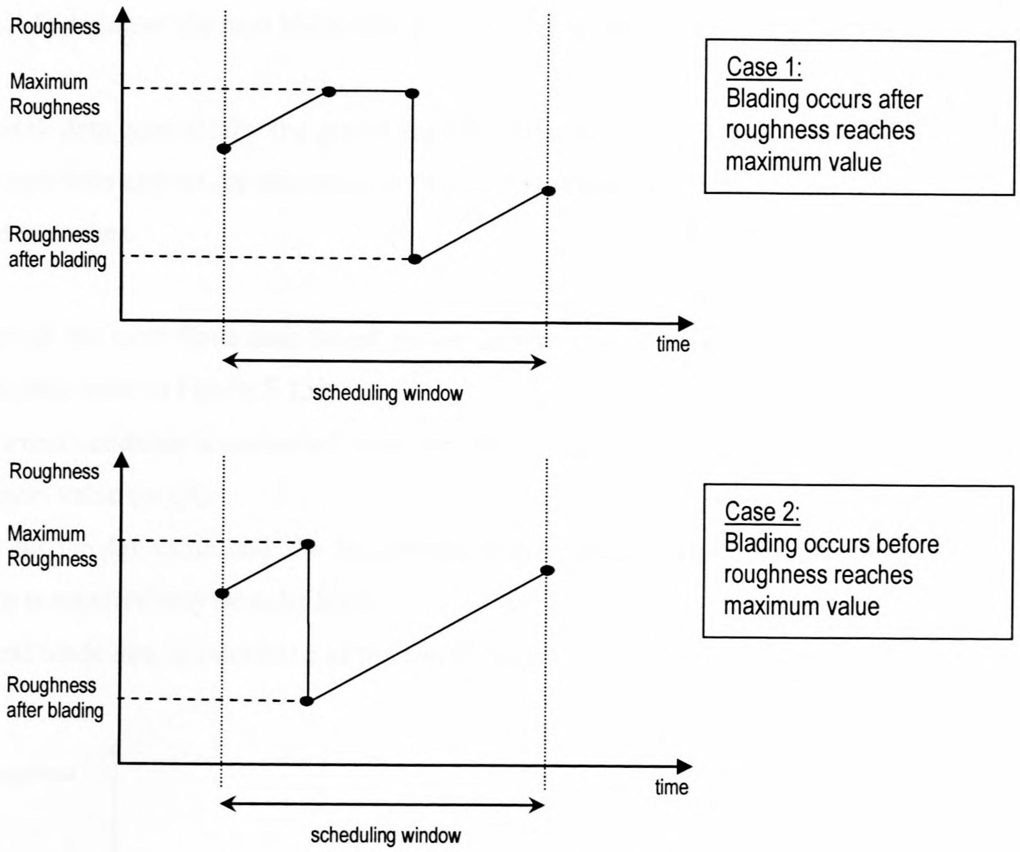


Figure 5-13: Cases where segment is bladed during scheduling window.

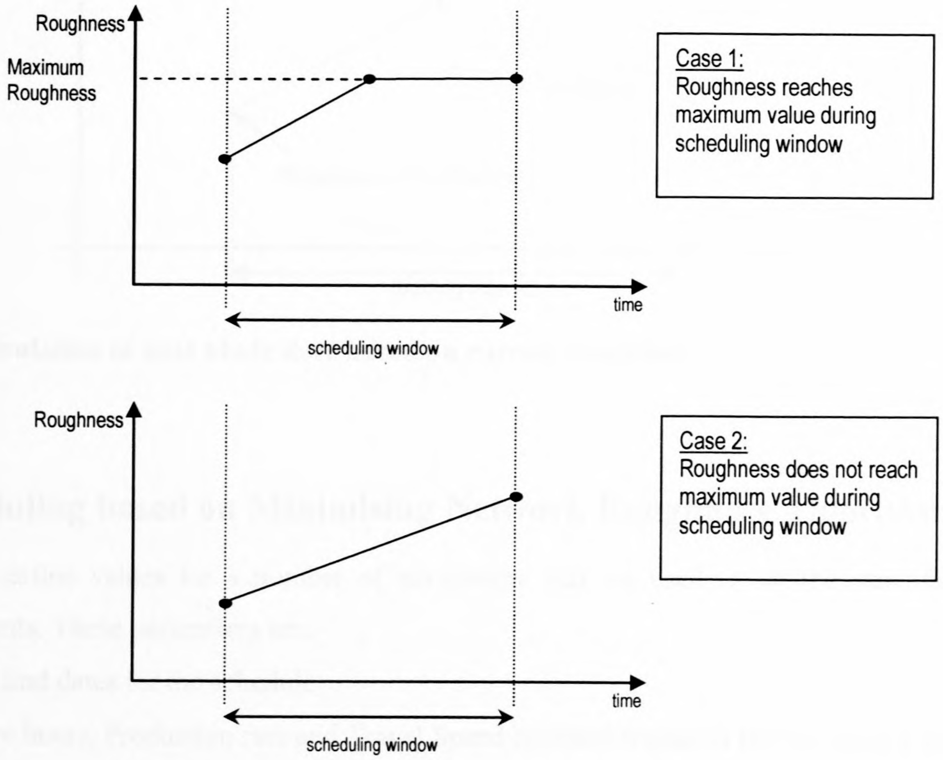


Figure 5-14: Cases where segment is not bladed during scheduling window.

Calculation of next blade date

As was stated elsewhere, the network roughness algorithm uses the next blade date when determining a schedule. The calculation of the next blade date based on linear deterioration is described in this section.

When the network data, specifically the gravel segment data, are initiated the current condition of the segment is read from the data base and set for the segment object. Also read from the data base are the blading frequencies of the different segments.

The calculation of the next blade date based on the current condition and blading frequency for a segment is outlined below (also refer to Figure 5-15):

- The current condition is converted from the category, e.g. VG (Very Good) or P (Poor) to the applicable roughness value (in QI)
- Based on the QI value and the Roughness Before Blade value, the time in days remaining before blading is required may be calculated
- The next blade date is calculated as the current date plus the number of days remaining

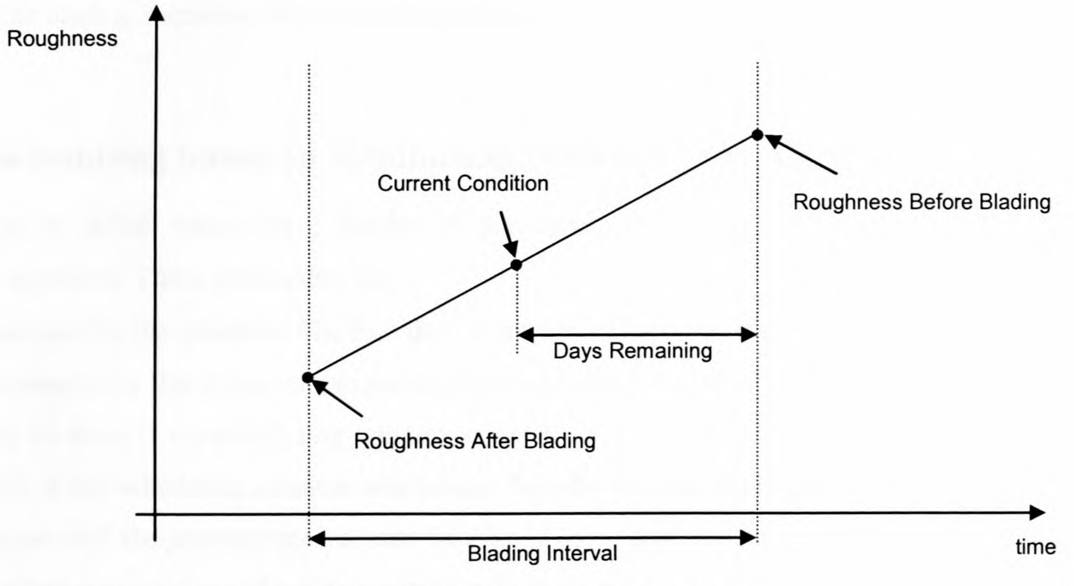


Figure 5-15: Calculation of next blade date based on current condition.

5.2.4.3 Scheduling based on Minimising Network Roughness Algorithm

The user has to define values for a number of parameters that are used to set the start data of the set of scheduling segments. These parameters are:

- Start and End dates for the schedule
- Productive hours, Production rate and Travel Speed between segments for the blading team
- The Start Segment

The results from the scheduling are two sets of data, i.e. the blading sequence and shortest routes between the segments of the sequence, and the TTC for the blading sequence.

The start segment is set as the first segment in the blading sequence. Scheduling is performed as follows:

1. For the last segment in the blading sequence:
    a. blading time is calculated and added to the CalendarTicker
    b. the segment's last blade date is set to the CalendarTicker's date
    c. the segment's next blade date is calculated
    d. the next segment to be bladed is set as follows:
        i. the set of scheduling segments are sorted based on the next blade date. If two segments' next blade date is equal, the AADT of the segments is used as second criterion
        ii. the first segment of the sorted set is set as the next segment in the blading sequence
    e. the shortest route to the next segment is obtained from the routes matrix (there are four possible routes between two segments and of these the shortest length route is chosen)
    f. the travel time between the two segments is calculated and added to the CalendarTicker

This sequence of calculation steps is repeated until the CalendarTicker's date is equal to the end date of the schedule. The blading sequence, set of routes and schedule TTC is reported.

5.2.4.4 Scheduling based on Minimising Network Cost Algorithm

The user has to define values for a number of parameters that are used to set the start data of the set of scheduling segments. These parameters are:

- Start date for the schedule. The End date is calculated based on the shortest blading interval in the ward. The reason for this is that simple paths, based on some measure of time, are to be calculated and this can only be done if the scheduling window is equal to or shorter than the shortest blading interval in the ward. If the scheduling window was longer than the shortest blading interval, one has to allow that the section with the shortest interval may be bladed more than once during the scheduling window and the resultant sequence would not be a simple path anymore.
- Productive hours, Production rate and Travel Speed between segments for the blading team

For this schedule a large number of section sequences are randomly generated. The length (measured in days) needed to blade any of the sequences may, in principle, not exceed length of the scheduling window. However, some leniency in length is allowed by accepting sequences with length greater than or equal to 80% of the scheduling interval and length less than or equal to 110% of the scheduling interval. Thus:

$$0.8*(\text{scheduling interval}) \leq (\text{sequence length}) \leq 1.1*(\text{scheduling interval})$$

The sections used in the generation of the random sequences are only those that contain gravel segments. The sequences are generated until the number of calculations is equal to the number set by the user, e.g. 2 million calculations. It is not necessarily true that 2 million calculations will result in 2 million sequences, since a sequence may well be generated twice in such a large number of calculations. The sequence that results in the minimum network TTC is further optimised in two steps as described below and then reported as the blading sequence. The network TTC for this sequence is also reported.

Reason for random generation of sequences:

For the Swellendam-Infanta maintenance ward, the number of sections containing gravel segments is 30. It is possible to calculate all simple routes in the ward. However, the number of possible routes is equal to 30!, i.e. 30 factorial, which is a number of order 10^{32} (i.e. 100 000 000 000 000 000 000 000 000 000 000). To calculate this large amount of routes will require a super computer. It is true that only the routes that comply with the scheduling interval criterion need to be considered, but all the routes have to be calculated first before the length criterion can be applied. Therefore, it was decided to generate sequences randomly that comply with the length criterion.

Further optimisation of the minimum TTC sequence:

The minimum sequence that was randomly generated may be further optimised by swapping around the order of the sections comprising the minimum sequence. This is done during the first optimisation step described below.

It may also happen that there may be in the set of sections not included in the minimum sequence, other individual sections of approximately equal blading time that may be swapped with the individual sections in the minimum sequence for which the inclusion of the firstmentioned section may result in a more favourable network TTC for the ward. In effect the "bad apples are removed from the basket" and replaced by "good apples". This optimisation is performed in the second step as described below.

First step: It is reasonable to assume that the sequence resulting in the minimum network TTC may well be further optimised by using the sequence as the set of sections and generating random sequences from this set. In this case the length criterion need not be applied, since it follows from the first calculation that the sequence length complies with the length criterion. The result of this optimisation step is then optimised again in the second step.

Second step: Each of the sections contained in the minimum sequence is "switched" with a section of approximate equal blading time not contained in the sequence. The network TTC for the resulting sequence is then calculated and if the network TTC for the "switched" sequence is lower than the minimum network TTC, the "switched" sequence is taken as the minimum network TTC sequence.

The scheduling steps may be summarised as:

1. Random generation of a large number of section sequences, and selecting the minimum TTC sequence
2. Optimising the sequence of the minimum TTC sequence.
3. Switching of sections in the minimum sequence with sections not contained in the sequence, aiming to further minimise the network TTC.

6 Evaluation of the scheduling algorithms

6.1 Introduction

Two scheduling algorithms for optimising routine maintenance in a maintenance ward of a gravel road network were described in Chapter 3. These algorithms were implemented in a pilot application as described in Chapter 5. In this chapter the performance of the two algorithms is evaluated insofar as it is possible. The issues addressed are:

- The performance of the 2 algorithms is compared on the basis of the ward network TTC of their resulting schedules. TTC is not used to determine the schedule of the minimum network roughness algorithm, but a ward TTC of the resulting schedule can be determined and compared to the TTC of the minimum network cost algorithm.
- The minimum network cost algorithm is based on the random generation of blading sequences. This raises the problem of how many sequences have to be generated before an acceptable minimum sequence will be found. A numeric approach is followed in which the results of a number of sample sizes are determined and observations regarding convergence and the required sample size are made.
- The issue of required computer capacity is also addressed based on a generally available medium range notebook computer.

The Swellendam/Infanta maintenance ward of the Overberg DM in the Western Cape was modelled and its field data were used in the evaluation. The functioning of the scheduling algorithms is not influenced by parameters of the specific ward to which they are applied, since:

- the algorithms were developed based on the constraints and variables identified in the interviews with the different DMs in the Western Cape (Chapters 2 and 3) and thus they are applicable to all the DMs
- the algorithms were developed to be used on a single maintenance ward basis
- the shortest simple path algebra used in the determination of routes in the network was developed in general and adapted to be used in the pilot application and this path algebra does not need any special data which may vary from ward to ward

Consequently, the algorithms may be evaluated on the basis of a single ward of which the location in the Western Cape does not matter.

6.2 Network Data used for evaluation

The scheduling algorithms require that some data sets be defined by the user. These data sets are reported in this section. Table 6-1 lists the gravel segments that were defined for this ward. Also contained in the table are the blading frequencies (BF) and the current condition values that were used for scheduling.

Table 6-1: List of gravel segments in Swellendam-Infanta ward.

RoadNo	kmfrom	kmto	Type	BF (per year)	Current Condition
DR01251	0	31.46	G	5	G
DR01250	0	13.23	G	7	P
MR00268	0.12	1.98	G	7	G
MR00268	1.98	21.95	G	4	F
MR00268	21.95	27.6	G	3	VP
MR00268	27.6	37.1	G	10	G
MR00268	37.1	38.46	G	6	P
MR00268	38.46	41.73	G	6	F
MR00268	43.02	68.82	G	3	G
DR01263	39.23	40.51	G	3	P
DR01273	0	26.33	G	5	VP
DR01277	2.7	13.5	G	8	P
DR01277	13.5	27.61	G	4	F
DR01314	0	2.56	G	5	F
DR01326	0.08	1.04	G	9	VG
DR01318	0.19	3.65	G	7	F
DR01321	0.06	5	G	8	P
DR01319	0	3.28	G	12	F
DR01325	0.35	1.84	G	4	P
DR01325	1.84	10.12	G	3	VG
DR01325	10.12	12.84	G	5	F
DR01327	0	3.63	G	7	F
DR01274	28.12	29.2	G	7	G
MR00270	0.08	9.29	G	9	VG
MR00270	9.29	11.35	G	7	G
MR00270	11.35	20.1	G	10	VG
DR01304	0	7.22	G	8	F
DR01304	7.22	10.08	G	7	G
DR01304	14.97	24.81	G	5	G
DR01352	0	26.69	G	8	F
DR01354	0	1.22	G	5	VP

Table 6-2 lists the sealed segments that were defined for the Swellendam-Infanta ward. The segment definitions were compiled from tables contained in the PGWC's Road Network Information System (RNIS). However, the RNIS tables do not contain data on the National Roads, hence the unusual stake values for the "N" roads in the table.

Table 6-2: List of sealed segments in Swellendam-Infanta ward.

RoadNo	kmfrom	kmto	Type
MR00268	0	0.12	S
MR00268	41.73	43.02	S
DR01277	0	2.7	S
DR01324	0	0.96	S
DR01326	0	0.08	S
DR01326	1.04	1.53	S
DR01326	1.53	2.41	S
DR01326	2.41	3.25	S
DR01326	3.25	3.92	S
DR01318	0	0.19	S
DR01321	0	0.06	S
DR01325	0	0.24	S
DR01325	0.24	0.35	S
MR00270	0	0.08	S
DR01304	10.08	14.97	S
DR01304	24.81	25.96	S
TR03104	0	2.71	S
TR03103	40.39	62.44	S
TR03201	32.73	44.35	S
TR03201	44.35	45.15	S
TR06501	0	10.65	S
TR06501	10.65	32.62	S
N002004	0	5	S
N002004	5	10	S
N002004	10	15	S
N002005	0	4	S
N002005	4	7	S
N002005	7	9	S
N002005	9	11	S
N002005	11	15	S
N002005	15	20	S

6.3 Evaluation of algorithms

6.3.1 Minimise network roughness algorithm

The use of this algorithm with the parameter values as listed in Table 6-1 results in the sequence listed in Sequence 6-1. The schedule was determined for the period 1 November 2004 to 30 November 2004 (one month) and start segment was chosen as DR01250 0.0 13.23. This start segment was picked arbitrarily. Because of the manner in which the sequence of segments is determined (refer to section 3.2), choosing a different start segment would not influence the resulting sequence. Thus, choosing any other start segment would still result in a sequence of segments similar, except for the first segment in the sequence, to the one reported in Sequence 6-1.

DR01250 0.0 13.23 → DR01274 28.12 29.2 → MR00270 0.08 9.29 → DR01314 0.0 2.56 → DR01318 0.19 3.65 → DR01325 1.84 10.12 → DR01277 2.7 13.5 → DR01251 0.0 31.46 → DR01304 7.22 10.08

Sequence 6-1: Minimise Network Roughness Algorithm blading sequence.

Since it is possible to calculate the average riding quality for the segments in the ward, it is possible to calculate the TTC for the ward. The use of this algorithm gives a network TTC value of **R 2 714 312.41**.

Use of this algorithm returns a blading schedule within a matter of milliseconds using the notebook computer that was available to test it.

6.3.2 Minimising Network Cost Algorithm

The network cost algorithm determines sequences randomly. If enough sequences are considered, it may be assumed that the sequence resulting in the minimum network TTC determined with the algorithm will be close to the sequence resulting in the absolute minimum TTC that may be achieved in the network. Furthermore, it is also logical to assume that the higher the number of sequences generated, the closer the minimum sequence generated will tend towards the absolute minimum sequence. However, the number of calculations needed to achieve the minimum sequence may be high or it may be achieved quite quickly. For these reasons three (3) sequences were generated for four (4) calculation limit values, i.e. where the number of calculations were limited to 500 000, 1 000 000, 1 500 000 and 2 000 000.

6.3.2.1 Convergence of network TTC

This section contains the convergence results of the network TTC based on the minimum TTC sequences. The results are contained in Table 6-3 and Figure 6-1 – based on calculation runs depicted in Figure 6-2 to Figure 6-5. The vertical scale of Figure 6-1 has been chosen on purpose to highlight the little difference in Average Minimum TTC obtained for the different calculation limit values.

Table 6-3: Average minimum network TTC versus number of calculations.

No. of Calculations	Calculation Run			Average Minimum TTC	Standard Deviation (COV [%])
	1	2	3		
500 000	R 2 434 808.63	R 2 447 296.96	R 2 444 841.88	R 2 442 315.16	R 6 615.72 (0.27)
1 000 000	R 2 442 622.99	R 2 439 964.75	R 2 439 156.62	R 2 440 580.78	R 1 814.17 (0.07)
1 500 000	R 2 433 635.69	R 2 432 393.59	R 2 454 653.23	R 2 440 227.17	R 12 508.79 (0.51)
2 000 000	R 2 433 541.27	R 2 440 026.44	R 2 433 025.67	R 2 435 530.79	R 3 901.91 (0.16)

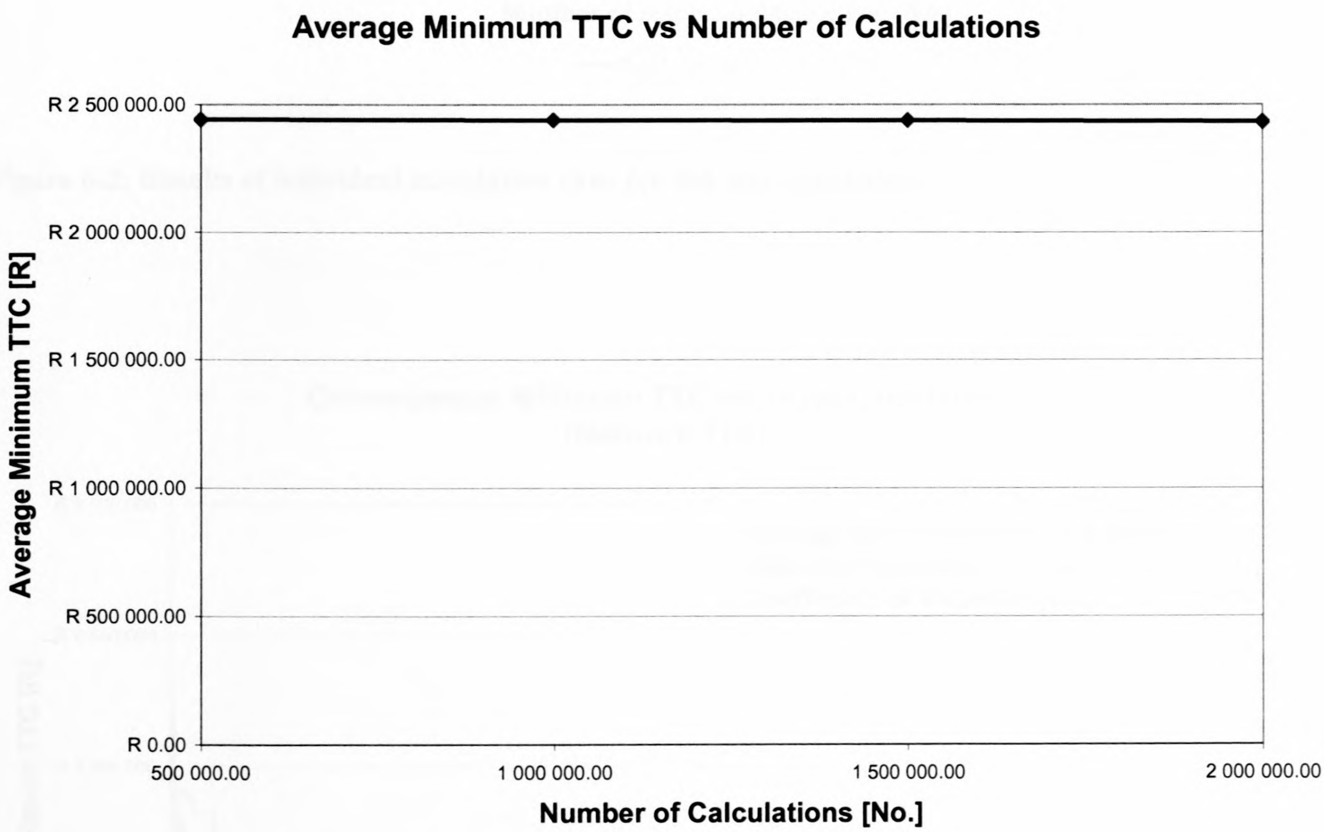


Figure 6-1: Average minimum network TTC versus the number of calculations.

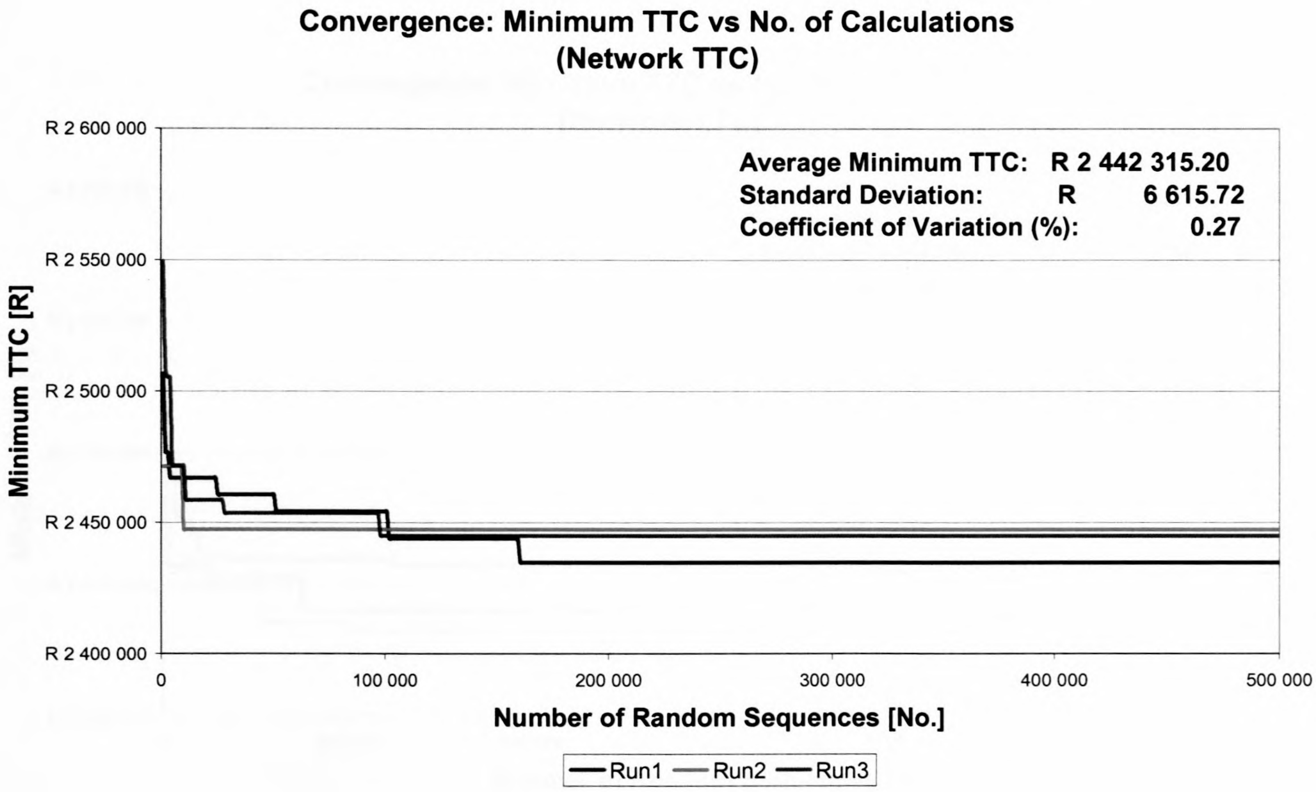


Figure 6-2: Results of individual calculation runs for 500 000 calculations.

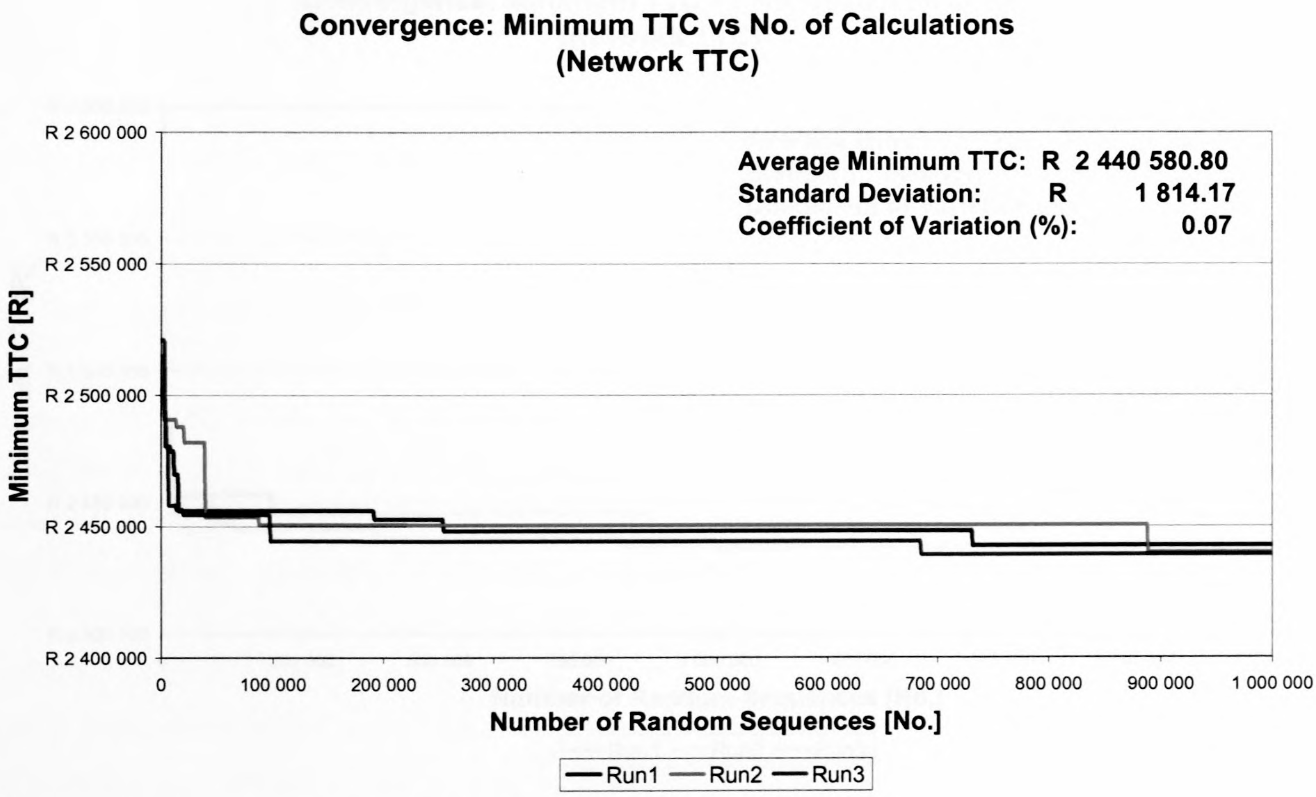


Figure 6-3: Results of individual calculation runs for 1 000 000 calculations.

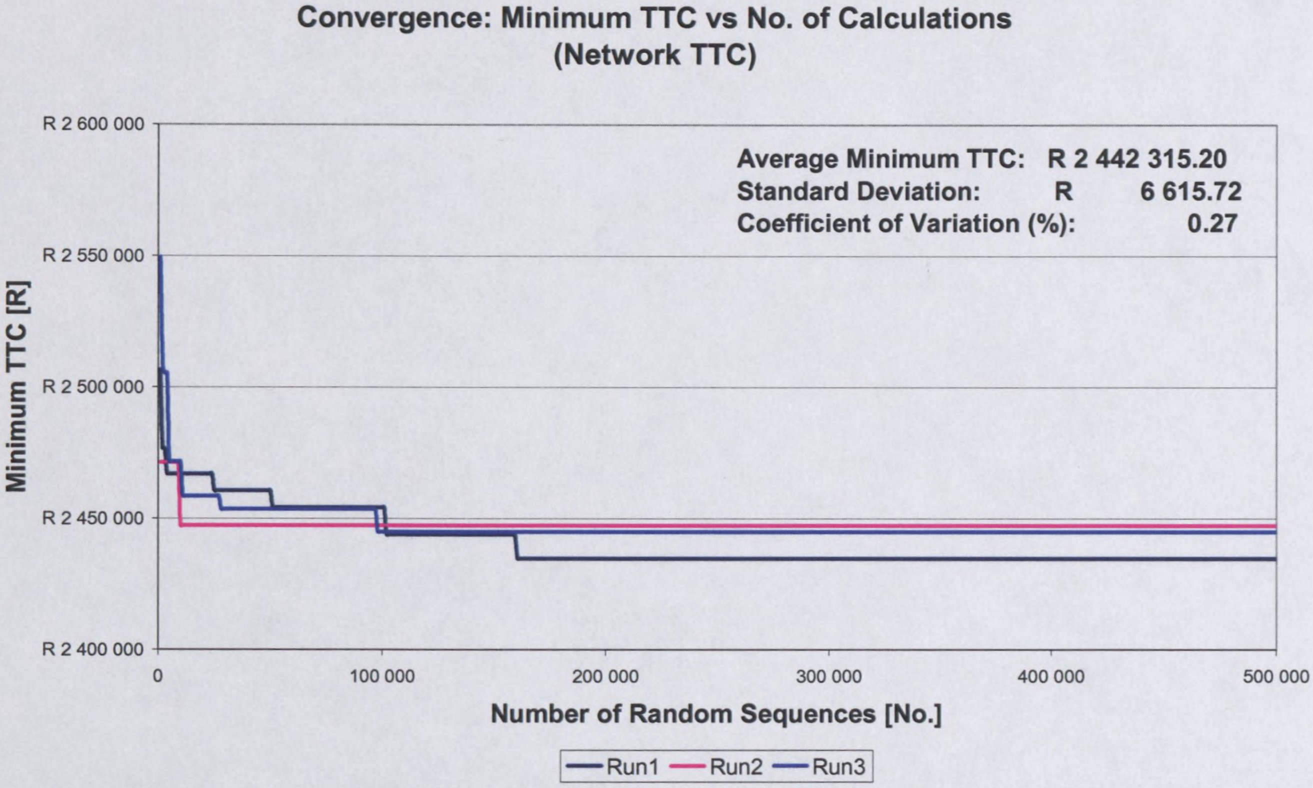


Figure 6-2: Results of individual calculation runs for 500 000 calculations.

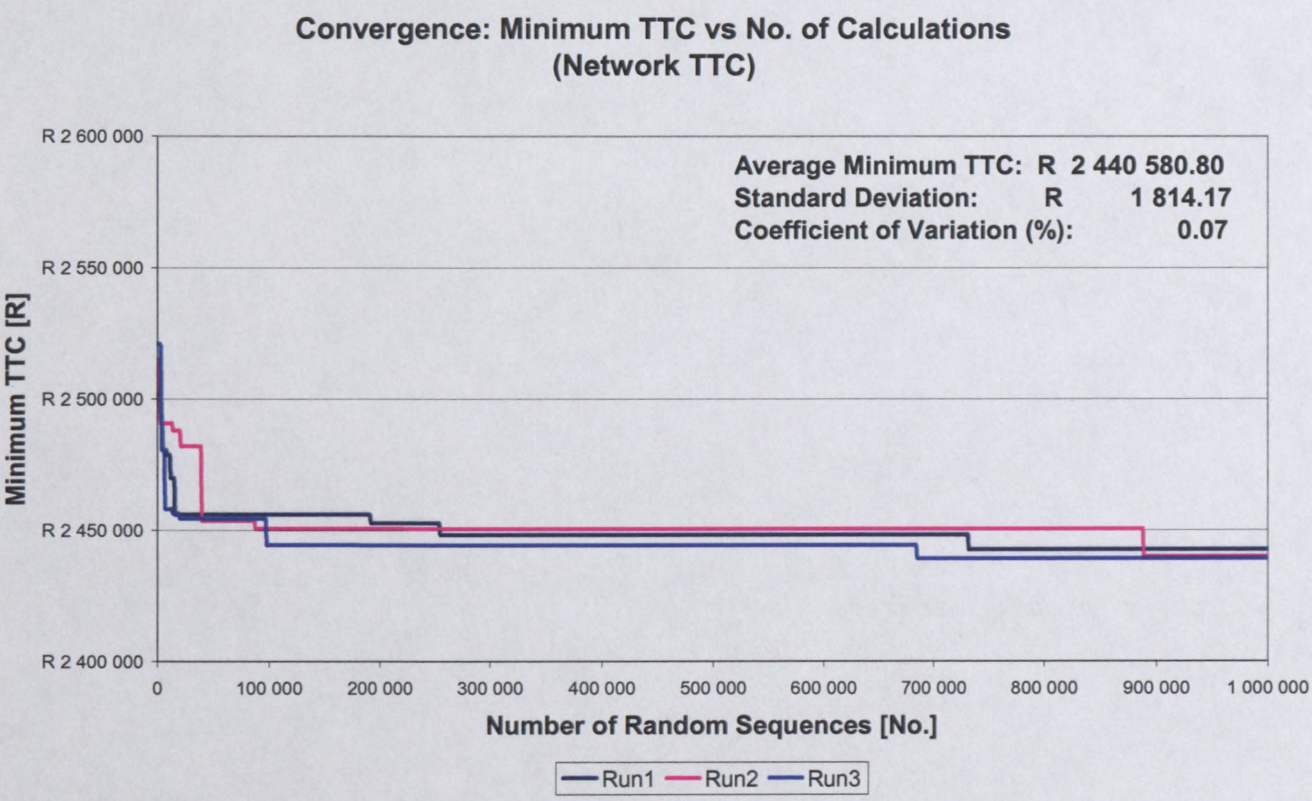


Figure 6-3: Results of individual calculation runs for 1 000 000 calculations.

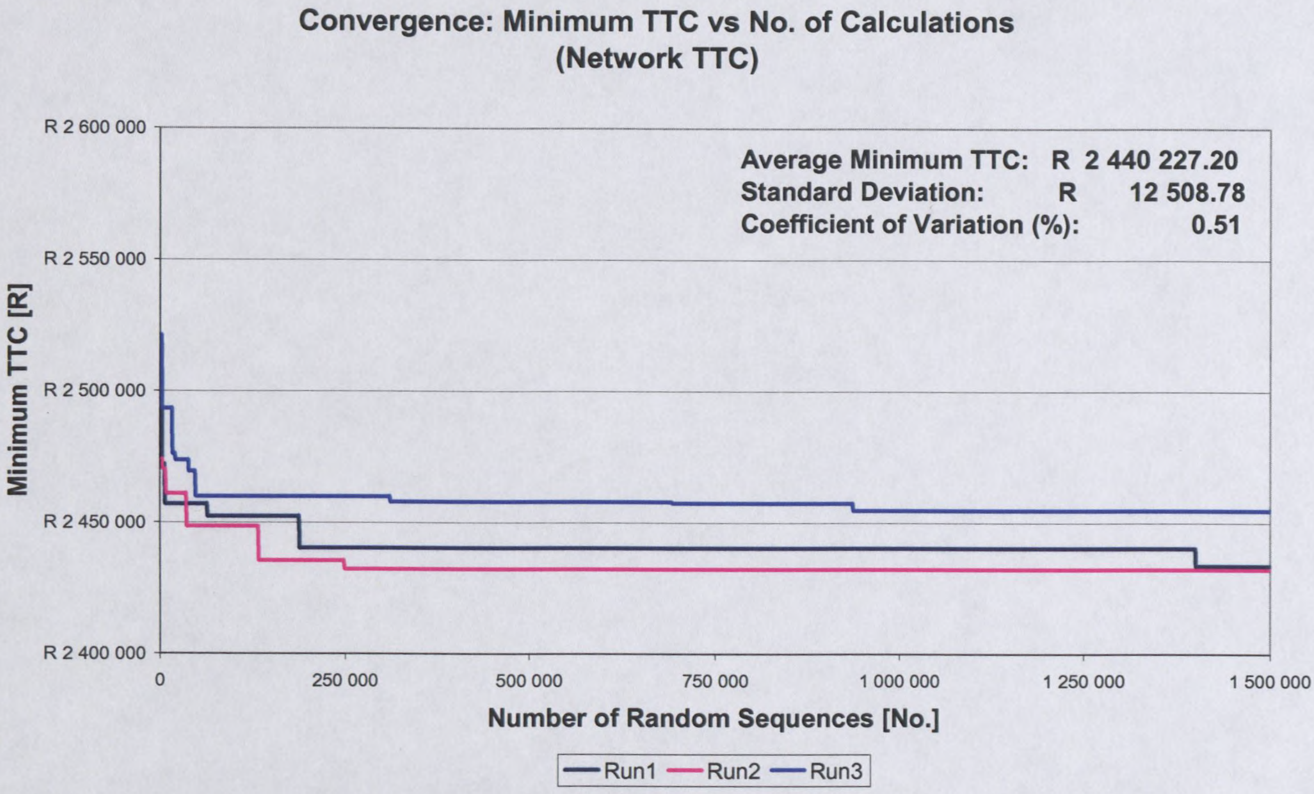


Figure 6-4: Results of individual calculation runs for 1 500 000 calculations.

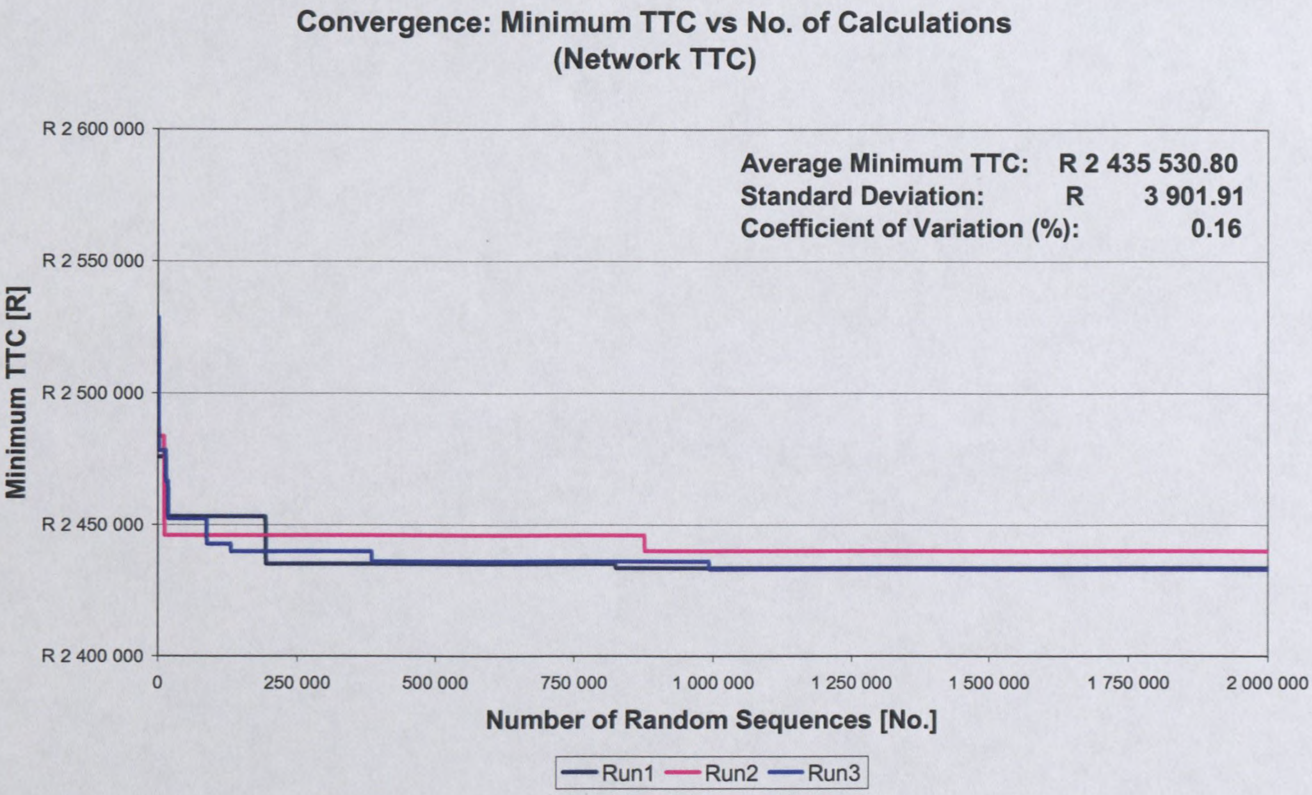


Figure 6-5: Results of individual calculation runs for 2 000 000 calculations.

6.3.2.2 Discussion of Network Cost Algorithm results

Based on the results reported in Table 6-3 it is clear that the network TTC seems to have a constant value. Furthermore, the coefficient of variation (COV) reported in the table show that the results for the network TTC are practically the same. Thus, for the maintenance ward considered and the values used as reported in Table 6-1, the network cost algorithm gives practically constant results for the network TTC.

The small difference in the values for the network TTC for the different calculation limit values, points toward the influence of the sections not bladed on the network TTC. Thus, even though higher and higher calculation limit values may be used it would not necessarily reduce the network TTC further. This is because the sections that are not bladed during the scheduling window influence the network TTC so drastically. On this point it may be worth noting that the contribution to the TTC of the sequences, i.e. only considering the sections contained in a sequence when calculating the TTC, is on average one fifth (20%) of the TTC for the ward. It has to be mentioned that an increase in the resources available to perform routine maintenance, as either reflected by decreasing the length of roads maintained by one blading team or increasing the production rate achieved, may result in bigger discrepancies between the average minimum TTC achieved at the different calculation limit values – due to the fact that the contribution of the sequence TTC to the network's TTC will be higher.

The Agency Cost and VOC that contribute to the TTC were calculated for the ward network. This was done for the results of the 1 500 000 sequences. The contributions of both the Agency Cost and VOC to the ward network TTC are listed in Table 6-4. It is clear that the biggest contributor to the network TTC is VOC. For the Swellendam-Infanta ward network with variables as used during the testing of the algorithms, the contribution of Agency Cost to TTC is approximately 13.6% and VOC contributes approximately 86.4% to TTC.

A calculation limit value of 1 500 000 is considered to be sufficient to arrive at a satisfactory result. From the reported results it is clear that using a calculation limit value of 1 500 000 would give a blading sequence resulting in a close to optimum network TTC. The fact that lower calculation limit values also give fairly good results is overshadowed by the fact that the number of possible sequences is 2.652×10^{32}. Consequently one should use as high a number as available computer equipment allows. Working in favour of calculation limit values which are much lower than the number of possible sequences is the fact that we are only interested in routes/sequences of which the time length conforms to the criteria:

$$0.8*\text{schedule window} \leq \text{sequence length} \leq 1.1*\text{schedule window}$$

Obviously, it is not known at this stage for this ward how many sequences with time length conforming to the criteria do exist – this may be not such a significant amount compared to the possible number of sequences. Furthermore, it is prudent to assume that the more sequences are generated the closer one would get to generating a sequence close to the absolute minimum sequence. Convergence aspects of the algorithm should be investigated further.

The determination of a blading schedule using this algorithm at a calculation limit value of 1 500 000 requires approximately three hours using the notebook computer available to test it.

Table 6-4: Contribution of Agency Cost and VOC to TTC for ward network.

Calculation run	Measure	Value	Contribution
Run 1	Minimum TTC	R 2 433 635.69	
	Agency Cost	R 331 087.50	13.60%
	VOC	R 2 102 548.19	86.40%
Run 2	Minimum TTC	R 2 432 392.59	
	Agency Cost	R 331 087.50	13.61%
	VOC	R 2 101 305.09	86.39%

6.4 Comparing the two algorithms' results

It is clear that the Network Cost Algorithm gives a better result in terms of lowering the network TTC. It is strictly not correct to compare the network TTC of the two algorithms, since the network roughness algorithm is not aimed at minimising the network TTC. However, since the aim of the network roughness algorithm is to minimise the network roughness and roughness is a key variable in the calculation of VOC, one may compare the resulting TTC for the two algorithms. Thus, it seems that the use of the network cost algorithm not only results in a lower network TTC, but by implication also in a better network roughness.

Comparison on the basis of the resulting blading sequences is not realistic, since for the network roughness algorithm the sequence consists of segments while for the network cost algorithm the sequence consists of sections.

The network cost algorithm has potential for further development to improve the quality of the optimisation process. Two aspects need to be addressed: 1. the deterioration model(s) that is used during scheduling need to be refined; 2. the optimisation function may be further expanded to include socio-economic factors in the calculation of the TTC.

6.5 Compliance with Requirements of the DM's

In summarising the different DM's system requirements for the BOM, two main themes emerged. Firstly, all of the DM's would appreciate a system that can aid the managers in producing works programmes (or schedules) for the blading teams. Secondly, another common theme that emerged was the need for a system linking GPS data to show up-to-the-minute blading team locations. Another important requirement linked to the scheduling of work is the ability to handle emergency situations, specifically rescheduling routine maintenance operations after emergencies have been addressed.

It was stated in the introductory chapter that this study was undertaken to (refer to section 1.2.):

> "...**develop, implement and test scheduling algorithms** that may be used **for the optimisation of routine maintenance** of the Gravel Road Network of the PGWC."

Thus, insofar as the algorithm development is concerned the first main requirement of the DM's were met, namely the algorithms may be used to schedule the routine maintenance work of the DM's on the gravel road network. It is also fair to state that the algorithms are efficient both in terms of time needed for scheduling and computer requirements of the computer on which they are employed. The second main requirement of the DM's was not met, namely providing a system linked to GPS data showing the location of the blading teams. This is because this study was not aimed at developing the interface between GPS transponders fitted to the graders and showing the GPS data received from such transponders on maps on screen. Certainly for a commercial system the GPS interface of the system will be of the utmost importance, but this study was only concerned with developing, implementing and testing scheduling algorithms that could be used in such a commercial system.

The algorithms are able to handle emergencies as follows:

1. When an emergency occurs, respond to it and send a team to address the problem. That team's work now has to be rescheduled.
2. Since the DM personnel have good communication with the respective blading teams, it is possible to enter the current condition of the gravel segments in the ward of the blading team responding to the emergency. This will be based on the work done up to the date of responding to the emergency for the current schedule.
3. It is now possible to determine a new schedule for that blading team.

In summary it may be stated that the algorithms developed addresses the need of the DM's for some tool that can assist personnel in efficiently scheduling routine maintenance work on the gravel road network. In developing the algorithms the knowledge gained from the interviews with the DM's were used to decide what variables had to be set by the user of the pilot application. This study was successful in using the information gained from the interviews in the development of the scheduling algorithms.

7 Conclusions and Recommendations

7.1 Conclusions

This study was successful in achieving its objective, namely:

> "...**develop, implement and test scheduling algorithms** that may be used **for the optimisation of routine maintenance** of the Gravel Road Network of the PGWC."

Two algorithms were developed that addresses to methods of optimisation of routine maintenance. The first method is aimed at minimising the network roughness on a ward basis. The second method aims to minimise the total transportation cost on a ward basis. These algorithms make use of sound mathematical and engineering theory, e.g. graph theory for route determination; TRH20 and HDM4 deterioration models; simplified HDM4 methodology for calculation of VOC. Furthermore, the constraints and variables identified in the interviews with the DM personnel were used in the development of these algorithms. As a result it may be concluded that the main criteria for success, namely sound theory and applicability to the Western Cape situation, were met.

The other criteria for success were also met in this study. Firstly, the algorithms were mapped efficiently to the computer using the Java programming language. This was done in the pilot application described in Chapter 5. Secondly, these algorithms are time efficient. A schedule is produced within a matter of milliseconds with the first algorithm, while the second algorithm requires approximately three hours to produce a schedule when the number of calculations are set at 1 500 000. Both these times were achieved on medium range notebook computer generally available in the market.

In Chapter 6 the two algorithms were evaluated and compared. It was shown that the network cost algorithm gives a lower TTC for the ward on which it was tested, compared to the network roughness algorithm. Since roughness is used to determine VOC, it may be reasonable to assume that the network cost algorithm also results in a lower network roughness for the ward under consideration. Thus, it was concluded that the network cost algorithm holds more promise than the network roughness algorithm for further development.

Even though it has not been verified by way of calculation, it is fair to assume that the use of either algorithm will be an improvement on current practice. This is because it is almost impossible for a person to schedule routine maintenance on a gravel road network optimally, by either minimising network roughness or minimising network cost, given the complexity of the problem and number of variables influencing the result. One could argue that it is possible to calculate the TTC for the current situation in the Western Cape in order to compare this to the results of the algorithms, but appropriate data sets are not available to do this.

7.2 Recommendations

The deterioration models used in the algorithms have to be refined. It was stated that use of the TRH20 and HDM4 deterioration models resulted in unrealistic blading frequencies. The source of uncertainty may be manifold and it is necessary to determine and address the model sensitivities for both deterioration models in order to employ them with confidence in the algorithms.

The optimisation function used in the network cost algorithm needs to be expanded to include socio-economic factors in the calculation of TTC. This could be done by use of utility analysis similar to the Eden DM's weighting method for the distribution of funds between segments. Convergence aspects of the Network Cost algorithm should be investigated further.

The algorithms developed in this study addresses scheduling of routine maintenance on a maintenance ward basis. Expansion of these algorithms to DM network level and later Provincial network level will provide the PGWC with a powerful tool that could be used in a Decision Support System for Gravel Road maintenance. Further research is necessary to develop the techniques to do this.

It is further recommended that the pilot application be used in the Swellendam/Infanta ward of the Overberg DM to test the use of the algorithms on a "live" ward network. Due to time constraints this was not possible during this study. The feedback from the blading teams and DM personnel will aid greatly in the further development and improvement of the algorithms and expansion to DM and Provincial level.

8 References

1. Burger, A.F., A. van der Gryp, G.D. van Zyl and H.G. Fourie. *"Simplification of HDM4 economic models for network level gravel road management systems"*. 8th International Conference on Low-Volume Roads. Reno, Nevada, USA. June, 2003.
2. du Plessis, F. *"Optimisation of the Maintenance of Trunk Road 84"*. M.Eng.(Civil) Thesis. Stellenbosch, 1991.
3. Highway Development and Management (HDM4) Documentation, Volume 4: Analytical Framework and model descriptions. World Road Association (PIARC). Paris, 2000.
4. Pahl, P.J. and R. Damrath. *"Mathematical Foundations of Computational Engineering"*. Springer, 2001.
5. Paige-Green, P. *"Re-evaluation of the southern African unpaved road deterioration models."* 6th International Conference on Low-Volume Roads. Minneapolis, Minnesota, USA. June, 1995.
6. Paige-Green, P and A.T. Visser. *"Comparison of the impact of various unpaved road performance models on management decisions."* 5th International Conference on Low-Volume Roads. Raleigh, North-Carolina, USA. May, 1991.
7. Personal Communication, January to March 2004. A series of interviews were conducted with the following personnel at the different District Municipalities:

Eden	Mr Hans Ottervanger
Overberg	Mr Mannie van Eeden
Boland	Mr Callie Calitz
West Coast	Mr Harmie Esterhuizen
Central Karoo	Mr Kobus Theron

8. Ruess, M. *"Graph Theory: An Introductory Approach"*. Lecture Notes, University of Stellenbosch. October 2000.
9. Technical Recommendations for Highways (TRH20): The Structural design, construction and maintenance of unpaved roads. Department of Transport. Pretoria, 1990.

Appendix A

Vehicle Operating Cost Calculation method based on HDM4 methodology: Extract from paper presented at the 8th International Conference on Low-Volume Roads. Reno, Nevada, USA. June 2003. [Burger, 2003]

SIMPLIFICATION OF HDM4 ECONOMIC MODELS FOR NETWORK LEVEL GRAVEL ROAD MANAGEMENT SYSTEMS

Abstract

This paper describes the procedure that is followed in the Gravel Road Management System (GRMS) of the Provincial Administration: Western Cape (PAWC) for the determination of scheduled maintenance priorities and upgrade to paved standard priorities for gravel roads.
An algorithm was developed that takes account of a number of factors to determine the costs and benefits of the two maintenance strategies of the PAWC. The cost and benefit streams are further used in the calculation of the Internal Rate of Return (IRR) for the different maintenance strategies. Prioritisation in the GRMS is based on the calculated IRR and the priority lists are further refined through a consultative process involving maintenance personnel, the community and head office decision makers.
As part of the implementation of the algorithm, the HDM4 equations for the calculation of Vehicle Operating Cost (VOC) were simplified for application in the Southern African context. This paper presents the results of a comparison between the output of the simplified HDM4 equations with the output of HDM4.
The conclusions that are reached show that the implementation of the procedure described in this paper had a number of positive consequences. This includes that the prioritisation of projects is based on sound principles that result in the effective and responsible use of available funds. It was also concluded that the simplified HDM4 VOC calculation results compare well with the HDM4 results.

SIMPLIFICATION OF HDM4 ECONOMIC MODELS FOR NETWORK LEVEL GRAVEL ROAD MANAGEMENT SYSTEMS

Section 2: Simplified HDM4 vehicle operating cost equations

The GRMS is a network-level management system that is used for the management of the gravel road network of the PAWC. The PAWC's road network is a well-established network on which problems with road geometry and drainage have been addressed effectively over time and the maintenance strategies described in this paper do not include changing the geometry of existing roads. Furthermore, if geometric changes are not considered relative to the Do Nothing option, it is senseless to include geometry in the calculation of VOC for further consideration in the economic analysis. This is because existing geometry is a contributing factor to VOC calculated for a specific road section. If maintenance actions are considered for that section of road but the actions do not include changes to the geometry, there will be no benefit arising from a change in geometry to be considered in the economic analyses of the maintenance options.

At the outset of the simplification of the HDM4 calculations it was decided that only roughness should influence decisions whether to regravel or upgrade and that terrain and curvature would not be considered in the calculations for the reasons mentioned above. In all cases HDM4 prescribes that the cost factors be calculated for uphill and downhill sections for a segment. The cost factor for the segment is then calculated by averaging the costs for up- and downhill. In the simplified procedure factors are calculated regardless of whether it is uphill or downhill. In certain cases road curvature is used as input for calculations and this is also ignored.

As it is necessary to base procedures on sound principles and to use recognised methodologies in the procedures of the GRMS, it was decided to use the HDM4 equations (*1*) for the calculation of VOC. HDM4 calculates VOC based on nine factors, which are:

1. Fuel Consumption*
2. Oil Consumption*
3. Tyre Consumption*
4. Vehicle Service Life* and Vehicle Utilisation *[These factors are not costs but are used to calculate costs]*
5. Parts Consumption*
6. Labour Hours*
7. Capital Costs: Depreciation* and Interest
8. Crew Hours
9. Overhead

Factors marked with an asterisk (*) are influenced directly by road roughness or are based on a factor that is influenced directly by roughness (e.g. labour hours is a function of parts consumption which is directly related to roughness).

The HDM4 methodology as used here has been simplified further by considering only four vehicle types (in stead of the sixteen used by HDM4) (*1, 2*). The vehicle types that are used are the types that are normally counted during a traffic survey by the PAWC:

- Light
- Heavy
- Taxi
- Bus

[The vehicle type "Taxi" is a common occurrence on South African roads. Typically a taxi would be a mini bus that is used for the transport of people and goods. The mini bus taxi forms the backbone of the South African public transport system.]

HDM4 uses vehicle type dependent parameters in the calculation of the VOC factors and these have been simplified (to ignore curvature and terrain) or condensed where applicable to reflect the vehicle types used in the modified methodology. The reader is referred to the HDM4 documentation to understand the complexity of the situation. The parameters related to vehicle types were based on the values of the following HDM4 vehicle types:

- Light – HDM4 type 4: Large Car
- Taxi – HDM4 type 12: Mini-Bus
- Heavy – HDM4 type 10: Heavy Truck
- Bus – HDM4 type 14: Medium Bus

Space does not permit an exhaustive explanation of the methodology that was followed to derive the equations presented in this paper. Again, the reader is referred to the HDM4 documentation (*1*) to appreciate the complexity of the procedure that was followed to develop the equations.

General VOC calculations

The VOC for a road segment with a specified AADT is calculated as (*1, 2*):

$$VOC = \left(TC_{av} + PARTSCOST + LABOURCOST + DEPCST_{av}\right) \times \frac{Length\ \ of\ road\ segment}{1000}$$
$$+ (FuelCost_{av} + OilCost_{av}) \times Length\ \ of\ road\ segment$$

where

$$FuelCost_{av} = \sum_{k=1}^{4} FC_k \times AADT_k \times TypeCost_k$$

TypeCost refers to the cost per litre of the two types of fuel, i.e. petrol or diesel.

$$OilCost_{av} = \sum_{k=1}^{4} OIL_k \times AADT_k \times Oilprice_k$$

Oilprice refers to the cost per litre of oil for the different vehicle types.

$$TC_{av} = \sum_{k=1}^{4} TC_{avk} \times TYRECOST_k$$

TYRECOST refers to the cost of a new tyre.

$$PARTSCOST = \sum_{k=1}^{4} PC_{avk} \times VEHCOST_k$$

VEHCOST refers to the cost of a new vehicle.

$$LABOURCOST = WAGECOST \times LH_{av}$$

WAGECOST refers to the average hourly wage for a technician working on a vehicle.

$$DEPCST_{av} = \sum_{k=1}^{4} AADT_k \times DEPCST_k$$

$$DEPCST = DEP \times NVPLT$$

NVPLT refers to the cost of a new vehicle less the cost of a set of new tyres.

The factor k in the summation refers to the vehicle type and the corresponding number and vehicle type is defined by the user (e.g. k = 1 refers to light, k = 2 refers to taxi, etc.), while the factors in the summations (e.g. TC_{avk}, LH_{av}, etc.) refer to the equations that are dependent on the roughness of the road. These equations are listed in Table to **Table** . Roughness is denoted by $\mathbf{RI_{av}}$ (where RI stands for Roughness Index) in the equations and is the average roughness (unit: **IRI**) for a road segment.

In the case of the Fuel and Oil Cost equations a limiting roughness is mentioned. This concept refers to the roughness in IRI above which a vehicle will not travel anymore at the constant speed assumed. When the equations were derived a vehicle operating speed of 80 km/h was assumed. This is the speed at which roughness measurements are normally conducted. The limiting roughness of each vehicle type was determined according to HDM4 (*1, 2*) and is listed below:

- Light, Taxi and Bus: IRI = 8.0
- Heavy: IRI = 7.0

The formulae for fuel consumption were derived as steady-state equations below the limiting roughness for the different vehicles. However, for the roughness values above the limiting roughness the fuel consumption was determined at representative roughness values and a regression analysis was performed to determine the fuel consumption equations for the vehicles above the limiting roughness.

The roughness values that were used for the regression analysis were:

- Light, Taxi and Bus: IRI = 8.0, 10.0 and 12.0
- Heavy: IRI = 7.0, 8.0, 10.0 and 12.0

The results of the regression analyses above the limiting roughness are listed as the fuel and oil consumption equations in **Table** and **Table** above the different limiting roughnesses.

It will also be noted that oil consumption is a function of fuel consumption. Thus there is a set of equations for the oil consumption above the limiting roughness and below the limiting roughness. For a full discussion of the derivation of the VOC equations the reader is referred to (*2*).

Comparing the simplified HDM4 equations with HDM4

The simplified equations used for the calculation of VOC compares well with the HDM4 equations. A comparison was made between the simplified equations and the HDM4 outputs (*2*, *7*) and an acceptable correlation was noted as can be seen in Figures 4, 5 and 6. Therefore, the simplified procedure was accepted for use in the GRMS. The comparison was made between the simplified equations and a set of equations that was derived from regression analyses on HDM4 output. The regression analyses (*7*) was done with the same traffic spectrum (light, taxi, heavy and bus) but a different HDM4 heavy vehicle was used (a type 11 was used). For this reason the equations for a HDM4 type 11 vehicle was derived to compare the results. The results of the comparison between the simplified type 11 and HDM4 type 11 vehicles are shown in Figure 6.

Figures 4, 5 and 6 show results for the HDM4 regressions including and excluding time-related costs. The simplified procedure did not include time-related costs in the derivation and that is why the HDM4 regression results are shown for both cases. It can be seen that the simplified equations give results that fall between the HDM4 regressions including and excluding time related costs. It is also evident from the figures that the simplified equations underestimate VOC at lower roughness values, but the results at higher roughness values compare well. Also apparent on the figures is the fact that the equations used previously by the PAWC totally underestimated VOC. Therefore, the simplified HDM4 equations were accepted and are now used in the PAWC GRMS for the calculation of VOC. The simplified HDM4 equations will also be implemented in the Pavement Management System (PMS) of the PAWC in the near future. *[The PMS is the management system for the paved roads under PAWC jurisdiction.]*

The VOC calculated according to the simplified HDM4 methodology is used as a direct input for use in the calculation of the IRR and the calculation thereof is discussed conceptually in the next section.

References

1. Odoki, J.B., and H.G.R. Kerali. *Analytical framework and model descriptions. Volume Four, HDM4 system documentation*. PIARC, France, 2000.
2. Burger A.F., and G.D. van Zyl. *Simplifying the HDM4 methodology for the calculation of vehicle operating cost.* Report VV1/346 v.5. V&V Consulting Engineers, South Africa, 2001.
3. Burger A.F., G.D. van Zyl and H.G. Fourie. *Implementing the HDM4 methodology for the calculation of VOC and IRR in the PAWC GRMS.* Report VV2/346 v.2. V&V Consulting Engineers, South Africa, 2001.
4. *Structural design, construction and maintenance of unpaved roads.* Draft TRH20. Department of Transport, South Africa, 1990. [TRH = **T**echnical **R**ecommendations for **H**ighways]
5. *Structural design of flexible pavements for interurban and rural roads.* Draft TRH4. Department of Transport, South Africa, 1996.
6. *Pavement management systems.* Draft TRH22. Department of Transport, South Africa, 1994.
7. *Vehicle operating cost study 2001 (using HDM4).* Report RNMS/T/14 Edition 2.0. Gautrans, Department of Transport and Public Works, Gauteng Provincial Government, South Africa, 2001.

List of Tables and Figures

Table 1 Fuel consumption factor on unpaved roads below and above limiting roughness. (*2*)

IRI	Vehicle	FC – Unpaved
≤ 8	Light	$74.08\times10^{-3}+1.895\times10^{-3}\times RI_{av}+2.739\times10^{-6}\times RI_{av}^2$
≤ 8	Taxi	$66.07\times10^{-3}+2.498\times10^{-3}\times RI_{av}+6.747\times10^{-6}\times RI_{av}^2$
≤ 7	Heavy	$221\times10^{-3}+7.936\times10^{-3}\times RI_{av}+8.101\times10^{-6}\times RI_{av}^2$
≤ 8	Bus	$97\times10^{-3}+4.112\times10^{-3}\times RI_{av}+5.864\times10^{-6}\times RI_{av}^2$
> 8	Light	$0.0613\times e^{0.047\times RI_{av}}$
> 8	Taxi	$0.0581\times e^{0.0495\times RI_{av}}$
> 7	Heavy	$0.2009\times e^{0.0455\times RI_{av}}$
> 8	Bus	$0.0983\times e^{0.0352\times RI_{av}}$

Table 2 Oil consumption on unpaved roads below and above limiting roughness. (2)

IRI	Vehicle	OIL – Unpaved
≤ 8	Light	$400\times10^{-6}+2.8\times10^{-6}\times(74.08\times10^{-3}+1.895\times10^{-3}\times RI_{av}+2.739\times10^{-6}\times RI_{av}^2)$
≤ 8	Taxi	$667\times10^{-6}+2.8\times10^{-6}\times(66.07\times10^{-3}+2.498\times10^{-3}\times RI_{av}+6.747\times10^{-6}\times RI_{av}^2)$
≤ 7	Heavy	$3.1\times10^{-3}+2.1\times10^{-6}\times(221\times10^{-3}+7.936\times10^{-3}\times RI_{av}+8.101\times10^{-6}\times RI_{av}^2)$
≤ 8	Bus	$1.75\times10^{-3}+2.1\times10^{-6}\times(97\times10^{-3}+4.112\times10^{-3}\times RI_{av}+5.864\times10^{-6}\times RI_{av}^2)$
> 8	Light	$400\times10^{-6}+2.8\times10^{-6}\times(0.0613\times e^{0.047\times RI_{av}})$
> 8	Taxi	$667\times10^{-6}+2.8\times10^{-6}\times(0.0581\times e^{0.0495\times RI_{av}})$
> 7	Heavy	$3.1\times10^{-3}+2.1\times10^{-6}\times(0.2009\times e^{0.0455\times RI_{av}})$
> 8	Bus	$1.75\times10^{-3}+2.1\times10^{-6}\times(0.0983\times e^{0.0352\times RI_{av}})$

Table 3 Tyre cost factor for unpaved roads. (2)

Vehicle	TC (unpaved roads)
Light	$42.98\text{x}10^{-3} + 569.4\text{x}10^{-6}\text{x}(309.4 + 23.2\text{xRI}_{av})^2 + 5.4\text{x}10^{-3}$
Taxi	$34.5\text{x}10^{-3} + 274\text{x}10^{-6}\text{x}(395.8 + 29.69\text{xRI}_{av})^2 + 5.4\text{x}10^{-3}$
Heavy	$50.73\text{x}10^{-3} + 3.099\text{x}10^{-9}\text{x}(1547.99 + 116.1\text{xRI}_{av})^2 + 27\text{x}10^{-3}$
Bus	$35.57\text{x}10^{-3} + 6.804\text{x}10^{-9}\text{x}(780.46 + 58.53\text{xRI}_{av})^2 + 16.2\text{x}10^{-3}$

Table 4 Parts consumption factor. (2)

Vehicle	PC
Light	$1.355x10^{-3} + 227.4x10^{-6}xRI_{av}$
Taxi	$1.348x10^{-3} + 227.4x10^{-6}xRI_{av}$
Heavy	$1.614x10^{-3} + 412.6x10^{-6}xRI_{av}$
Bus	$228.5x10^{-6} + 196.4x10^{-6}xRI_{av}$

Table 5 Labour hour factor. (2)

Vehicle	LH
Light	$77.14x(1.355x10^{-3} + 227.4x10^{-6}xRI_{av})^{0.547}$
Taxi	$77.14x(1.348x10^{-3} + 227.4x10^{-6}xRI_{av})^{0.547}$
Heavy	$301.46x(1.614x10^{-3} + 412.6x10^{-6}xRI_{av})^{0.519}$
Bus	$293.44x(228.5x10^{-6} + 196.4x10^{-6}xRI_{av})^{0.519}$

Table 6 Depreciation cost factor. (2)

Vehicle	DEP
Light	$\dfrac{980 + RI_{av}}{\left[\dfrac{240000}{1+e^{\left(-65.8553\times RI_{av}^{-1.9194}\right)}}\right]}$
Taxi	$\dfrac{980 + RI_{av}}{\left[\dfrac{240000}{1+e^{\left(-65.8553\times RI_{av}^{-1.9194}\right)}}\right]}$
Heavy	$\dfrac{980 + RI_{av}}{\left[\dfrac{1204000}{1+e^{\left(-65.8553\times RI_{av}^{-1.9194}\right)}}\right]}$
Bus	$\dfrac{980 + RI_{av}}{\left[\dfrac{490000}{1+e^{\left(-65.8553\times RI_{av}^{-1.9194}\right)}}\right]}$

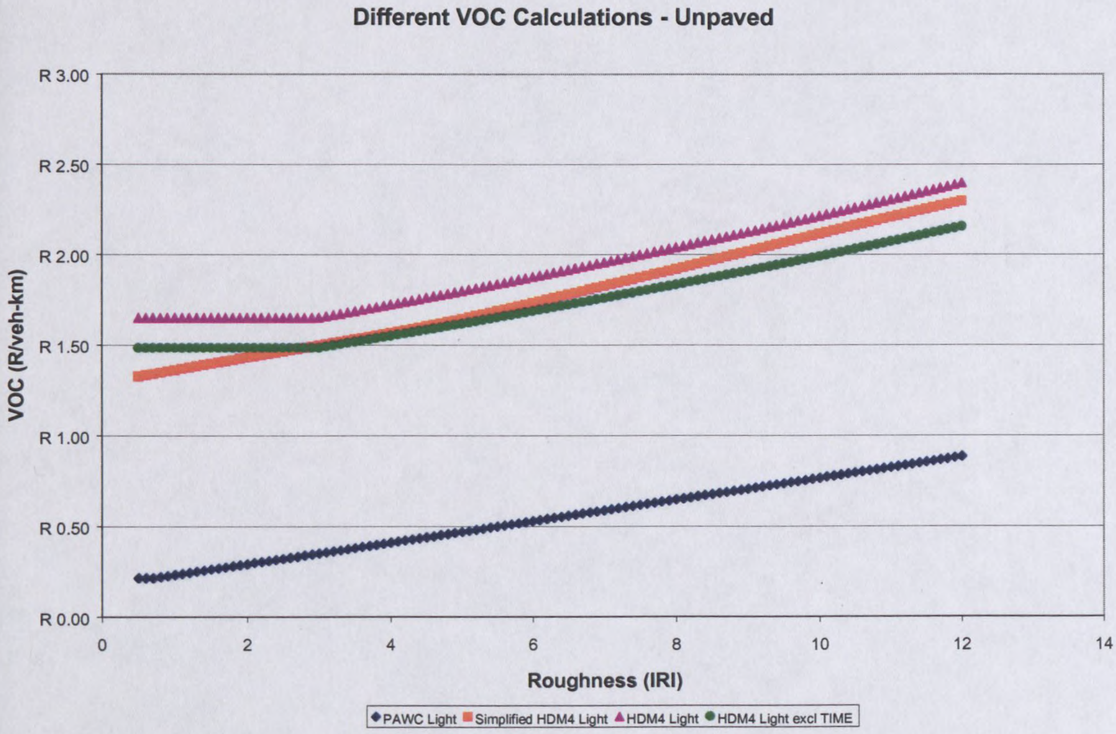


Figure 4 Comparison of simplified HDM4 and HDM4 VOC calculations for light vehicles.

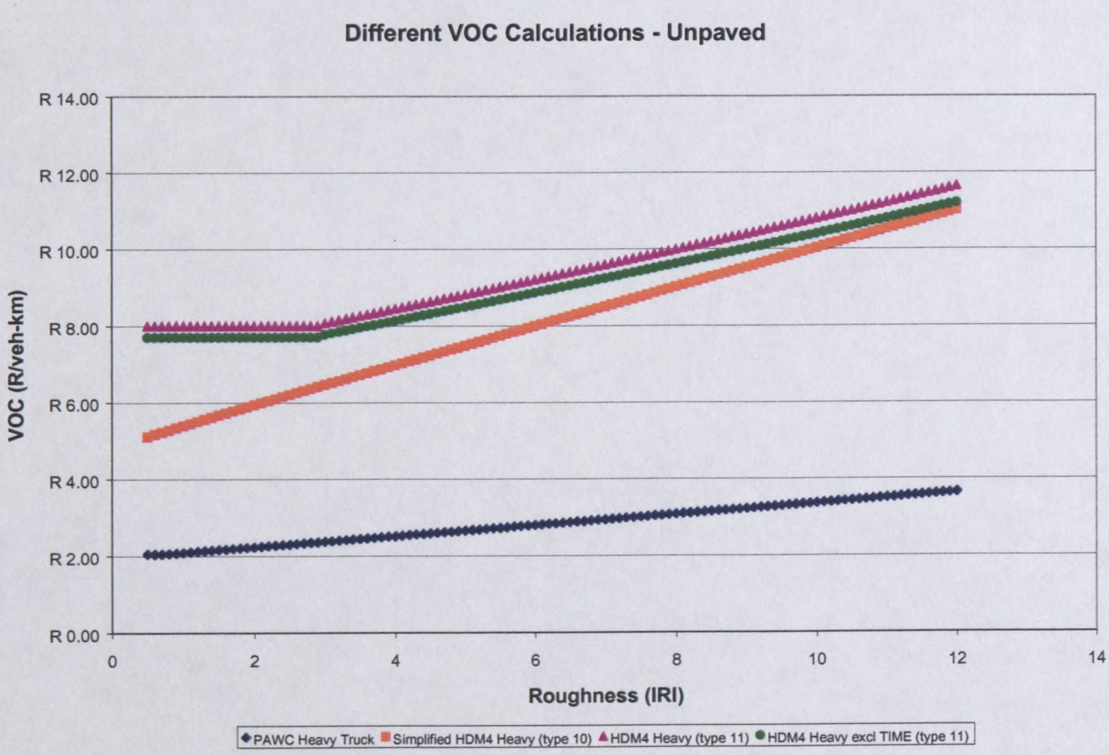


Figure 5 Comparison of simplified HDM4 and HDM4 VOC calculations for heavy vehicles.

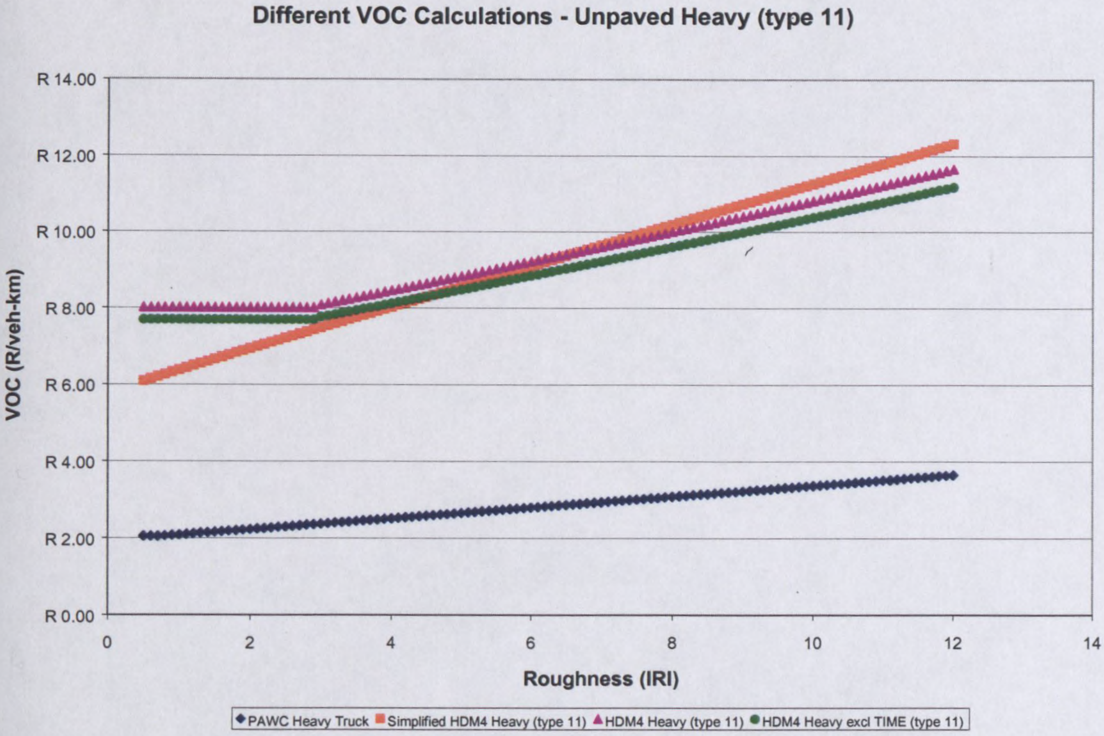


Figure 6 Comparison of simplified HDM4 and HDM4 VOC calculations for vehicle type 11.