A FIELD STUDY ON HAUL TIME VARIABILITY IN OPEN PIT MINES

by

Andrew Chapman

A thesis submitted to the Department of Mining Engineering
In conformity with the requirements for
the degree of Master of Applied Science (Engineering)

Queen’s University
Kingston, Ontario, Canada
(August, 2012)

Copyright © Andrew Chapman, 2012
Abstract

A Field Study on Haul Time Variability in Open pit Mines

Andrew Chapman
Master of Applied Science (Engineering)
The Robert M. Buchan Department of Mining Engineering
August, 2012

As open pit mines get larger and their road networks more complex, haul road traffic is an area of increasing concern for mine operators. In particular, engineers from Barrick Gold Inc. have observed haul truck interactions in some of their larger open pit mines that are clearly disruptive to the haulage cycle. Such traffic effects cause uncertainty in haul times over identical routes and commercial dispatch systems do not explicitly account for this variability. This research, based on first-hand data collected from the Goldstrike Open Pit in Northern Nevada, focuses on assessing the level of variability in haul times, identifying the sources of said variability, and determining the effects on haulage optimization.

Initial data analysis shows different levels of variability over the range of observed haul times, and further investigation identifies certain haulage parameters as significant sources of said variability, such as road topography and intersection effects. Focusing on these areas of interest, relevant data sets were manipulated to reduce levels of data dispersion and compared to observed results in order to quantify the effects in terms of changes to overall mean haul times along identical routes. It was observed that variability has a negative impact on travel times and moderate variability reduction was observed to increase productivity by 1–2 % when used as an input to a simple dispatch simulation program based on the Goldstrike Open Pit.

KEYWORDS: cycle time, dispatch, variability, optimization, linear programming.
Acknowledgements

First I would like to thank my advisor Dr. Joshua Marshall for his unwavering support and guidance. He helped me to stay the course of my research, and motivated me to continuously challenge myself. Our meetings were always a source of fresh perspective, which time after time proved to be crucial for the advancement of my research. I am humbled to have worked with such a brilliant researcher. Thank you to Dr. Laeeque Daneshmend (Department Chair) for supporting me through the initial application process. With his help, the process was relatively quick and painless. Many thanks to Chris Cameron, for his coding efforts and input during the final stages of my data analysis. I must also acknowledge the administrative help of Wanda, Tina and Kate from the Department office for what can only be described as ‘headache-avoidance’ over the past two years.

Several individuals from Barrick Gold Inc. went above and beyond to facilitate my research. Thanks to Andrew Scott (Director of Technology) for providing the research opportunity, through his vision and subsequent funding of our research group. I would also like to thank Salar Javid (Project Engineer) who was always available for guidance, whether it be a quick chat or a long discussion. Thanks to Russ Downer, for allowing me to access the site and work alongside his exceptional team at Goldstrike: namely Casey Erickson, Kiley Zacharias and Ben McKnight, who went out of their way on a daily basis to accommodate me during the field trip. I owe these three a special debt of gratitude for their patience and kindness.

Thank you to Dr. Jonathan Peck for inspiring me to pursue a career in Mining Engineering and providing invaluable guidance throughout my career. He is a remarkable role model and I attribute much of my success to him.

Lastly I wish to thank my family and friends for their love and encouragement. You truly are my motivation to succeed.
# Table of Contents

Abstract........................................................................................................................................ii
Acknowledgements....................................................................................................................iii
Chapter 1 Introduction ................................................................................................................1
  1.1 Motivation............................................................................................................................1
  1.2 Context: Goldstrike Open Pit.............................................................................................2
    1.2.1 Fleet ............................................................................................................................4
    1.2.2 Road Network .............................................................................................................5
    1.2.3 Productivity ...............................................................................................................6
    1.2.4 Problem Statement: Haul Times .................................................................................7
  1.3 Scope of Work .......................................................................................................................7
  1.4 Thesis Format .......................................................................................................................8
Chapter 2 Dispatch Systems .......................................................................................................9
  2.1 Fleet Management................................................................................................................9
    2.1.1 Components of a Dispatch System ...........................................................................11
    2.1.2 MMS DISPATCH® .....................................................................................................12
  2.2 Variability in Current Dispatching Systems .....................................................................16
Chapter 3 Measuring Variability in Haul Times .......................................................................18
  3.1 Experiment Design..............................................................................................................20
    3.1.1 Data Requirements ....................................................................................................20
    3.1.2 Data Collection .........................................................................................................22
      3.1.2.1 Qstarz Units .......................................................................................................22
      3.1.2.2 MMS GPS data .................................................................................................23
    3.1.3 Methodology .............................................................................................................24
  3.2 Qstarz Data .......................................................................................................................25
  3.3 Data Analysis .....................................................................................................................27
    3.3.1 MATLAB® .................................................................................................................29
    3.3.2 IMPORT .....................................................................................................................30
    3.3.3 POLYGONS ..............................................................................................................33
    3.3.4 PARSE ........................................................................................................................36
    3.3.5 LEG ID ......................................................................................................................37
    3.3.6 ANALYZE ..................................................................................................................38
    3.3.7 Support Programs.....................................................................................................38
List of Figures

Figure 1-1- Map showing location of Carlin-type deposits (black dots), including the specific location of the Goldstrike Mine and its relative position to the towns of Carlin and Elko [1]........... 3
Figure 1-2- the Betze-Post Open Pit at Goldstrike [4]................................................................. 4
Figure 2-1- DISPATCH® algorithms (adapted from [25]) ................................................................. 14
Figure 2-2- Example of moving back average calculation for haul time estimation......................... 17
Figure 3-1- Qstarz GPS receiver/datalogger used for data collection, model BT-Q1000XT........... 23
Figure 3-2- MATLAB® code flowchart............................................................................................ 30
Figure 3-3- Plot of all Qstarz truck position data (latitude vs. longitude) ........................................ 32
Figure 3-4- Dispatch screenshot of open pit network, July 25, 2011 .............................................. 33
Figure 3-5- Polygons (from ac_polygons) superimposed on all Qstarz data................................... 35
Figure 3-6- An example of parsed data, showing only points lying outside the polygons, i.e., those travelling between nodes................................................................. 37
Figure 3-7- Dispatch screenshot of pit road network taken on July 15, 2011, with the road section between Node 3 (Q3&DUMPS) and Node 10 (SWSHOVLS) highlighted ...................... 40
Figure 3-8- Topographic map of Goldstrike pit, July 13, 2011, with the road section between Node 3 (Q3&DUMPS) and Node 10 (SWSHOVLS) highlighted.............................. 41
Figure 3-9- Histogram plot for haul leg 03to10.......................................................... 42
Figure 3-10- Histogram plot for haul leg 10to03.............................................................................. 43
Figure 3-11- Dispatch screenshot of pit road network taken on July 15, 2011, with the road section between Node 8 (4630WD01) and Node 10 (SWSHOVLS) highlighted .............. 44
Figure 3-12- Topographic map of Goldstrike pit, July 13, 2011, with the road section between Node 8 (4630WD01) and Node 10 (SWSHOVLS) highlighted......................... 45
Figure 3-13- Histogram plot for haul leg 08to10............................................................................. 46
Figure 4-1- PDF generated for a new method of truck localization [65].............................................. 51
Figure 4-2- Goldstrike pit with intersection Q3INT of interest highlighted.................................... 53
Figure 4-3- Schematic of the Q3INT intersection at Goldstrike, showing stop signs and traffic flows......................................................................................................................... 54
Figure 4-4- Goldstrike pit with intersections, polygons and connecting Routes A & B highlighted ................................................................................................................................. 56
Figure 4-5- Histograms for Route A (10-SWSHOVLS to 03-Q3&DMPS along the South haul road). All data (top), filtered at 2 km/h (middle), filtered at 4 km/h (bottom).................. 58
Figure 4-6- Histograms for Route B (10-SWSHOVLS to 03-Q3&DMPS along the North haul road). All data (top), filtered at 2 km/h (middle), filtered at 4 km/h (bottom). ............................... 59
Figure 4-7- Mean haul times of data vs. speed threshold applied ........................................ 61
Figure 4-8- Standard deviation of data vs. speed threshold applied ........................................ 61
Figure 4-9- Instantaneous speed vs. cumulative distance travelled along Route A, from shovel (polygon #10) to dump (polygon #3). Raw data (top) and median filtered using a 5 data point window .................................................................................................................. 64
Figure 4-10- Instantaneous speed vs. cumulative distance travelled along Route A, from dump (polygon #3) to shovel (polygon #10). Raw data (top) and median filtered using a 5 data point window .................................................................................................................. 65
Figure 4-11- Speed distributions for loaded one-way haul traffic between shovel (polygon #10) and dump (polygon #3) along Route A using a bin width of 45 metres. Haul road elevation profile superimposed in red ................................................................................................. 67
Figure 4-12- Speed distributions for empty one-way haul traffic between dump (polygon #3) and shovel (polygon #10) along Route A using a bin width of 45 metres. Haul road elevation profile superimposed in red ................................................................................................. 68
Figure 4-13- Goldstrike pit with polygons and connecting Route C highlighted ............................... 71
Figure 4-14- Speed distributions for empty one-way haul traffic between dump (polygon #10) and shovel (polygon #5) along Route C using a bin width of 15 metres. Haul road elevation profile superimposed in red ................................................................................................. 73
Figure 4-15- Speed distributions for loaded one-way haul traffic between dump (polygon #5) and shovel (polygon #10) along Route C using a bin width of 15 metres. Haul road elevation profile superimposed in red ................................................................................................. 74
Figure 4-16- Histogram for Route C (10-SWSHOVLS to 05-5010WD01 along the South haul road). All data ........................................................................................................................................ 75
Figure 4-17- Histogram for Route C (05-5010WD01 to 10-SWSHOVLS along the South haul road). All data ........................................................................................................................................ 76
Figure 4-18- Histogram of node residence times at the shovels on the main mining bench (polygon #10- SWSHOVLS) ...................................................................................................................................... 80
Figure 4-19- Histogram of node residence times at the 4630 in-pit dump (polygon #8- 4630WD01) ...................................................................................................................................... 80
Figure 5-1- Data Analysis Flowchart Part 1- for variability identification & visualization .......... 86
Figure 5-2- Data Analysis Flowchart Part 2- data modelling and simulation ............................... 88
Figure 5-3- Group 1 routes and polygons- 03to10 (green) and 08to10 (blue) ............................... 90
Figure 5-4- Instantaneous speed and cumulative haul time vs. cumulative distance travelled, route 03to10 .................................................................................................................................................. 91
Figure 5-5- Box plot distributions of incremental speeds (top) and times (bottom) over haul leg 03to10. Relative elevation profile is superimposed in red.................................................................................................................. 92
Figure 5-6- Simulated bin times vs. Distance along haul route- leg 03to10, Group 1- before (top) and after (bottom) application of variability exclusion rule (removal of data in Q1 and Q4) ........................................ 94
Figure 5-7- Simulated bin times vs. Distance along haul route- leg 03to10, Group 1- before (top) and after (bottom) correction for speeding trucks (>40 km/h)........................................................................................................... 95
Figure 5-8- Simulated bin times vs. Distance along haul route- leg 03to10, Group 1- before (top) and after (bottom) correction for slow/stopping trucks (<5 km/h) ............................................................................................ 96
Figure 5-9- Box plot distributions of incremental speeds (top) and times (bottom) over haul leg 10to03. Relative elevation profile is superimposed in red................................................................................................. 99
Figure 5-10- Simulated bin times vs. Distance along haul route- leg 10to03, Group 2- before (top) and after (bottom) application of variability exclusion rule (removal of data in Q1 and Q4) .... 100
Figure 5-11- Simulated bin times vs. Distance along haul route- leg 10to03, Group 2- before (top) and after (bottom) correction for speeding trucks (>40 km/h) ......................................................................................... 101
Figure 5-12- Simulated bin times vs. Distance along haul route- leg 10to08, Group 2- before (top) and after (bottom) correction for slow/stopping trucks (<5 km/h) ............................................................... 102
Figure 5-13- Speed vs. Cumulative Distance Travelled along leg 10to03ALT, with intersection circled in black......................................................................................................................................................... 105
Figure 5-14- Speed vs. Cumulative Distance Travelled along leg 10to03ALT, zoomed in on the intersection area (~2450-2600 m) ................................................................................................................... 105
Figure 5-15- Intersection behaviour analysis for Group 3 routes- 10to03ALTand 10to04ALT (136 total one-way hauls)........................................................................................................................................ 106
Figure 5-16- Before (left) and after (right) box-and-whisker plots of Q3INT intersection crossing times. Span: 2450-2600 m. ........................................................................................................................................ 108
Figure 5-17- Simulated bin times vs. Distance along haul route- leg 10to04ALT, Group 3- before (top) and after (bottom) application of variability exclusion rule. Removal of data in Q1 & Q4 for section 3- intersection (blue) only ........................................................................................................... 109
Figure 5-18- Box plot distributions of incremental speeds (top) and times (bottom) over haul leg 10to04ALT. Relative elevation profile is superimposed in red ......................................................................................... 111
Figure 5-19- Box plot distributions of incremental times over haul leg 10to04ALT with bin size of 30 m ......................................................................................................................................................... 112
Figure 5-20- Flowchart of the discrete event simulator basic structure [69] ......................................................................................................................... 115
Figure 5-21- Network diagram of simplified Goldstrike network used for simulation experiments ................................................................. 118
Figure 5-22- Experiment 1 results .................................................................................................................. 122
Figure 5-23- Experiment 2 results .................................................................................................................. 124
Figure 5-24- Experiment 3 results .................................................................................................................. 126
List of Tables

Table 1-1- Goldstrike Truck and Shovel Fleet Details [5], [6] ................................................................. 5
Table 3-1- Comparison of Advanced Travel Time Collection Techniques (reproduced from [33]) ................................................................. 19
Table 3-2- Comparison of available data collection techniques using GPS .............................................. 22
Table 3-3- Data sample from a Qstarz GPS receiver/datalogger unit. Deployed on truck MD620, Aug. 10, 2011 ......................................................................................................................... 26
Table 3-4- Open pit locations and their graph theory equivalents ............................................................. 28
Table 4-1- Parameters involved in haul time calculation ......................................................................... 51
Table 4-2- Statistical analysis of data presented in Figure 4-11 .................................................................. 69
Table 4-3- Statistical analysis of data presented in Figure 4-12 .................................................................. 69
Table 4-4- Statistical analysis of data presented in Figure 4-14, Route C (10to05) ................................. 75
Table 4-5- Statistical analysis of data presented in Figure 4-15, Route C (05to10) ................................. 75
Table 4-6- Comparison of haul time distributions for both directions of travel along Route C .... 76
Table 5-1- 6 most common hauls observed at Goldstrike in order of decreasing frequency .......... 84
Table 5-2- Groups of similar routes for effects of variability analysis .................................................... 85
Table 5-3- Comparison of simulated mean haul times for Group 3 haul routes before and after intersection "removal" ............................................................................................................. 113
Table 5-4- Most common hauls used as inputs for simulations ............................................................... 117
Table 5-5- Simulation Experiment 1- “Theoretical upper limit on productivity” ....................................... 120
Table 5-6- Simulation Experiment 2- "Introduction of variability based on observed levels at Goldstrike" ........................................................................................................................................ 120
Table 5-7- Simulation experiment 3- "Effects of intersection removal" ..................................................... 121
Table 5-8- Economic analysis of Experiment 2 results ............................................................................. 125
Table 5-9- Economic analysis of Experiment 3 results ............................................................................. 126
Chapter 1

Introduction

1.1 Motivation

Traffic is a phenomenon that has negative effects on its parent process. This is true for a wide range of systems that are transportation-based, such as computer networking, mail delivery and air travel. The mining industry is also affected; as mines continuously grow in size and scale, so do fleet sizes and in turn, traffic volumes. Any improvement in traffic flow will ultimately lead to higher productivities and increased revenues, while lowering fuel and maintenance costs. Specifically, here is a list of potential cost reduction elements achievable through traffic optimization:

- Reduction in truck requirement (i.e., fleet size) due to increase haulage efficiency;
- Reduction in diesel fuel usage due to reduction and/or elimination of intersection stops;
- Reduction in maintenance costs due to:
  - Decreased wear on brake pads due to reduction/elimination of intersection stops and/or other delays;
  - Decreased engine wear due to lower usage.

This broadly summarizes the motivation for this research: improving traffic conditions to yield a more profitable operation, with the added benefits of lower fuel costs, lower emissions, and in turn a reduction in environmental impact.

With this in mind, authorities from Barrick Gold Inc.—the world’s largest gold mining company—were keen to review their current operations with respect to traffic optimization. In
recent years, they have noticed an increase in traffic problems at their larger open pit mines. Today, some of their open pits require upwards of 30 haul trucks to sustain production, resulting in increased traffic congestion on their haul road networks. The effects of traffic were tangible, and reflected in rising operating costs such as fuel, tires and engine wear. In addition, traffic interactions between trucks were being observed at an increased rate, suggesting that material movement was being carried out sub-optimally. In 2009, authorities from Barrick’s corporate office decided to fund academic research in the field of traffic optimization. This led to a collaborative effort between industry and academia with the formation of the Mine Traffic Optimization Group (MTO). This team included 5 researchers at varying levels of study from Queen’s University and the University of Toronto, supervised by both academic advisors and Barrick representatives. Initially, the scope of the research was broad—as suggested by the group’s title—and the students were encouraged to pursue a topic of individual interest. After some discussion, the fields of research were narrowed down to traffic interactions, fuel economy, haul road profiling and human factors. This thesis focuses on open pit traffic interactions.

Barrick was willing to accommodate the necessary field research at any of their relevant sites worldwide, and made several suggestions based on the desired conditions. The Goldstrike site was eventually chosen for several reasons. Firstly, it met all the desired experimental conditions. Secondly, its relative ease of access was favorable, when compared to Barrick’s other suitable sites (i.e., the Veladero Mine in Argentina, or the Zaldivar Mine in Chile).

1.2 Context: Goldstrike Open Pit

The Goldstrike property is located on the Carlin Trend, a 60-km long north-northwest alignment of sedimentary rock-hosted gold deposits in Northern Nevada [1]. The closest towns are Carlin (40 km South) and Elko (60 km Southwest) as seen in Figure 1-1. There are several
other mines in the area, owned and operated by either Barrick or Newmont. At Goldstrike, the mining activity includes both open pit and underground extraction methods on adjacent ore zones. The research presented herein was conducted exclusively at the Betze-Post Open Pit Mine, and so details about the underground operations have been omitted. For simplicity, this site is referred to as the “Goldstrike Open Pit” or “the Open Pit” in this thesis.

Figure 1-1- Map showing location of Carlin-type deposits (black dots), including the specific location of the Goldstrike Mine and its relative position to the towns of Carlin and Elko [1]

Barrick acquired the Goldstrike property from Western Nevada Minerals in 1986. The following year marked the discovery of two sulphide ore zones, which were subsequently named the Betze and Deep Post deposits. The mine entered production in 1993 as a conventional truck-and-shovel operation using diesel/electric trucks and electric shovels. The mine has been in
continuous operation since then, and although it has expanded operationally since its early days, the mining methods remain largely unchanged. Its steady expansion through the 1990’s generated large revenues that provided Barrick with the financial flexibility to grow internationally and solidify its position as a world-class gold company. In 2010, the Betze-Post Open Pit (Figure 1-2) produced 1.24 million ounces of gold at total cash costs of $530 per ounce [2], compared to an industry average of $613 per ounce [3]. Goldstrike’s proven and probable mineral reserves as of December 31, 2010 were 12.6 million ounces.

![Figure 1-2- the Betze-Post Open Pit at Goldstrike](image)

**Figure 1-2- the Betze-Post Open Pit at Goldstrike [4]**

1.2.1 Fleet

There is a wide range of equipment needed to carry out mining activities in an open pit such as Goldstrike. The required fleet consists of both heavy- and light-duty equipment. Heavy-
duty equipment includes off-highway, mining-specific vehicles and machinery such as shovels, trucks, drills, graders and dozers. The shovels and trucks are directly responsible for primary ore production, used in the early stages of mineral extraction and transportation. The shovels dig the freshly blasted rock, either ore or waste, and load it into haul trucks that transport the material to appropriate dumping points. The Goldstrike fleet is made up of five shovels of varying size, and 33 identical trucks [4] as outlined in Table 1-1 (see Appendix A for technical specifications).

Light-duty units are standard vehicles such as pick-up trucks, semis and fuel trucks. The role of light-duty equipment is to support the continuous mining activity of heavy-duty equipment by performing tasks such as personnel and tool movement, in-pit maintenance and surveying work.

Table 1-1- Goldstrike Truck and Shovel Fleet Details [5], [6]

<table>
<thead>
<tr>
<th># Units</th>
<th>Manufacturer</th>
<th>Model</th>
<th>Type</th>
<th>Payload/capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shovels</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>P&amp;H</td>
<td>4100</td>
<td>Rope</td>
<td>45.9 cu. m.</td>
</tr>
<tr>
<td>2</td>
<td>P&amp;H</td>
<td>2800</td>
<td>Rope</td>
<td>35.7 cu. m.</td>
</tr>
<tr>
<td>1</td>
<td>Hitachi</td>
<td>5500</td>
<td>Hydr.</td>
<td>27.0 cu. m.</td>
</tr>
<tr>
<td>Trucks</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>Komatsu</td>
<td>930E</td>
<td>Diesel-Elec.</td>
<td>300.0 metric tons</td>
</tr>
</tbody>
</table>

1.2.2 Road Network

The Goldstrike Open Pit is accessible by a network of haul roads. These roads are specifically designed to accommodate heavy-duty equipment: although haul trucks are the most frequent users of these roads, they must also be suitable for shovels, drills, etc. As of January 1, 2011, haul roads are designed at a total width of 140 feet, including 15 foot safety berms on either edge. This leaves a designed 110 feet of usable road width for traffic. Before these new design standards took effect, the roads were narrower, with 90 ft. usable width. Therefore the road network has a variety of widths, depending on the age of any particular road section. All roads
can accommodate two-way truck traffic, with the exception of a 1 km stretch on the North side of the pit which is one-way Eastbound.

**1.2.3 Productivity**

In the mining business, revenue is directly proportional to ore productivity. This metric can be defined in different ways, but is almost always described by units of mass per unit time. For example, one can describe a mine’s output in tons per hour—an average of how much new material is extracted or moved in sixty minutes. Depending on the application, mine authorities can calculate the productivity of different types of materials such as ore and waste. Many mines try to achieve a certain ore/waste productivity ratio that satisfies both short-range and long-range production targets. The key concept here is that of material flow: the rate at which the truck and shovel fleet is moving material from the active mining benches to other locations in the mine. With the assumption of adequate capacity at the mineral processing facilities, as the flow of valuable material (ore) to the crusher increases, so does gold production and subsequent sales revenue. Ore productivity or ore flow (both terms are used interchangeably throughout this thesis) is a high priority metric for mine authorities, and its optimization is desired.

As discussed, the two main variables involved in productivity are material tonnages and units of time. It is important to note that ore flow is expressed as a continuous variable even though the truck and shovel mining process is based on moving discrete amounts of material at a time. Any given truck has an upper limit on how much ore it can carry (i.e., maximum capacity) per haul. It is desirable to always carry a quantity of material that approaches this maximum capacity, but the actual amount will vary from load to load. For example, material characteristics (i.e., density) will play a role in how many tons are moved per haul; shovel operators with varying degrees of expertise and/or experience might load the trucks differently. Another source
of variability is the actual time it takes for trucks to move around in the pit. This variable is often measured as cycle time, which represents the total duration of a complete truck haulage cycle: loading, travelling, dumping, travelling back, queuing, and spotting at the shovel. Any reduction in this cycle time, regardless of origin and destination, will increase overall productivity. Many factors are involved such as road conditions, network topology, load times, shovel queues, engine health, tire pressures and traffic disruptions. For these reasons, it can be argued that material flow is affected by many underlying parameters with varying degrees of influence.

1.2.4 Problem Statement: Haul Times

The Goldstrike open pit employs a large fleet of trucks that travel on a complex network of haul roads. Variability in haul times has been observed at Goldstrike in recent years. Can this variability be quantified? If so, can we identify its sources? Are current systems able to account for variability during dispatch optimization? Could we obtain a different result from a dispatch optimization using a stochastic approach and is this result better or worse? Is there a quantifiable economic impact?

1.3 Scope of Work

The scope of this thesis includes, but is not limited to, the following items:

- A literature review of topics pertaining to dispatch optimization, stochastic methods in linear programming, statistical analysis and state-of-the-art in current dispatch systems;
- Description of field research, experiment methodology, procedure, tools and resources used, description of experiment conditions;
- Generation of data sets reflecting variability in haul times over identical routes;
- Data analysis and simulations to support a discussion of different effects on dispatch optimization as well as any downstream economic impacts;
• Recommendations for current system users and suggestions for future research.

1.4 Thesis Format

The chapter structure herein relates back to the five main research questions presented in Section 1.2.4. Each chapter includes its own literature review, background, analysis and results. The questions/chapters were ordered hierarchically, so that each chapter expands on its predecessor. The objective is to present the research results in an accessible, structured and ultimately digestible manner. In this way, the discussion begins with an introduction to dispatch systems, with a focus on the system used at the Goldstrike Open Pit.
Chapter 2

Dispatch Systems

2.1 Fleet Management

The concept of optimized fleet management was first introduced by Dantzig and Ramser in 1959 [7]. They were concerned with the optimum routing of a fleet of gasoline delivery trucks between a bulk terminal and a large number of service stations. Since then, this field of research has been applied to a wide variety of relevant industries, such as emergency services [8], courier and delivery services [9] and of course open pit mining [10], [11], [12]. This chapter provides an overview of fleet management optimization strategies specific to the mining industry and sets the stage for the remaining chapters.

The application of optimized fleet management in mining started in the 1970’s. At the time, certain open pit mines such as the Mt. Wright iron ore mine in Quebec [13] and the Palabora copper mine in South Africa [14] had grown into large scale operations that employed large truck fleets and relatively complex haul routes. In order to remain competitive in the mining industry, researchers saw the need to move away from conventional radio/visual dispatching methods. The adequacy of this type of fleet management was being questioned in the early years of the computer age. Therefore, these aforementioned mines became testing grounds for some of the first computer-aided linear optimization exercises in the industry, using programming languages such as FORTRAN [15], [16]. At the forefront of this research was a small company based out of Tucson, Arizona called Modular Mining Systems (MMS) Ltd. They are credited with the deployment of the world’s first computer based mine management system at the Tyrone Mine in 1982 [11]. By 1993, MMS was manufacturing, installing and supporting its computer-
based DISPATCH® system at mines worldwide [17]. Today, there are several major vendors competing in the fleet management market, such as Caterpillar, Leica Geosystems/Jigsaw, MMS and Wenco. Many of these vendors offer full ‘suites’ of mine management software, reaching beyond the scope of this thesis into the realms of ore blending, mine planning, reliability, etc. However, in terms of truck/shovel dispatching and optimization, virtually all of the products currently available are based on the same underlying principles.

The goal of fleet management systems is to maximize productivity by implementing a variety of strategies, based on a combination of heuristic and algorithmic approaches. Heuristics are experience-based techniques for problem-solving, often referred to as “rules of thumb” or simply “common sense”. Their appeal stems from their potential to produce quickly near-optimal solutions to optimization problems [18]. A purely heuristic approach is often adequate for small mining operations, but less reliable for complex, large-scale mining operations such as Goldstrike. Such mines also require algorithmic support to make use of the vast amounts of highly-dynamic data for optimization [19].

The system that controls, manages and implements these strategies is generally referred to as a fleet management system or a dispatch system (these terms are used interchangeably throughout this document). The main goal of a such a system is to help mines optimize efficiency of their loading and haulage resources [20]. To this end, White et al. [11] and Hagenbuch [12] proposed that two of the principal objectives are maximum shovel utilization (i.e., minimum shovel idle time) and maximum truck utilization (i.e., minimum truck queue time). Hagenbuch describes these as the “two key premises of truck dispatching” as reproduced below:

1. “Where can a hauler go to be loaded the quickest (it needs to be fully comprehended and understood that haulers are only productive when they are carrying a load)? Empty haulers are the essence of nonproductive equipment. In the highway trucking industry,
any empty travel is called dead heading and justly so likewise in mining. Yes, some dead heading is always going to occur in a mining environment, but the point is it has to be minimized.

2. What piece of equipment will need a hauler soonest (with the corollary being where can a hauler get loaded the quickest)? It needs to be understood that loading equipment is also only productive when loading material for haulage. Idle loading equipment is again the essence of nonproductive equipment.” [12]

These key principles of truck dispatching were written in 1987 but remain as pieces of core logic in all dispatching systems. Alaire and Gamache [21] summarize by stating that the goal of the dispatch system is to answer one question each time a truck leaves a site in the mine: “Where should this truck go now?” This section attempts to describe this seemingly simple yet paradoxically complex question.

2.1.1 Components of a Dispatch System

There are two major components to any dispatch system: the dispatch criteria and the dispatch strategy. Dispatch criteria are chosen based on the desired metric to be optimized within the mining process. Some examples are: Minimizing shovel wait time, minimizing truck cycle time, or minimizing deviation from the shovel production target. Each of these criteria has its own benefits and drawbacks, and work done by Tan and Ramani [22] has shown that the best rule is highly dependent on the characteristics of the mine in question, especially the ratio of trucks to shovels [23].

The dispatch strategy is the approach used to meet the desired dispatching criteria. The two main strategies are denoted as single-stage and multi-stage by Alaire and Gamache [21]. Single-stage strategies are based on heuristic methods and do not acknowledge specific production targets or constraints. These were the precursors to multi-stage systems and are no longer used in
the industry. Modern dispatch systems employ a multi-stage strategy, which divides the global dispatching problem into sub-problems or stages. Typically the problem consists of an upper stage, used for calculating best paths and production targets, and a lower stage for truck allocation[21]. This is the approach adopted by the DISPATCH® software made by MMS, which is deployed at Goldstrike. The functioning of this system is described in the following section.

2.1.2 MMS DISPATCH®

The DISPATCH® system is a large scale, computer-based mine management system that controls the dispatching of all haul trucks in any open pit [11]. It is based on a set of algorithms that require a comprehensive set of data, consisting of both real-time information (collected continuously during operation), and quasi-static data based on mine specific parameters (usually inputted and updated frequently by mine dispatchers). White et al. [17] provide a list of what is required:

- Haul road information such as positions, elevations and distances;
- Truck positioning data (from GPS);
- Cycle time information such as truck loading and dumping times;
- Operational states of trucks and shovels;
- Miscellaneous mining constraints such as shovel priorities, dump capacities, truck capacities, and scheduled operator breaks.

Due to the proprietary nature of the MMS DISPATCH® software, a full set of details pertaining to its dispatching method are not available in the literature. Attempts by the author to obtain further information were fruitless. According to MMS authorities, they have not and do not publish much in the way of technical details due to the competitive nature of their business. Therefore, the bulk of what is publicly available in terms of technical details is rather outdated,
with papers published by MMS such as White et al. in 1982 [11] and White et al. in 1993 [17]. Other publications such as [24] and [25] summarize the content of these papers. Despite the lack of disclosure, the author was told by a current MMS authority that the fundamental architecture of the DISPATCH® software has remained largely unchanged since its inception [26].

As stated previously, DISPATCH® uses a multi-stage approach. The three stages are Best Path determination (BP), Linear Programming (LP) for each significant change in a time-dependent variable, and Dynamic Programming (DP) for truck assignment in real-time. Figure 2-1 shows a flow chart describing the inputs and outputs for each module and how they interact.
Figure 2-1- DISPATCH® algorithms (adapted from [25])
Initially, the BP stage calculates the shortest paths between all pairs of locations along the mine road network using Dijkstra’s shortest path algorithm [27], and uses a travel time correlation function to translate haul grades (slopes) and distances to equivalent travel times. As trucks travel the haul roads, DISPATCH® logs the actual travel times and uses them to update this parameter. The underlying calculations for travel times are not in the public domain. However, two independent sources from Barrick describe their understanding of the system as follows. Each truck records GPS readings at 30-second intervals and sends this information to the main server. The travel times are calculated using these data along with arrival/departure event flags. These values are stored in arrays—one for each active haul leg. Finally, the estimated travel time used by the LP module is updated using a moving back average of the four most recent travel times along each active route [28], [29].

The LP inputs the routes and travel times calculated in BP and calculates a theoretical solution to the dispatch problem, with respect to the various site-specific constraints (refer to Figure 2-1). The Simplex method [30], a common algorithm for linear programming, is used to minimize a prioritized truck haulage function based on these inputs and constraints [17]. The output contains optimized production circuits that run on the best travel routes between shovels and dumps, the recommended truck types for each route, and the required material flow rates for each route in tonnes per hour [25]. These flow rates essentially dictate the number of trucks required when truck capacities are known. The LP recalculates whenever any major changes happen within the system as listed in Figure 2-1, but mine authorities can also define a time for periodic recalculation at their discretion [29].

The function of the DP module is to execute LP’s optimized plan in real-time by assigning trucks along the paths to achieve the specified flow rates. It inputs a current snapshot of the pit
configuration, flow rates from LP, and an estimate of expected future assignments, and outputs truck assignments in real time. Again, the details of how DP calculates the estimate of expected future assignments are not in the public domain.

2.2 Variability in Current Dispatching Systems

There is no indication in the literature of any acknowledgement of haul time variability in the dispatch algorithms employed at Goldstrike, or anywhere else. Again, since the details of haul time calculations are not available, this assumption is based on the information available and observations made on site.

First, consider the way in which the estimated haul times are calculated and used in the optimization algorithm. Every time a truck completes a given haul leg, a corresponding haul time updates a vector containing the four most recent hauls for this particular route. The average of these four hauls is used to estimate the next travel time for the optimization algorithm. An example of this method is shown in Figure 2-2 for 100 simulated haul times. Since there is variability in the system, the moving back average fluctuates depending on the outcomes of the four previous hauls. The use of this method implies the assumption that future travel times are somehow dependent on past outcomes. Field observations contradict this assumption. For example, consider a haul route with a 4-way intersection along the route. The intersection does not require traffic travelling along this route to stop, only to slow down and yield to cross traffic. At time $t$, truck is observed stopping at the intersection for 120 seconds while waiting for a grader to cross. The next three trucks slow down at the intersection, but cross unimpeded since there is no cross-traffic. Therefore, the truck observed at time $t$ took significantly longer than the next three trucks, due to the stop at the intersection. A back average will include this anomalous haul time, and adjust the haul time estimation accordingly. Does the fact that one truck stopped at
some point in the past have any influence on future haul times? One could argue that, in fact, haul times are time-independent, conditional only on the present state of the system. The process can be thought of as ‘memoryless’; in other words, one could make predictions for the future of the process based solely on its present state just as well as one could with historical knowledge.

Figure 2-2- Example of moving back average calculation for haul time estimation
Chapter 3

Measuring Variability in Haul Times

The goal of this chapter is to present results about the measurement of variability in truck haul times at the Goldstrike Open Pit mine. In order to select an appropriate and effective measurement technique, a review of existing methods was first conducted.

Variability in travel times has been measured in the public transport industry using different data collection methods. These data were used to support analyses of different facets of each system. An influx of relevant studies in the 1980’s focused on the reliability of public transport systems, as seen in [31] and [32]. The data for these early studies were collected by passengers on a trip-by-trip basis using basic stopwatches. As research progressed, new techniques emerged that showed promise for this field. In 1995, Turner [33] provided an overview and comparison of the available techniques for travel time data collection for freeway traffic. His findings are shown in Table 3-1, where seven techniques are compared based on four main criteria. Turner identifies the main methods as being either active or passive in nature.

Passive methods estimate travel time by calculating the elapsed time for a target vehicle to travel between fixed way-points. The majority of the documented techniques fall in this category. An example is the Automatic Vehicle Identification (AVI) method, where probe vehicles equipped with on-board transponders are detected and time-logged at fixed waypoints along the roadway. Alternatively, travel times can be estimated using active detection methods. Examples are Automatic Vehicle Location (AVL) and electronic Distance Measurement Instrument (DMI) methods. AVL relies on positioning technology such as GPS to monitor and record real-time changes in position during travel, whereas DMI determines travel times along a corridor based
upon speed and distance information provided by an instrument connected to the transmission of the test vehicles [34].

Table 3-1- Comparison of Advanced Travel Time Collection Techniques (reproduced from [33])

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Travel Time Collection Technique</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Electronic DMI</td>
</tr>
<tr>
<td></td>
<td>Computerized License Matching</td>
</tr>
<tr>
<td></td>
<td>Video License Matching</td>
</tr>
<tr>
<td></td>
<td>Cellular Phone Tracking</td>
</tr>
<tr>
<td></td>
<td>AVI</td>
</tr>
<tr>
<td></td>
<td>AVL</td>
</tr>
<tr>
<td></td>
<td>Video Imaging</td>
</tr>
<tr>
<td>Field-Proven Technology</td>
<td>Yes</td>
</tr>
<tr>
<td>Accuracy of Travel Times</td>
<td>Good</td>
</tr>
<tr>
<td>Capital Costs</td>
<td>Low</td>
</tr>
<tr>
<td>Operating Costs</td>
<td>Moderate</td>
</tr>
</tbody>
</table>

The acceptance and widespread use of new technology over the past twenty years has improved the availability of active travel time measurement methods. For instance, the rapid growth in cellular phone use in the late 1990’s provided a powerful new tool for researchers in this field. Early studies [35] focused on using the frequent communication signals between cellular phone and service towers as a source for positioning data and in turn, travel time estimation. To address the initial concerns that these cellular data might be too sparse and/or unreliable for this purpose, Bar-Gera [36] compared a data set obtained from cellular phone location reporting to conventional vehicle detector data, and found the results to be almost identical, stating that “the cellular phone data set appears to be suitable for usage in practical applications”.

Alternatively, GPS technology has become an indispensable tool for travel time data collection due to its high accuracy, decreasing cost of deployment, and ease of use [37]. Yim and
Cayford [35] validated this statement in their 2001 report, recommending GPS not only for its accuracy, but also its flexibility—they highlight its adjustable sampling rate as one of the major advantages over the other available methods. This was the method used in a wide variety of studies including system reliability [38] and routing choice [39] for public transit, truck fleet dispatching [37], and even in air traffic control [40].

Open pit mining operations lend themselves well to GPS implementation. Since 2000, virtually all such mines use GPS to generate databases of position data that are frequently used to estimate travel times [41]. The Highland Valley Copper Open Pit Mine located near Logan Lake, BC is an example of a large-scale mine that took advantage of GPS to increase the functionality of their dispatching system, increase truck productivity and provide more information for management and supervision [19]. This was a worldwide phenomenon—similar results were reported from GPS implementation at several bauxite mines in India [42]. At present, all commercialized fleet management systems use GPS, such as MMS and Leica Geosystems Ltd. The general role of such systems is to increase productivity and minimize idle time for the haulage fleet [43]. MMS deploys its DISPATCH® fleet management software at nine of the ten largest surface mining operations in the world [43], and uses GPS for all of its travel time estimations. Leica’s Jigsaw® software is the second most popular fleet management software among open pit mines, and is similarly reliant on GPS for position and travel time estimation [44].

3.1 Experiment Design

3.1.1 Data Requirements

For this research, travel time data were collected from the Goldstrike Open Pit mine by extracting this information from GPS logs. In order to address the proposed research questions, it
was decided that the data should be collected at a large-scale truck and shovel operation with the following general characteristics: multiple shovels and multiple dumps, a complex road network with intersections, and a large number of trucks on the roads at any given time (i.e., more than 20). The Goldstrike site, introduced in Section 1.2, met the desired criteria and was the chosen location for data collection.

Several methodologies were originally considered to gather information about haul times. At the operational level, mine authorities often perform time studies manually. This involves sitting at a certain location in the mine—typically a shovel or dump—and timing arrivals of trucks using a stopwatch. Although this method can generate cycle times for several trucks over a given time period, it does not provide any information about the route travelled, nor the disruptions that might have affected the cycle duration. Furthermore, it is a tedious process requiring a significant number of field hours for a relatively small amount of data. For these reasons it was deemed inadequate for the purpose of this research.

Instead of simply gathering cycle times, it was decided that truck positioning data would not only be more useful, but also easier to collect. Furthermore, as discussed previously, this method of travel time collection has been widely used in many different fields with good results. The main concern with using this approach was that it would need to be collected at a high frequency in order to be useful, on the order of 1-2 Hz (i.e., position readings per second). This would provide a rich data set with sufficient resolution to track positions, approximate velocities and even accelerations. With this in mind, two options were identified for collecting this data. Option 1 was to use standalone GPS receivers/dataloggers, while Option 2 was to modify and use the existing infrastructure and datalogging capabilities at Goldstrike. Both options had advantages and disadvantages as shown in Table 3-2.
Table 3-2: Comparison of available data collection techniques using GPS

<table>
<thead>
<tr>
<th>Data Collection Options</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Option 1: ‘Standalone’ GPS/datalogger units</td>
<td>• Generation of unique, tailored, ‘first-hand’ data</td>
<td>• Cost</td>
</tr>
<tr>
<td></td>
<td>• Guarantee of desired sampling rate (1 Hz)</td>
<td>• Requires physical installation/removal</td>
</tr>
<tr>
<td></td>
<td>• Ease of use/data dumping</td>
<td>• ‘Discrete’ sampling of time blocks, constrained by battery life/storage capacity</td>
</tr>
<tr>
<td></td>
<td>• Requires minimal involvement of GS Engineers</td>
<td></td>
</tr>
<tr>
<td>Option 2: Make use of existing infrastructure (DISPATCH®)</td>
<td>• No installation/removal requirements</td>
<td>• No guarantee of desired sampling rate</td>
</tr>
<tr>
<td></td>
<td>• Potential of monitoring any/all trucks simultaneously</td>
<td>• Requires significant involvement of GS Engineers</td>
</tr>
<tr>
<td></td>
<td>• No associated cost</td>
<td>• ‘Second-hand’ data from proprietary system</td>
</tr>
</tbody>
</table>

It was decided that both options were feasible and would be attempted for data collection. Since the data collection was to be done on site and neither of these methods could be fully tested or validated prior to departure, it was deemed prudent to retain both options as possibilities depending on how things went at Goldstrike. Both options are discussed in the following subsections.

3.1.2 Data Collection

3.1.2.1 Qstarz Units

Five GPS units were purchased for the MTO project, three of which were exclusively used for this research. The units are made by Qstarz, model BT-Q1000XT as shown Figure 3-1. This specific model was chosen for several reasons: portability, sample rate, battery life, storage
capacities, GPS chipsets and ease of use. The units are small and portable, roughly half the size and weight of a mobile phone, making them very easy to manipulate and install temporarily on haul trucks. The units were capable of recording at 1 Hz, which was sufficient for this research. At this sampling rate, the unit could record upwards of 50,000 data points per charge, lasting approximately 14 hours. This was a convenient length since it was long enough to span part of a day shift and a night shift. Finally, this model uses the MTK II GPS chipset, which combines a high sensitivity rating and reliable GPS fixing with low power consumption [45].

![Image](image.jpg)

**Figure 3-1- Qstarz GPS receiver/datalogger used for data collection, model BT-Q1000XT**

3.1.2.2 MMS GPS data

As described in section 3.1- Experiment Design, Option 2 was to make use of the existing infrastructure and database at Goldstrike to collect relevant GPS data. Due to the proprietary nature of MMS, the dispatch engineers at Goldstrike were skeptical about whether it would be possible to collect high frequency GPS data using the existing setup. There was some uncertainty
as to whether the Modular code could be customized to this end. The desired data would need to be collected somewhere in the order of 1-2 Hz; the default set up is for trucks to log their position every 30 seconds (0.033 Hz). Overall, this option seemed risky and was considered as a “Plan B” in case the standalone units did not work as expected.

After some brief correspondence with authorities at MMS, they provided directions to change the GPS logging frequency in their code. Furthermore, they pointed out that it was possible to increase the GPS sampling rate on a truck-by-truck basis. This was favorable, because the prospect of collecting data from all trucks raised concern over bandwidth usage on the mine’s wireless network.

3.1.3 Methodology

The Qstarz units were deployed as frequently as possible on as many different trucks as possible, in an attempt to generate a representative sample of the truck fleet at Goldstrike. The installation was simple and quick. While pit operations were winding down just before each daily blast, most trucks would park at a staging area to break while the shot was fired. This was a convenient time to access the trucks, avoiding any unnecessary downtime during normal haulage operations. Once a truck was chosen, the units were activated and duct-taped to its front railing on its center axis. This installation location was chosen after some trial and error—it was the easiest to access and had the best overall view of the sky to ensure optimal GPS coverage. The units would record until their storage limit was reached, at which point they would shut off automatically. The units were collected the following day, dumped and recharged, before being re-deployed.

The MMS data were collected and stored automatically per the usual functioning of the DISPATCH® system. The sampling rate of 1 Hz was achieved for trucks MD623, MD624 and
MD635 from July 26 – August 11, 2011 inclusive. The Author collaborated with Goldstrike engineers to write a reporting script that pulled the relevant data from the main database. This yielded a large quantity of continuous data that looked promising. However, the quality of the data was much lower than the Qstarz data. The first major issue was significant ‘jumping’ of GPS readings at prolonged stops—for these events, the recorded data were incoherent, with consecutive position readings coming from all over the pit area. This was not desirable, but tolerable as long as the rest of the data were sound. This was not the case—there was also a precision issue in reported truck positions. The observed GPS traces for the trucks were accurate in terms of movement, but were consistently off the haul road by approximately 50 metres. This phenomenon was investigated, but was difficult due to the proprietary nature of the DISPATCH® software. Based on some testing at different sampling rates, it was theorized that the imprecisions were being caused by a conflict between the abnormally high sampling rate of 1 Hz and the data transfer protocol between trucks and mainframe. Due to time constraints, the inaccessibility of the Modular code and the continued success of the Qstarz data, the problem was not investigated further. However the data were collected and kept, having recognized that they might still be of some use at a later stage of the research.

3.2 Qstarz Data

In total, 31 deployments resulted in 31 files with a total 457 MB of generated data. A sample of the raw data is shown in Table 3-3. Each file contains approximately 14 hours of 1 Hz GPS data or roughly 180 kilometres of truck haulage. This data set spans the entire field trip duration, June 23 – August 10, 2011. It is important to note that this data set is discontinuous—not only does it represent many different trucks, but it only records 14 hours of motion per deployment.
Table 3-3- Data sample from a Qstarz GPS receiver/datalogger unit. Deployed on truck MD620, Aug. 10, 2011

<table>
<thead>
<tr>
<th>INDEX</th>
<th>LOCAL DATE</th>
<th>LOCAL TIME</th>
<th>LATITUDE</th>
<th>N/S</th>
<th>LONGITUDE</th>
<th>E/W</th>
<th>HEIGHT</th>
<th>SPEED</th>
<th>HEADING</th>
<th>PDOP</th>
<th>HDOP</th>
<th>VDOP</th>
<th>NSAT(USED/VIEW)</th>
</tr>
</thead>
<tbody>
<tr>
<td>46328</td>
<td>2011/08/11</td>
<td>02:00:00</td>
<td>40.97738398</td>
<td>N</td>
<td>116.3914337 W</td>
<td>1397.936 M</td>
<td>35.123 km/h</td>
<td>312.67099</td>
<td>3.55</td>
<td>2.39</td>
<td>2.63</td>
<td>6/10</td>
<td></td>
</tr>
<tr>
<td>46329</td>
<td>2011/08/11</td>
<td>02:00:01</td>
<td>40.97744588</td>
<td>N</td>
<td>116.3915202 W</td>
<td>1397.539 M</td>
<td>36.250 km/h</td>
<td>314.879822</td>
<td>1.93</td>
<td>1.69</td>
<td>0.93</td>
<td>6/10</td>
<td></td>
</tr>
<tr>
<td>46330</td>
<td>2011/08/11</td>
<td>02:00:02</td>
<td>40.97751384</td>
<td>N</td>
<td>116.3916038 W</td>
<td>1397.057 M</td>
<td>37.288 km/h</td>
<td>314.866008</td>
<td>1.93</td>
<td>1.69</td>
<td>0.93</td>
<td>6/10</td>
<td></td>
</tr>
<tr>
<td>46331</td>
<td>2011/08/11</td>
<td>02:00:03</td>
<td>40.97758227</td>
<td>N</td>
<td>116.3916917 W</td>
<td>1396.730 M</td>
<td>38.573 km/h</td>
<td>314.395905</td>
<td>1.93</td>
<td>1.69</td>
<td>0.93</td>
<td>6/10</td>
<td></td>
</tr>
<tr>
<td>46332</td>
<td>2011/08/11</td>
<td>02:00:04</td>
<td>40.97765216</td>
<td>N</td>
<td>116.3917837 W</td>
<td>1396.410 M</td>
<td>39.549 km/h</td>
<td>314.900238</td>
<td>1.93</td>
<td>1.68</td>
<td>0.93</td>
<td>6/11</td>
<td></td>
</tr>
<tr>
<td>46333</td>
<td>2011/08/11</td>
<td>02:00:05</td>
<td>40.97772321</td>
<td>N</td>
<td>116.3918775 W</td>
<td>1396.329 M</td>
<td>40.190 km/h</td>
<td>315.375275</td>
<td>1.93</td>
<td>1.69</td>
<td>0.93</td>
<td>6/11</td>
<td></td>
</tr>
<tr>
<td>46334</td>
<td>2011/08/11</td>
<td>02:00:06</td>
<td>40.97779679</td>
<td>N</td>
<td>116.3919713 W</td>
<td>1396.374 M</td>
<td>40.893 km/h</td>
<td>316.329956</td>
<td>1.93</td>
<td>1.69</td>
<td>0.93</td>
<td>6/10</td>
<td></td>
</tr>
<tr>
<td>46335</td>
<td>2011/08/11</td>
<td>02:00:07</td>
<td>40.9778717</td>
<td>N</td>
<td>116.3920645 W</td>
<td>1396.407 M</td>
<td>41.372 km/h</td>
<td>317.158722</td>
<td>1.93</td>
<td>1.68</td>
<td>0.93</td>
<td>6/10</td>
<td></td>
</tr>
<tr>
<td>46336</td>
<td>2011/08/11</td>
<td>02:00:08</td>
<td>40.97794962</td>
<td>N</td>
<td>116.3921532 W</td>
<td>1396.332 M</td>
<td>41.237 km/h</td>
<td>319.088104</td>
<td>1.93</td>
<td>1.69</td>
<td>0.93</td>
<td>6/10</td>
<td></td>
</tr>
<tr>
<td>46337</td>
<td>2011/08/11</td>
<td>02:00:09</td>
<td>40.97802862</td>
<td>N</td>
<td>116.3922404 W</td>
<td>1396.325 M</td>
<td>40.840 km/h</td>
<td>319.536224</td>
<td>3.55</td>
<td>2.39</td>
<td>2.62</td>
<td>6/10</td>
<td></td>
</tr>
<tr>
<td>46338</td>
<td>2011/08/11</td>
<td>02:00:10</td>
<td>40.97810952</td>
<td>N</td>
<td>116.3923265 W</td>
<td>1396.298 M</td>
<td>40.536 km/h</td>
<td>320.443726</td>
<td>1.93</td>
<td>1.68</td>
<td>0.93</td>
<td>6/10</td>
<td></td>
</tr>
<tr>
<td>46339</td>
<td>2011/08/11</td>
<td>02:00:11</td>
<td>40.97819052</td>
<td>N</td>
<td>116.3924097 W</td>
<td>1396.516 M</td>
<td>38.366 km/h</td>
<td>322.155579</td>
<td>4.03</td>
<td>3.06</td>
<td>2.62</td>
<td>5/10</td>
<td></td>
</tr>
<tr>
<td>46340</td>
<td>2011/08/11</td>
<td>02:00:12</td>
<td>40.97826664</td>
<td>N</td>
<td>116.3924824 W</td>
<td>1396.761 M</td>
<td>34.812 km/h</td>
<td>324.654633</td>
<td>4.03</td>
<td>3.06</td>
<td>2.62</td>
<td>5/10</td>
<td></td>
</tr>
<tr>
<td>46341</td>
<td>2011/08/11</td>
<td>02:00:13</td>
<td>40.97833762</td>
<td>N</td>
<td>116.3925443 W</td>
<td>1396.862 M</td>
<td>32.381 km/h</td>
<td>327.196808</td>
<td>4.03</td>
<td>3.06</td>
<td>2.62</td>
<td>5/10</td>
<td></td>
</tr>
<tr>
<td>46342</td>
<td>2011/08/11</td>
<td>02:00:14</td>
<td>40.97840515</td>
<td>N</td>
<td>116.3925963 W</td>
<td>1396.664 M</td>
<td>30.910 km/h</td>
<td>332.098206</td>
<td>4.03</td>
<td>3.06</td>
<td>2.62</td>
<td>5/10</td>
<td></td>
</tr>
<tr>
<td>46343</td>
<td>2011/08/11</td>
<td>02:00:15</td>
<td>40.97847641</td>
<td>N</td>
<td>116.3926354 W</td>
<td>1396.605 M</td>
<td>30.404 km/h</td>
<td>338.141754</td>
<td>4.03</td>
<td>3.06</td>
<td>2.62</td>
<td>5/10</td>
<td></td>
</tr>
</tbody>
</table>
This data sample shows 15 data points representing a 15 second time span starting at August 11, 2011 2:00:00 AM, for truck MD620. The first eight columns contain information pertaining to time and position (latitude/longitude/elevation). The units also have a built-in speed and heading calculation, which was not used, since there was no information available from the manufacturer as to how they were calculated. The last four columns contain information about the GPS satellite coverage quality. PDOP, HDOP and VDOP are dilution of precision measurements describing overhead satellite positions relative to the GPS receiver on the Earth’s surface [46]. The last column shows how many satellites were used to estimate the receiver position. To obtain high accuracy GPS readings, lower dilution of precision (DOP) values and a high number of satellite fixes are needed. The entire Qstarz data set has DOP values in the ‘excellent coverage’ to ‘good coverage’ range, as described in [46], with 99% of the values less than 5.0. The units themselves have an overall accuracy of +/- 3.0 m according to the manufacturer specifications [45]. This combination of DOP values and overall GPS accuracy was assumed to be adequate for travel time research, since the trucks themselves are 15.6 m long and 9.2 m wide [47].

3.3 Data Analysis

Some preparation was needed before processing the Qstarz data. First, a model of the open pit environment was required. To this end, the open pit was treated as a network of unique destinations (i.e., shovels and dumps) connected by roads—see Appendix B for a complete graph representation of the haulage network. Some basic concepts are borrowed from graph theory to provide a simple yet powerful representation of the system\(^1\). Consider the definition of a graph, adapted from:

\(^1\) For background on graph theory, please see [71].
Definition. A graph $G = (V,E)$ consists of two sets $V$ and $E$. The elements of $V$ are called vertices (or nodes); the elements of $E$ are called edges.

We thus consider the open pit as a graph made up of nodes and edges. Any and all “locations of interest” within the open pit network are represented by nodes, while the roadways are represented by edges. Table 3-4 shows a list of open pit locations and their graph theory equivalents. This basic representation is important for the remainder of the data analysis, for it facilitates sorting of unique routes to and from unique node pairs.

**Table 3-4- Open pit locations and their graph theory equivalents**

<table>
<thead>
<tr>
<th>Open pit locations</th>
<th>Graph theory equivalent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shovel</td>
<td>Source node</td>
</tr>
<tr>
<td>Dump</td>
<td>Sink node</td>
</tr>
<tr>
<td>Queuing/Break area</td>
<td>Delay node</td>
</tr>
<tr>
<td>Intersection</td>
<td>Delay node</td>
</tr>
<tr>
<td>Fuel/wash bay</td>
<td>Delay node</td>
</tr>
<tr>
<td>Road</td>
<td>Edge</td>
</tr>
</tbody>
</table>

A list of all the “interesting” pit locations was compiled, to be represented as nodes in the pit graph network. A naming convention was adopted for nodes to facilitate their identification. Each node was given an eight character ID of the form XXXXYZ## where XXXX is a character string describing the location, Y is a single character describing associated material types, Z is a character string describing node type, and ## is a number of the form 01, 02, etc. allowing for multiple nodes with similar names. For example, the in-pit waste dump on the 5110 level is named 5110WD01: ‘5110’ describes the location, ‘W’ indicates that the associated material type is waste, ‘D’ indicates it is a dump, and ‘01’ indicates this is the first (in this case, the only) node with this particular designation. This naming convention was not extended to
edges, since for the most part they were identifiable using origin and destination nodes. There were a few exceptions, for example where multiple paths existed between two nodes; these were treated on a case-by-case basis, and given self-explanatory names. Refer to Appendix B for a complete list of nodes.

Mining plans and dispatch graphics were collected at regular intervals throughout the data collection period so that the positioning data could be analyzed in a real-world context.

3.3.1 MATLAB®

The data analysis was carried out in MATLAB®, a numerical computing software package based on a fourth-generation programming language [48]. It was chosen for its ease of use, comprehensive statistics package, and availability through Queen’s University. All of the MATLAB® code needed to manipulate the data was written by the Candidate. The suite of programs was divided into several modules, each with a specific function. A flowchart depicting these different modules is shown in Figure 3-2. The code is available in Appendix C.
3.3.2 IMPORT

The IMPORT module (Appendix C-1) consists of one main program that inputs MS Excel files containing raw data from the Qstarz units and saves them in a MATLAB®-friendly format.
This allows for an initial latitude vs. longitude plot of the entire data set, which can be compared to existing pit maps and graphics for node orientation and identification. See Figure 3-3 and Figure 3-4 for examples. Roads are very clearly defined in Figure 3-3, but the shovel and dump locations are less obvious, due to their slow yet constant movement over time. Regardless, the overall configuration of both figures is similar. They were compared visually to identify node locations in preparation for the next module.
Figure 3-3: Plot of all Qstarz truck position data (latitude vs. longitude)
3.3.3 POLYGONS

In this module (Appendix C-2), the relevant locations identified during IMPORT are delineated by user-defined polygons. The goals here:

1. Identify the different node types listed in Table 3-4;

2. Identify suitable arrival/departure boundaries for each node;
3. Delineate each node with a polygon that reflects these choices, so that the positioning data can be parsed accordingly.

This is performed by choosing coordinates surrounding the location in question on a plot similar to the one in Figure 3-3. This allows for custom definition of each node area. The final product is a new MATLAB® data file containing this polygon information, used from this point forward. The outcomes of the downstream modules (see Figure 3-2) are dependent on these polygon definitions, which essentially act as a filter for the raw data. Therefore, a number of polygon files were created to test different hypotheses and circumstances, and stored in separate locations. Figure 3-5 shows an example of a polygon definition file along with the data.
Figure 3-5- Polygons (from ac_polygons) superimposed on all Qstarz data
3.3.4 PARSE

The goal of this module (seen in Appendix C-3) is to organize the data in such a way that we are able to distinguish between haulage and node activity. Haulage activity is defined by a state of movement between two nodes along an edge, whereas node activity is defined as movement within a node, related to loading, queuing, spotting, tipping, etc. Haulage activity data are used to estimate haul times, while node activity provides node residence times.

The raw data are now sortable according to the newly defined polygons. This program parses the data points based on location; the results are several data subsets, one for each node polygon defined in the SPLIT ID module. The option to parse data using all polygons is default, but it is possible to exclude certain polygons as desired (i.e., at intersections, road sections, etc.). The data from within the polygons, or the node activity data, are stored for later analysis of node residence times. At this point, the remaining data consist of points lying outside the polygons, or the haulage activity data. As discussed, these are data corresponding to trucks travelling between origin-destination nodes (i.e., shovel and dump), along the corresponding edges (i.e., haul roads). An example of a parsed data set using a set of polygons is shown in Figure 3-6. The haulage activity data are copied to a new MATLAB® file, to be used as an input for the next Module: haul leg identification.
Figure 3-6- An example of parsed data, showing only points lying outside the polygons, i.e., those travelling between nodes

3.3.5 LEG ID

At this point, the haul activity has been identified, allowing us to focus on extracting haul time estimations from the positioning data. The haul activity data consist of many one-way haul paths between nodes—the functions of the LEG ID Module (see Appendices C-5 and C-6) are to:

1. Identify each of these paths based on origin node, destination node, and path;
2. Group the data for each haul leg by flagging departure time from origin node and entry time at destination node;
3. Calculate haul times by elapsed time between origin/destination node boundaries and estimate instantaneous velocities between consecutive data points;

4. Count all observed combinations of origin, destination and path, to use for aggregation of identical hauls;

5. Create a haul times database file for each unique origin/destination pair.

The result of this module is individual haul leg data, sorted according to origin node and destination node, with detailed information specific to each leg. The one second sampling rate yielded the following:

- Detailed path traces, useful for identifying anomalies in normal trajectory;
- Accurate estimation of instantaneous velocity, useful for identifying slow-down and stop events;
- Precise haul time measurements, necessary for statistical analysis.

3.3.6 ANALYZE

The ANALYZE Module (Appendices C-6, C-7, C-8 and C-9) consists of several programs that perform statistical analyses on the aggregated haul leg data from the leg identification process. This includes values such as mean, median and standard deviation. It also uses several built-in MATLAB® functions from the Statistics Toolbox to perform distribution fitting and subsequent goodness-of-fit tests. These programs can be inputted with a variety of variables, the most common being haul times, haul distances and node residence times. The results are useful for comparing situations such as haul legs between identical nodes using different paths.

3.3.7 Support Programs

A number of functions and programs were used alongside the main modules of the MATLAB® code for specific tasks. For example, several programs were created to perform
repetitive calculations, such as those needed to obtain velocities, road grades and incremental distances. Another important set of programs were made to visualize data—these were useful for debugging, data validation and reporting purposes. An example of a support program is shown in Appendix C-4.

3.4 Results of Variability Measurements in Haul Times

The culmination of this chapter is measurement of haul time variability in the data collected from Goldstrike. The results are demonstrated by examining the three most common one-way haul legs from the data collection period, as enumerated below:

1. Node 3 to 10 (Q3&DUMPS to SWSHOVLS): The empty haul from the Eastern in-pit dumps to the shovels in the Southwest;
2. Node 10 to 3 (SWSHOVLS to Q3&DUMPS): The loaded haul from the Southwest shovels to the Eastern in-pit dumps;
3. Node 8 to 10 (4630WD01 to SWSHOVLS): The empty haul from the 4630 in-pit dump to the Southwestern shovels.

The following sections describe each path individually along with their variability analysis. Histograms are presented to show data dispersion, and include key information pertaining to each haulage scenario. The parameters used to generate these plots were kept constant, using 60 second bins and unfiltered data. The goal of this chapter is to show that variability exists in the data set, and was quantifiable—subsequent chapters deal with causes and effects.

3.4.1 Analysis 1: Node 3 to 10 (Q3&DUMPS to SWSHOVLS)

This haul leg represents traffic along the south haul road between the Eastern in-pit dumps (Q3&DUMPS) and the active mining bench to the Southwest (SWSHOVLS). This haul leg was the most commonly observed, with 105 trips. This route is highlighted (pink) on a dispatch
screenshot in Figure 3-7 and on a topographic map in Figure 3-8. The total haul distance was 3517 m with an average decline of 4.7 %. A notable feature along this route is the switchback on the south road. Haul times were calculated and grouped to produce the histogram plot (frequency vs. haul times) seen in Figure 3-9.

Figure 3-7- Dispatch screenshot of pit road network taken on July 15, 2011, with the road section between Node 3 (Q3&DUMPS) and Node 10 (SWSHOWLS) highlighted
Figure 3-8- Topographic map of Goldstrike pit, July 13, 2011, with the road section between Node 3 (Q3&DUMPS) and Node 10 (SWSHOVLS) highlighted
The data in Figure 3-9 are grouped into eight bins, spanning 30 seconds each. The minimum recorded haul time was 314 seconds and the maximum was 548 seconds, with a mean haul time of 393 seconds. The standard deviation was measured to be 51 seconds, while the interquartile range (IQR) was 56 seconds\(^2\). A left skew is apparent from this visual representation.

3.4.2 Analysis 2: Node 10 to 3 (SWSHOVLS to Q3&DUMPS)

The second most common haul leg was between the Southwest shovels (SWSHVLS) and the Eastern in-pit dumps (Q3&DUMPS) along the south road. This represents the same haul path

\(^2\) The interquartile range (IQR) is centered on the mean and contains 50% of the data. For example, in this case the IQR is 56 and the mean is 393, so IQR includes haul times in the region defined by [365,421].
as seen in the previous case, but with traffic travelling in the opposite direction. A total of 86 trips were recorded. The haul path is similar to those seen in Figure 3-8 and Figure 3-11; however, the haul distance is slightly longer at 3548 m. This difference of 31 m is negligible and can be attributed to slight variations in road surface. The average grade is a 4.7 % incline. Figure 3-10 shows the haul time data in a histogram plot.

![Histogram plot for haul leg 10to03](image)

Figure 3-10- Histogram plot for haul leg 10to03

Figure 3-10 shows the data grouped into 10 bins of 30 seconds each. The minimum and maximum haul times were 651 seconds and 944 seconds, respectively. The mean haul time was 771 seconds, with a standard deviation of 54 seconds and an IQR of 65 seconds. A slight left-
hand skew is visible, but is much less pronounced than in the first analysis 1. The shape of the histogram plot is reminiscent of a normal distribution.

3.4.3 Analysis 3: Node 8 to 10 (4630WD01 to SWSHOVLS)

The third and final haul leg analyzed was distinct from the first two, although it used a portion of the same route. This particular case, the third most common with 71 recordings, represents traffic between the 4630 in-pit dump (4630WD01) and the Southwest shovels (SWSHOVLS) on the active mining bench. The total haul distance was 1595 m at an average decline of 2.2 %. Figure 3-11 and Figure 3-12 illustrate the haul path and Figure 3-13 shows the histogram plot of haul times.

![Image of pit road network with Node 8 and Node 10 highlighted]

Figure 3-11- Dispatch screenshot of pit road network taken on July 15, 2011, with the road section between Node 8 (4630WD01) and Node 10 (SWSHOVLS) highlighted
Figure 3-12- Topographic map of Goldstrike pit, July 13, 2011, with the road section between Node 8 (4630WD01) and Node 10 (SWSHOVLS) highlighted
Figure 3-13- Histogram plot for haul leg 08to10

The data in Figure 3-13 span 10 bins of 60 seconds each, with the shortest haul time being 115 seconds and the longest being 403 seconds. The mean haul time was 169 seconds, with a standard deviation of 45 seconds and an IQR of 30 seconds. Note the extreme values in the upper bins of the histogram distribution. Apart from these outliers, the data are tightly clustered around the mean, as demonstrated by the small IQR. Again, a left-skew is discernible from the plot.

3.5 Discussion

The histograms presented in this Chapter all look significantly different. The haul time distributions in Figure 3-9 and Figure 3-10 are skewed to the right, while in Figure Figure 3-13 the distribution looks relatively normal. The levels of variability are also significantly different,
ranging from 7% to 27% in terms of relative standard deviation (ratio of standard deviation to mean). These observations might lead one to ask: what are the distinguishing factors that cause such marked differences between hauls? Some are apparent such as haul distance and elevation changes. However, it is hypothesized that many other parameters have an influence on haul time variability. The next chapter focuses on identifying and quantifying the various sources of haul time variability within the studied Goldstrike Open Pit haulage scenario.
Chapter 4
Sources of Haul Time Variability

4.1 Literature Review

Sources of travel time variability have been identified in the air travel industry, and there has been some research into their overall effect on the system. A recent study by Ball et al. [49] found that the total cost of domestic flight delays in the U.S. was $28.9 billion, thus the motivation to reduce delays is apparent and certainly significant. The similarities between air traffic and mine traffic are perhaps unobvious, but relevant. The systems have similar components: airports can be thought as shovels and dumps, or source and sink nodes, while flight paths between airports are like haul roads. Thus, a portion of the literature focuses on air traffic optimization, since it shows significant similarities to open pit mine traffic optimization.

In 1987, Odoni [50] acknowledged that variability exists in both main stages of an airplane’s travel, namely the ground portion and the flight portion. He first introduced the concept of scheduling flights in real time to minimize congestion costs associated with these delays. His work led to the realization that a major source of variability and subsequent delays were related to aircraft ground-holding times. Many authors expanded upon his work by modelling problems and formulating solutions using deterministic optimization programming, such as Terrab and Odoni [51], Vranas et al. [52], and Richetta and Odoni [53]. Terrab and Paulose [54] take a different approach by addressing the stochastic air traffic flow management problem. They assumed that the capacities of the system (in this case, airports and sectors in the airspace) are probabilistic in nature; in essence, acknowledging the inherent variability present in air travel times. However, their results were not scalable to the complexity of real-life air traffic. In 1996, Bertsimas and

48
Patterson[55] managed to tackle large scale, realistic size problems consisting of several thousand flights, and concluded that “optimally controlling the flow of aircraft either by adjusting their release times into the network (ground-holding) or their speed once they are airborne is a cost effective method to reduce the impact of congestion on the air traffic system”.

Several different sources of travel time variability have been identified and studied in passenger vehicle traffic. Noland and Polak [56] proposed that dispersion in travel times can be observed from day-to-day, hour-to-hour and even from vehicle-to-vehicle. They focus on traffic incidents (or random delays), pointing out that such events cause a sudden reduction in effective road capacity and in turn an increase in travel time variability. Hanson and Huff [57] took a different approach by attempting to determine the extent to which there is a systematic, or non-random, component to travel time variability. They identified driver behaviour as a major source of variability in travel times. They showed that a driver’s daily patterns (i.e., the assemblage of behaviours that occur over a given day) exhibit a great deal of variety, making it difficult to show a non-random component in travel time variability.

Variability was also studied on a smaller scale, over the course of a single day. Many publications focus on the effects of signalized intersections at different points of the day, in an effort to better synchronize traffic lights. For example, Fu and Helinga [58] developed an analytical model for estimating the variance of delay at signal-controlled intersections for different simulated traffic conditions, representing different times of day (i.e., rush hour). Their research clearly identifies intersections as a significant source of variability in travel times, especially during periods of high traffic congestion. Similar conclusions were drawn by Olszewski [59] while attempting to model the probability distribution of delay at signalized intersections. Another source of variability in travel times over the course of a single day are
vehicle-to-vehicle differences. In [56] and [60] this is attributed to variations in driver behaviour, such as aggressiveness and or lane choice decisions.

Another source of variability identified in travel times is route choice. De Palma and Picard [61] studied route choice for a number of driver’s when travel time was uncertain, and found that certain personality types were more risk averse than others when it came to route selection. They also found that virtually all drivers had the inclination to select the route with the shortest perceived travel time, but that their perception was rarely correct in a complex road network. A 2010 study by Tian et al. [62] focused on the effects of this skewed perception of route with shortest travel time by comparing route selections of two groups—one equipped with Advanced Traveler Information Systems (ATIS), an on-board feedback system designed to aid in route selection, and one without. The findings showed that the drivers using ATIS had overall shorter mean travel times, showing that route selection is indeed a source of variability, and is often detrimental to overall travel times due to human error.

At present, there is a lack of published literature dealing with mine-specific sources of travel time variability. The only acknowledgement of variability in mine truck travel is found in a series of recent papers by Worrall and Nebot. They propose an alternative method for haul truck localization [63] that does not require a comprehensive fleet management system nor full wireless network coverage. Instead, their system is based on a mesh topology network to facilitate vehicle to vehicle, and vehicle to base communication. Historical GPS data are used to construct route maps and populate a velocity probability density function for each road section (see Figure 4-1). Their algorithm inputs real-time positioning data and uses the resulting probability density functions (PDF’s) to make a prediction of a truck’s position. The predicted locations are relayed back to the base station at the end of every trip, eliminating the need for constant coverage typical
of commercial fleet management systems. In this way, they are respecting the variability present in truck travel and using it to predict truck positions along with corresponding confidence limits. This method was implemented successfully in follow up work by the same authors in [64], [65] and [66].

Figure 4-1- PDF generated for a new method of truck localization [65]

4.2 Nature of a Haul Time Variable

The time it takes for a haul truck to travel from one location to another in a large open pit mine is dependent on many parameters. It is based on a truck’s route and velocity, which are both dependent on environmental, physical, psychological and technological parameters. Some of the most important variables influencing haul times are listed in Table 4-1.

Table 4-1- Parameters involved in haul time calculation

<table>
<thead>
<tr>
<th>Environmental parameters:</th>
<th>Physical parameters:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Distance &amp; topography</td>
<td>• Truck state (empty/loaded)</td>
</tr>
<tr>
<td>• Road conditions</td>
<td>• Fill factor</td>
</tr>
<tr>
<td>• Weather conditions</td>
<td>• Truck configuration (mechanical)</td>
</tr>
<tr>
<td>• Traffic volume</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Measurement parameters:</th>
<th>Behavioural parameters:</th>
</tr>
</thead>
<tbody>
<tr>
<td>• GPS readings</td>
<td>• Driver decisions</td>
</tr>
<tr>
<td>• Boundary definition</td>
<td>• Right-of-way guidelines</td>
</tr>
</tbody>
</table>
Haul times can take on any value over a given time interval, or sample space. In other words, the set of recorded haul times $s$ falls within a sample space $S$. A real-valued function that is defined on the space $S$ is called a random variable. In other words, in a particular experiment, a random variable $X$ would be some function that assigns a real number $X(s)$ to each possible outcome $s \in S$ [67].

That is not to say that every value in the interval can or will be found in sample data, even as $|s| \to \infty$. For example, for a set of individual haul times $s \in S = [200, 500]$, one may never observe a haul time of exactly 231.34 seconds. However, neither this specific time, nor any single value within this range can be ruled out as a possible observation. Since random variables of this type have a continuum of possible values, they are called continuous random variables. The function $f(x)$ which models the relative frequency behaviour of $X$, is called the probability density function. A formal definition is provided by Scheaffer et al. [68].

These statistical definitions are useful tools for investigating the different sources of variability in haul times. By definition, variability is present in all experiments involving random variables. The question is, can we identify sources of variability? The data were analyzed according to the most likely contributing factors identified in the literature review. These are listed below, and described in a mining context:

- Intersections: where trucks have a high probability of slowing or stopping;
- Varying road grades: inclines, declines and flat sections;
- Different routes between identical nodes;
- Time spent at nodes: variability in total loading, dumping and queuing times;
- Different truck states: loaded or empty.
In each case, identical polygons were used to identify origins and destinations. This ensures identical arrival/departure boundaries, and uniformity of haul distances between origin and destinations.

4.3 Intersections

The literature review identified intersections as areas of interest in terms of travel time variability for passenger vehicle traffic. Given that intersections were present in the Goldstrike haul road network, it was hypothesized that intersections would also be a source of variability in mine haul traffic. The most congested intersection was analyzed: it is labeled as Q3INT, shown on Figure 4-2.

Figure 4-2- Goldstrike pit with intersection Q3INT of interest highlighted
4.3.1 Q3INT

Q3INT was located on the eastern side of the pit, at the entrance to the upper most (approximately 1600 m elevation) Eastern in-pit dumps. This intersection was crossed 589 times by the monitored trucks over the over the collection period, making it the busiest road section in the mine. The intersection configuration is shown in Figure 4-3. It accepted four-way traffic, with two stop signs on the Eastern and Western entrances. In that way, the traffic travelling up to the pit exit or down towards the south haul road were given the right of way.

![Figure 4-3](image)

**Figure 4-3- Schematic of the Q3INT intersection at Goldstrike, showing stop signs and traffic flows**

To illustrate the Q3INT intersection as a source of variability, two hauls are compared between identical nodes, but using different routes. They represent one-way trips between the main mining bench in the Southwest to the 5200 level dumps to the East of Q3INT. These routes are labeled as ‘Route A’ and ‘Route B’ on Figure 4-4, along with the origin and destination
polygons. Both routes cross the intersection in question, but from different directions. Route A approaches from the South and is given right of way, whereas Route B approaches the intersection from the West and includes a stop sign.

The haul time data for each route were analyzed to investigate the overall effect of this intersection. First, histograms showing the frequency distributions of all haul times were produced. Then, speed thresholds were used to filter the data based on stop (or significant slow-down) events. These speed thresholds were chosen to be 2 and 4 km/h. This filtering process removed any hauls that stopped or slowed down, based on the previous assumptions. In this way, the filtered data sets represent only those haul times that did not stop or slow down considerably at the intersection. Finally, by comparing the effects of filtering for both two routes, observations and remarks can be made about the intersection.
Figure 4-4- Goldstrike pit with intersections, polygons and connecting Routes A & B highlighted
The main hypotheses to be tested were:

1) The stop sign in Route B would include many stop events at Q3INT, and these events are a source of variability in the data set. Variability would decrease in the data set for Route B if the effects of intersection stopping/slowing down are removed;

2) The unimpeded traffic flow in Route A would be largely unaffected by the slow/stop filtering, since this route had no stop sign.

Figure 4-5 and Figure 4-6 show the histograms for Route A and B, respectively. The bins were set at 60 seconds, with the actual bin ranges shown along the bottom of each Figure. These figures are similar in format and layout: the top histogram shows the frequency distribution for all the data, without any speed filtering applied. This represents all recorded hauls that travelled the unique route between shovels and dumps along each given route. The middle and bottom histograms represent subsets of the total data, with hauls slower than 2 km/h and 4 km/h respectively omitted.
Figure 4-5- Histograms for Route A (10-SWSHOVLS to 03-Q3&DMPS along the South haul road). All data (top), filtered at 2 km/h (middle), filtered at 4 km/h (bottom)
Figure 4-6 - Histograms for Route B (10-SWSHOVLS to 03-Q3&DMPS along the North haul road). All data (top), filtered at 2 km/h (middle), filtered at 4 km/h (bottom)

The top histogram in Figure 4-5 shows the distribution of all one-way haul times between the active mining bench and Upper East dumps. The total haul length for this leg was 3547 metres. There were 84 such hauls ranging in duration from 651 to 944 seconds, with a mean of 769 seconds and a standard deviation of 53.5 seconds.

The other histograms in Figure 4-5 represent the same data set but filtered by speed as previously discussed. At first glance, they are both similar in shape and size to the first histogram. This is because this data set had very few low speed events and no stop events. The middle histogram shows that only three hauls went slower than 2 km/h, while the bottom histogram
shows that only seven went below 4 km/hr. The filtered data remained largely unchanged in terms of variability. The overall change in mean and standard deviation was less than 1 second for both filtered data sets.

At first glance, Figure 4-6 seems much different. The data set represented here consists of one-way hauls between the same two locations, but using Route B (see Figure 4-4). This route measured 2749 m, 798 m shorter than Route A. The top histogram shows all haul times collected for this scenario, with a total of 68 trips. These range in duration from 556 to 767 seconds with a standard deviation of 42.39 seconds. A grouping of four significantly longer hauls is also observed at the upper tail of the frequency distribution.

This data set included many hauls with stop events. 45 of the 68 total hauls included stop events (i.e., speeds lower than 2 km/h), 66 % of the total sample size; 61 hauls were filtered at the 4 km/h level, or 90 % of the sample size. It is observed that these hauls are filtered from the upper tail of the frequency distribution, showing that hauls with slow and/or stop events along their route tend to take longer than average. This is reflected in the successive decreases in mean haul times with increasing speed threshold as shown in Figure 4-7. The mean haul times for Route A remains relatively constant with increasing speed threshold, whereas the mean haul times for Route B decrease steadily. It was also observed that the overall variability of the data decreased with increasing speed thresholds as shown in Figure 4-8. The standard deviation of the Route A data remains constant at approximately 53 seconds, regardless of speed threshold, while the standard deviation of the Route B data progressively decreases.
Figure 4-7: Mean haul times of data vs. speed threshold applied

Figure 4-8: Standard deviation of data vs. speed threshold applied
Revisiting the hypotheses pertaining to intersections, the following conclusions are drawn based on the analysis:

1) The stop sign in Route B would include many stop events at Q3INT, and these events are a source of variability in the data set. Variability would decrease in the data set for Route B if the effects of intersection stopping/slowing down are removed.

For this analysis, stop events were defined as any reduction of speed below a threshold of 2 km/h. According to this definition, many stop events were recorded at the Q3INT (45 of 68 haul times, or 66% of the total data set). There were also many significant slow-down events at Q3INT (< 4 km/h). By applying incrementally increasing speed thresholds, the effects of these stop/slow-down events were quantified. The mean haul time decreased (see Figure 4-7 - Route B in red) with increased speed threshold, as well as the standard deviation (see Figure 4-8 - Route B in red) of the data set. This analysis supports this hypothesis, showing that the removal of intersection interference decreases the overall variability in the haul time data.

2) The unimpeded traffic flow in Route A would be largely unaffected by the slow/stop filtering, since this route had no stop sign.

The removal of intersection effects had virtually no impact on haul times along Route A. The mean haul times remained constant at approximately 768 seconds (see Figure 4-7 - Route A in blue) and the standard deviation behaved similarly, constant at approximately 53.5 seconds (see Figure 4-8 - Route A in blue). Again, the stated hypothesis is supported by the results of this analysis.

4.4 Road Topography

Another suspected source of variability in haul times was road topography. The road network at the Goldstrike Pit is made up mostly of ramps, which can significantly influence truck
travel speeds. For example, the operator of a haul truck travelling on a flat or decline section of road is free to travel at any speed, bounded only by the mechanical limits of the engine and/or the posted speed limits. However, a truck travelling on an incline is under higher engine load, and will therefore have a lower maximum possible speed. This observation led to the following hypotheses:

1) Trucks travelling along haul legs that are mostly flat and/or declined (empty) will show higher variability in speeds than others that are travelling over mostly inclined road sections (loaded);

2) This increased variability in speeds along flat and/or decline haul routes will increase the overall variability of haul times along those routes.

To test these theories, instantaneous speeds for traffic travelling in both directions along Route A (see Figure 4-4) were plotted against cumulative distance along the road. These plots are shown in Figure 4-9 and Figure 4-10 for the shovel to dump and dump to shovel hauls, respectively. The top plot shows the raw data, while the bottom plot shows the speed data with a median filter (window size of 5 data points) applied to remove any single-point spikes related to calculation error.
Figure 4-9: Instantaneous speed vs. cumulative distance travelled along Route A, from shovel (polygon #10) to dump (polygon #3). Raw data (top) and median filtered using a 5 data point window.
Figure 4-10- Instantaneous speed vs. cumulative distance travelled along Route A, from dump (polygon #3) to shovel (polygon #10). Raw data (top) and median filtered using a 5 data point window.
The different levels of speed variability at different locations along the route are apparent in these graphics. However it is difficult to visualize and/or quantify from a statistical point of view. The next step in the analysis was to present the data in a more manageable format; an analysis similar to that seen in [63] and [64] was desired (recall Figure 4-1). To this end, the road was sub-divided into bins equal to three truck lengths (approximately 45 m), and the data are presented in box and whisker plots for each of these 45 meter sections. The road elevation profile is superimposed in red to investigate the effects of road grade on speed variability. Figure 4-11 shows the output for the shovel to dump haul leg and Figure 4-12 for the return trip, dump to shovel. The box (or bold section) for each 45 meter road section represents the interquartile range of speeds, while the whiskers extend out to cover 99.5% of the data (+/- 3σ). For code detail see Appendix C-8.
Figure 4-11- Speed distributions for loaded one-way haul traffic between shovel (polygon #10) and dump (polygon #3) along Route A using a bin width of 45 metres. Haul road elevation profile superimposed in red
Figure 4-12: Speed distributions for empty one-way haul traffic between dump (polygon #3) and shovel (polygon #10) along Route A using a bin width of 45 metres. Haul road elevation profile superimposed in red.
Figure 4-11 shows a haul route for a loaded truck that starts out relatively flat, transitioning into a steady incline at approximately 1000 metres. Along the flat section, speeds from 20-50 km/h hour are observed, with an average of approximately 33 km/h. Over the incline, speeds range from 6-25 km/h, with an average of 11 km/h. Table 4-2 shows a significant contrast in terms of variability between the flat section and inclined section. Using either the 3σ or IQR spans as indicators (see Table 4-2, in bold) it is observed that speeds over the flat section are roughly three times more scattered than those over the inclined section.

Table 4-2- Statistical analysis of data presented in Figure 4-11

<table>
<thead>
<tr>
<th>Statistics</th>
<th>Flat section, 0-1000 m</th>
<th>Incline section, 1000-3100</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Span (km/h)</td>
<td>Range (km/h)</td>
</tr>
<tr>
<td>3σ (99.5 %)</td>
<td>31.83</td>
<td>16.24 - 48.07</td>
</tr>
<tr>
<td>IQR (Middle 50 %)</td>
<td>7.98</td>
<td>28.18 - 36.16</td>
</tr>
<tr>
<td>Median</td>
<td>32.70</td>
<td></td>
</tr>
</tbody>
</table>

Figure 4-12 represents the empty, return trip along Route A—this is evident when comparing the elevation profiles (shown in red) on both figures. The scatter of these data is comparable to that seen in the flat section of the previous analysis. Internally, the variability observed here is more significant in the flat section than the decline section.

Table 4-3- Statistical analysis of data presented in Figure 4-12

<table>
<thead>
<tr>
<th>Statistics</th>
<th>Decline section, 250-2500 m</th>
<th>Flat section, 2500-3000</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Span (km/h)</td>
<td>Range (km/h)</td>
</tr>
<tr>
<td>3σ (99.5 %)</td>
<td>28.49</td>
<td>16.72-45.21</td>
</tr>
<tr>
<td>IQR (Middle 50 %)</td>
<td>7.48</td>
<td>27.03-34.51</td>
</tr>
<tr>
<td>Median</td>
<td>31.37</td>
<td></td>
</tr>
</tbody>
</table>
Variability is observed in truck speeds along the haul routes, especially in flat or declined road sections. Does this translate into higher variability in haul times? This is difficult to prove with the preceding analysis, since the return route includes a stop at an intersection (Q3INT - see Figure 4-3) while the loaded trip does not. In order to link speed variability to haul time variability, it was necessary to analyze a route that was fundamentally the same for traffic travelling in both directions. Consider Route C shown in Figure 4-13. This route connects the main mining bench with the 5010 in-pit dump in the South. This route is similar to Route A, but does not cross any intersections with stop signs. Therefore, any significant slowing or stopping along the route can be considered as situational—meeting another truck when exiting the dump and stopping, slowing down considerably when navigating around the switchback, etc. The configuration of this route, along with these simple assumptions, makes it suitable for the desired comparison.
Figure 4-13- Goldstrike pit with polygons and connecting Route C highlighted
Figure 4-14 and Figure 4-15 show the variability in speeds for both directions along Route C, using the identical box and whisker plots seen previously. Key statistics related to the speed vs. distance plots are presented in Table 4-4 and Table 4-5, and finally the haul time histograms in Figure 4-16 and Figure 4-17. Finally, a summary of the results are presented in Table 4-6.
Figure 4-14 - Speed distributions for empty one-way haul traffic between dump (polygon #10) and shovel (polygon #5) along Route C using a bin width of 15 metres. Haul road elevation profile superimposed in red
Figure 4-15- Speed distributions for loaded one-way haul traffic between dump (polygon #5) and shovel (polygon #10) along Route C using a bin width of 15 metres. Haul road elevation profile superimposed in red.
Table 4-4- Statistical analysis of data presented in Figure 4-14, Route C (10to05)

<table>
<thead>
<tr>
<th>Statistics</th>
<th>Flat section, 0-1000 m</th>
<th>Incline section, 1000-2700 m</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Span (km/h)</td>
<td>Range (km/h)</td>
</tr>
<tr>
<td>3σ (99.5 %)</td>
<td>24.21</td>
<td>24.06-48.27</td>
</tr>
<tr>
<td>IQR (Middle 50 %)</td>
<td>8.11</td>
<td>28.78-36.89</td>
</tr>
<tr>
<td>Median</td>
<td>31.85</td>
<td></td>
</tr>
</tbody>
</table>

Table 4-5- Statistical analysis of data presented in Figure 4-15, Route C (05to10)

<table>
<thead>
<tr>
<th>Statistics</th>
<th>Decline section, 0-1700 m</th>
<th>Flat Section, 1700-2700 m</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Span (km/h)</td>
<td>Range (km/h)</td>
</tr>
<tr>
<td>3σ (99.5 %)</td>
<td>32.52</td>
<td>13.77-46.29</td>
</tr>
<tr>
<td>IQR (Middle 50 %)</td>
<td>8.31</td>
<td>25.86-34.17</td>
</tr>
<tr>
<td>Median</td>
<td>30.71</td>
<td></td>
</tr>
</tbody>
</table>

Figure 4-16- Histogram for Route C (10-SWSHOVLS to 05-5010WD01 along the South haul road). All data.
Figure 4-17- Histogram for Route C (05-5010WD01 to 10-SWSHOVLS along the South haul road). All data.

Table 4-6- Comparison of haul time distributions for both directions of travel along Route C

<table>
<thead>
<tr>
<th>Statistics</th>
<th>10-SWSHOVLS to 05-5010WD01</th>
<th>05-5010WD01 to 10-SWSHOVLS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elevation profile</td>
<td>flat 1000 m; incline 1700 m</td>
<td>decline 1700 m; flat 1000 m</td>
</tr>
<tr>
<td>Loaded/empty</td>
<td>Loaded</td>
<td>Empty</td>
</tr>
<tr>
<td>Haul distance (m)</td>
<td>2723</td>
<td>2748</td>
</tr>
<tr>
<td>Speed variability</td>
<td>low(^3)</td>
<td>high(^4)</td>
</tr>
<tr>
<td>Count (n)</td>
<td>46</td>
<td>34</td>
</tr>
<tr>
<td>Bin size (secs)</td>
<td>60</td>
<td>60</td>
</tr>
<tr>
<td>Mean, μ (secs)</td>
<td>603.20</td>
<td>307.76</td>
</tr>
<tr>
<td>IQR, Q3-Q1 (secs)</td>
<td>47.00</td>
<td>64.00</td>
</tr>
<tr>
<td>Std. dev., σ (secs)</td>
<td>35.96</td>
<td>56.25</td>
</tr>
<tr>
<td>Relative std. dev., σ/μ (%)</td>
<td>5.96 %</td>
<td>18.27 %</td>
</tr>
</tbody>
</table>

\(^3\) See Table 4-4.
\(^4\) See Table 4-5.
The one-way haul along Route C between shovel and dump, represented in Figure 4-14, starts off flat for approximately 1000 metres and then inclines steadily for the remainder of the route. Once again, the contrast between speed distributions over the two sections is apparent. Statistically, the numbers shown in Table 4-4 support this observation, with significantly higher spans for both 3σ and IQR in the flat section.

The return trip, represented in Figure 4-15, is declined for 1700 m before leveling off for the remainder of the trip. The level of variability in speeds seems relatively high and constant over the entire trip. This is indicated in Table 4-5 by similar spans for 3σ and IQR over both sections.

The frequency distributions of haul times seen in Figure 4-16 and Figure 4-17 are also dissimilar with respect to variability. Table 4-6 compares the key statistics of both data sets. The relative standard deviation (RSD, simply σ/μ) of the incline/flat haul is 5.96 %, while that of the decline/flat haul is 18.27 %. This is a threefold increase in variability between these routes.

Revisiting the proposed hypotheses from earlier in this section, the following conclusions are drawn:

1) Haul legs over mostly flat and/or decline road sections will show higher variability in speeds than others that are mostly over uphill road sections.

The data clearly support this hypothesis, as presented in the various speed vs. distance plots in this section. Flat and declined sections of haul road showed the highest level of speed variability. It is suggested that this is largely due to reduced engine constraints over these sections of road. Furthermore, given the mine plan at Goldstrike over the data collection period, all trucks were empty when travelling over the decline portion of the route in question. This configuration allows for the operators to travel as slow or as fast as possible, restricted only by the truck’s
maximum speed and/or by the speed limits imposed by the mine operators. The slight increase in variability along the flat sections is most likely attributed to a heightened level of caution when travelling downhill, and the use of a lower gear.

2) This increased variability in speeds along flat and/or decline haul routes will increase the overall variability of haul times along those routes.

Table 4-6 summarizes both haul time frequency distributions, showing that the variability is significantly greater in the return haul leg with the declined/flat elevation profile. We see an increase of standard deviation of approximately 20 seconds, and a threefold increase in relative standard deviation. Similarly, the interquartile range (IQR) also increases by 17 seconds. These observations support the stated hypothesis.

4.5 Alternate Routes

In a large open pit such as Goldstrike, there is occasionally more than one feasible route between two nodes that are available for travel. Inevitably, these routes are not identical and can have significantly different configurations. This situation is exemplified by Figure 4-4 in section 4.3.1. Routes A and B connect identical nodes: the main mining bench polygon (#10) and the Eastern dumps polygon (#3). However, Route A uses the south haul road and switchback while Route B uses the North haul road. This introduces variability into the system, since DISPATCH® does not take routes into consideration when calculating back-averaged haul times. The result is an average haul time that reflects travel times over two different routes that are independent of each other. This is especially significant when looking at the usage statistics of each route. Route A was travelled 87 times with a mean haul time of 770 seconds, while Route B was travelled 69 times with a mean haul time of 625 seconds. These were observed to happen on similar days,
meaning that there was a high probability of these significantly different haul times being averaged, resulting in a loss of accuracy in haul time estimation.

Although this observation was deemed significant, it is related to the functioning of the DISPTACH® system being used on-site, and therefore outside of the scope of this research (see Section 1.3). The focus is to provide different information for the dispatch algorithms and to explore the results, rather than change the algorithms themselves. However, lessons learned from this observation will be acknowledged in the following sections and chapters, in that it might be possible to convey the haul time information in a different way to make up for this perceived error.

4.6 Node Residence Times

As outlined in the scope statement earlier in this document, this research focuses on variability in travel times along edges (i.e., haul roads) and between nodes (i.e., shovels and dumps). In fact, an early step in the data analysis was to parse the data set by isolating the various nodes in the open pit network with polygons, allowing for an independent analysis of travel times. However, it is hard to ignore the significant levels of variability that were observed within the node polygons. Figure 4-18 and Figure 4-19 show two examples of “node residence time” distributions, or the time elapsed for each truck between consecutive hauls, for the polygon encapsulating the main mining bench and the 4630 in-pit waste dump, respectively.
Figure 4-18- Histogram of node residence times at the shovels on the main mining bench (polygon #10- SWSHOVLS)

Figure 4-19- Histogram of node residence times at the 4630 in-pit dump (polygon #8-4630WD01)
The data from the main mining bench (Figure 4-18) show the highest level of variability, with a RSD of 58%, while the data from the in-pit dump are less scattered (RSD = 26%). These values are significantly higher than those calculated for travel times, and are therefore noteworthy in this discussion. These “residence times” include several components that contribute to variability: travel within the polygon to a constantly changing shovel/dump position, queue time (shovels), and load/dump times. They can also include any unforeseen delays that might have occurred over the data collection period, such as traffic congestion, shovel movement delays, blast crew interference, etc. Separate studies have focused on components of this problem, such as how to reduce queuing times or spotting times. As stated, this falls outside the scope of this research. However, these data were collected and recognized as potentially relevant for simulation purposes and/or future research in this field.

4.7 Inconsistent Arrival Notifications

In this context, some other relevant observations were made during the field trip. The first was an inconsistent arrival notification for trucks. GPS boundaries (often referred to as fences) are sometimes employed to automatically notify the dispatch system of arrivals at certain frequently used areas (i.e., shovels and dumps). These are created by dispatchers and must be kept up-to-date for changes in position; however they are not used at every origin/destination. For those that are not covered by GPS fences, truck operators must manually declare their arrival at their assigned destination. The problem is that there was no standard procedure in place for truck drivers to follow when arriving at a destination. Certain operators would notify dispatch while turning into the dump, some when they are in queue, etc. This results in variable arrival boundaries and in turn variability in calculated travel times. The data analysis herein uses
consistent polygon boundaries for each and every node in the network, an idealized (yet arguably attainable) scenario.

### 4.8 Summary of Results

Several different sources of variability were identified in Section 4.2 and investigated. The first three were intersections, road grades, and truck state (loaded/empty). These areas of interest share a common theme: they all reflect differences in instantaneous speed at different points along the same route. In these cases, such differences were shown to be a cause of haul time variability. For example, the presence of an intersection can be seen as an area of uncertain speed, where some trucks slow down to a dead stop while others pass through largely unimpeded; an incline lowers the maximum speed of a truck significantly; and the range of speed of a loaded truck is restricted as compared to an empty one (especially on inclines). These factors cause inherent variability in subsequent haul times on identical routes, depending on physical features that are unique to each haul.

The final two sources that were discussed, alternate routes and node residence times, are external to the previously discussed system. The use of one or more alternate routes between identical nodes will increase levels of haul time variability only if they are not distinguished by the dispatch system, since each route has unique physical characteristics that influence travel. Referring back to Section 1.3 (Scope of Work), this relates to the logic used in the proprietary dispatch system and is considered out of scope; however it is acknowledged and discussed briefly in the following section.

In the context of this research, node residence times contribute to overall variability in cycle times, not individual haul times. This distinction was a motivating factor behind the initial data analysis set-up; high levels of variability in and around shovels and dumps were observed.
and isolated in order to focus primarily on travel times. Again, this source of variability was deemed to be out of scope; it is acknowledged but not featured.

Thus far, variability in haul times has been measured and its major sources identified. The levels of variability are non-trivial—in some cases, they were shown to reach RSD levels of 20%. One could argue that ignoring variability might have a significant impact on a linearly optimized system. The following section explores how haul time variability is currently handled by the dispatch system, and attempts to show its effects through simulations using manipulated data. Intersections, road grades and truck state (loaded/empty) are the focus of the simulations due to their direct influence on intra-haul variability.
Chapter 5

Effects of Haul Time Variability

As discussed previously, it is suspected that haul time variability is not completely ignored when estimating expected haul times in the dispatch system. This section is meant to explore the possible effects of variability in haul times on dispatch optimization. Three haulage parameters were identified in the previous chapter as areas of interest: intersections, road grades, and truck state (empty/loaded). A subset of the total data set is used for this analysis: the five most common hauls were chosen due to their high frequency of observation and their suitability in terms of the areas of interest. Some are familiar, such as leg 03to10 and 10to04ALT, which were the focus of previous discussions. Table 5-1 shows the considered hauls, listed in order of decreasing number of hauls observed:

Table 5-1- 6 most common hauls observed at Goldstrike in order of decreasing frequency

<table>
<thead>
<tr>
<th>Leg ID</th>
<th>Count (# hauls)</th>
<th>Origin</th>
<th>Destination</th>
<th>Length (m)</th>
<th>Average grade (%)</th>
<th>Intersection stop?</th>
<th>Truck state</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>03to10</td>
<td>106</td>
<td>5250 in-pit dump SW shovels</td>
<td>3517</td>
<td>-4.7</td>
<td>No</td>
<td>Empty</td>
<td>South road, decline, Switchback</td>
<td></td>
</tr>
<tr>
<td>10to03</td>
<td>87</td>
<td>SW shovels</td>
<td>5250 in-pit dump</td>
<td>3547</td>
<td>+4.7</td>
<td>No</td>
<td>Loaded</td>
<td>South road, Switchback</td>
</tr>
<tr>
<td>08to10</td>
<td>72</td>
<td>4630 in-pit dump SW shovels</td>
<td>1594</td>
<td>-2.2</td>
<td>No</td>
<td>Empty</td>
<td>South road, mostly flat</td>
<td></td>
</tr>
<tr>
<td>10to04ALT</td>
<td>71</td>
<td>SW shovels</td>
<td>5110 in-pit dump</td>
<td>2988</td>
<td>+4.0</td>
<td>Yes</td>
<td>Loaded</td>
<td>North road, incline</td>
</tr>
<tr>
<td>10to03ALT</td>
<td>69</td>
<td>SW shovels</td>
<td>5250 in-pit dump</td>
<td>2749</td>
<td>+5.3</td>
<td>Yes</td>
<td>Loaded</td>
<td>North road, incline</td>
</tr>
<tr>
<td>10to08</td>
<td>63</td>
<td>SW shovels</td>
<td>4630 in-pit dump</td>
<td>1642</td>
<td>+2.2</td>
<td>No</td>
<td>Loaded</td>
<td>South road, mostly flat</td>
</tr>
</tbody>
</table>
This subset of haul routes contains all of the attributes desired for analysis: intersections, changes in elevation, loaded trucks, empty trucks, etc. Based on their general configurations and corresponding similarities, these routes are sorted into three groups and analyzed accordingly.

The groups are defined Table 5-2:

<table>
<thead>
<tr>
<th>ID</th>
<th>Truck state</th>
<th>General road profile</th>
<th>Intersection</th>
<th>Leg names</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group 1</td>
<td>Empty</td>
<td>Decline/flat</td>
<td>No</td>
<td>03to10, 08to10</td>
</tr>
<tr>
<td>Group 2</td>
<td>Loaded</td>
<td>Incline/flat</td>
<td>No</td>
<td>10to03, 10to08</td>
</tr>
<tr>
<td>Group 3</td>
<td>Loaded</td>
<td>Incline/flat</td>
<td>Yes</td>
<td>10to03ALT, 10to04ALT</td>
</tr>
</tbody>
</table>

For each Group, sample plots are shown for only one route while the results for both are discussed. The approach to this analysis was to first manipulate the levels of variability in the raw data collected at Goldstrike and observe the effects on mean haul time. To investigate the effects of haul time variability for this route, the experiment design from Section 4.4 (Road Topography) is expanded upon. The main difference in approach is that the incremental travel times are of primary focus, not the instantaneous speeds. The Data Analysis Flowchart in Figure 5-1 provides some clarification. Note that the speed analysis remains unchanged from previous chapters, but the times are now calculated concurrently as well. See Appendix C-6 for the code detail.
**Figure 5-1**: Data Analysis Flowchart Part 1- for variability identification & visualization

**Raw Data- ‘haul-by-haul’ basis**

For $h$ haul legs between location $a$ and location $b$ along the same route of distance $d$, and $n$ data points per haul:

- Positioning data and distances travelled @ 1 Hz meters $f(x)$
- Instantaneous speeds @ 1 Hz km/h $f'(x)$

---

**LEGANALYSIS.m**

- Speeds, $v_n$
- Times, $t_n$

---

**VARTUNESPEEDS.m**

- Instantaneous speeds ($v_n$, km/h) vs. cumulative distance travelled along route (m)

---

**VARTUNETIMES.m**

- Cumulative travel times ($t_n$, seconds) vs. cumulative distance travelled along route (m)

---

**Choose road segment bin size $b$**

- Average bin speeds per truck ($\overline{v_n}$, km/h) vs. cumulative distance travelled along route (m)

---

- Incremental bin travel times per truck ($\Delta t_n$, seconds) vs. cumulative distance travelled along route (m)

---

**Remarks:**

- Points per bin $n_B = # ~ haul ~ legs ~ h$
- Each bin contains the mean speed $\overline{v_n}$ per truck for all corresponding $v_n$ in each bin $s$.

---

- Points per bin $n_B = # ~ haul ~ legs ~ h$
- Incremental bin travel times $\Delta t_n$ are interpolated (linear) per truck for each bin $s$. 

The incremental (bin) travel times go on to form the basis of the data modelling step before simulation, as shown in Part 2 of the Data Analysis Flowchart (Figure 5-2). The data are filtered to remove selected amounts of variability and compared to the original data. For every comparison, two sets of simulated haul times are generated based on the incremental bin times. The first uses a summation of randomly sampled bin times from the observed, unaltered data (control). The second is made up of randomly sampled bin times from a data set with a certain amount of variability removed from the data (experiment). A MATLAB® script was written to perform this task, inputting distributions of incremental haul times and outputting a truncated data set based on a user-defined exclusion rule. These exclusion rules were chosen for each Group, in order to analyze specific phenomena. For each experiment, 200 simulated haul times are generated. See Appendix C-9 for the code detail.

The two simulated data sets are then plotted, compared and discussed. Finally, the new haul times were used to simulate a simple dispatching scenario to investigate the possible effects on production and overall optimization.
DATA ANALYSIS FLOWCHART- DATA MODELLING AND SIMULATION

“Remove desired level of variability”
- Flexible- the road can either be split by bins (as shown here) or by user-defined road sections (i.e. intersections)
- Variability defined by quartiles (varying % of data) or by truck behavior (slow trucks, stopping trucks, speeding trucks, etc.)

Observed Incremental bin times
- Points per bin $n_b = \#$ haul legs $h$
- Incremental bin travel times $\Delta t_s$ are interpolated (linear) per truck for each bin $s$.

“Generate simulated haul times”
- For each analysis, two sets of simulated data are generated for comparison— one based off raw data, and one based off modified data (removal of variability)
- Simulated data sets are compared to investigate the effects of variability removal— i.e. does the mean haul time change?

Figure 5-2- Data Analysis Flowchart Part 2- data modelling and simulation
5.1 Group 1 (Empty, Decline/Flat)

5.1.1 Analysis

This Group contains empty hauls from dumps returning to shovels along a predominantly downhill route. The first route chosen for analysis connects the SW shovels and the 5250 in-pit dumps to the East as shown in Figure 5-3. It is referred to as leg ‘03to10’ in this analysis (see Table 5-1). Recall that this route starts just before the intersection Q3INT, through the main switchback and along the main south haul road. Overall it is a downhill route with an average grade of 4.7 %, with the first two-thirds at a 10-12 % incline and the remainder being relatively flat. The total haul distance was 3517 m. All trucks travelling on this route were empty, since it was a dump-to-shovel leg. It was the most common haul leg travelled, with 106 recorded haul times.
Figure 5.3- Group 1 routes and polygons- 03to10 (green) and 08to10 (blue)
The first step in this analysis was to generate the instantaneous speed vs. cumulative distance travelled, as seen previously in Section 4.4 (Road Topography), but this time with the addition of a second plot of cumulative distance travelled. These plots are shown in Figure 5-4 below.

![Graphs showing instantaneous speed vs. cumulative distance and cumulative haul time vs. cumulative distance](image)

**Figure 5-4** Instantaneous speed and cumulative haul time vs. cumulative distance travelled, route 03to10

The speed plot shows a high level of variability, typical of an empty truck travelling unrestricted on a decline and/or flat road. The time plot shows clear variability in cumulative haul times, with a range of approximately 300-550 seconds total haul time at full distance. Box and whisker plots of these data are shown in Figure 5-5 using bin sizes equivalent to three truck...
lengths (45 m). Each box and whisker group shows the variability of data from within that group, and together show a representation of how the dispersion varies along the route. Recall that the ‘boxes’ represent the middle fifty percent of the data distribution (i.e., Q2 and Q3), while the ‘whiskers’ represent the top and bottom 25% of the data (i.e., Q1 and Q4), and the black dots represent median values.

Figure 5-5- Box plot distributions of incremental speeds (top) and times (bottom) over haul leg 03to10. Relative elevation profile is superimposed in red.

With knowledge of route layout, one can discern certain route attributes by analyzing Figure 5-5. The trucks slow down at the very beginning of the leg, when turning onto the main haul road from the dump. The data show that the trucks do not come close to full stops however,
with the lowest average bin speed in this section at approximately 10 km/h and a corresponding maximum bin time of 16 seconds. The truck then travels along the South road decline. It passes a switchback (1700 m – 2300 m) which is less apparent in the speed plot (top) but clearly visible in time plot (bottom). Here we see spikes in bin travel times, some upwards of 50 seconds per 45 metre bin. These spikes occur at 1750 m and 2250 m respectively, corresponding to the locations of hairpin turns. The road flattens out after the switchback and bin travel times return to normal levels of 5-6 seconds per bin for the remainder of the journey.

5.1.2 Results

As outlined in Figure 5-2 (Data Analysis Flowchart Part 2) the next step is to remove certain data points to modify haul time variability and investigate the effects. Three data exclusion rules were chosen for analysis. The first is the removal of all incremental times falling in the 1st and 4th quartiles of each distribution (i.e., the ‘whiskers’ as seen in Figure 5-5), which corresponds to a general reduction in travel time variability: the ‘slowest’ and ‘fastest’ trucks are removed, leaving only the middle fifty percent of the data. The results are shown in Figure 5-6. As expected, the plot of simulated times before variability modification (top) is similar to the plot of actual bin time distributions (Figure 5-5, bottom), since it is based off those unmodified numbers. Using these numbers, the mean haul time for this leg is 394 seconds. The lower plot is generated with the exclusion rule applied, so that Q1 or Q4 incremental haul times are not sampled, resulting in much tighter distributions for each bin. The mean haul time for this modified data set is 374 seconds.
The haul time distributions in each bin are reduced significantly and are the difference is clearly visible in Figure 5-6. A decrease in mean haul time of 19 seconds or 4.8% is observed after removal of data in the 1st and 4th quartiles. This result is reproduced for the second route in Group 1, leg 08to10, where a reduction in mean haul of 14 seconds was observed. Since an identical amount of data was removed from both extremities of each bin distribution, this observation leads to the conclusion that in this case, the slower trucks are more influential on overall haul time than the faster trucks. If the mean haul time was unaffected, the conclusion would be that both tails have equal influence and their effects are negated.
To explore the effects of extreme values in the time distributions, experiments were designed to isolate these outlier points. The first deals with speeding trucks. Any truck travelling in excess of 40 km/h (Goldstrike’s posted speed limit for in-pit trucks) is flagged at the point of infraction. The instantaneous speeds at these points are then set to equal the maximum speed limit, effectively removing the effects of all speeding trucks in the data set. The second focuses on trucks that unexpectedly slowed to a near stop along the route; that is, since there are no intersections along these routes where trucks are required to stop, these events are unplanned and occur randomly. A similar flagging process is followed, and these excessively slow events (less than 5 km/h) are removed by setting their corresponding bin times to the median bin time. Sample results for leg 03to10 are shown in Figure 5-7 and Figure 5-8, respectively.

![Figure 5-7](image-url)  
*Figure 5-7- Simulated bin times vs. Distance along haul route- leg 03to10, Group 1- before (top) and after (bottom) correction for speeding trucks (>40 km/h)*
Speeding events were more common, with trucks observed to have exceeded the speed limit of 40 km/h in 70 of the 78 bins. 32% of the data contained in those bins were recorded at speeds greater than 40 km/h. These data points were set to equal 40 km/h to simulate a scenario where all trucks travel at speeds less than or equal to the posted speed limit. The results show a modest increase in mean haul times, from 394 to 401 seconds. This increase of 6 seconds or 1.7% shows is relatively small, given that approximately one third of the trucks were caught speeding at some point. Therefore, one might reasonably conclude that given their frequency, speeding events have little effect on overall mean haul times.

Stopping events, defined as trucks slowing to 5 km/h or less, were less frequently observed. Only five data points in five of the 78 bins were flagged as stopping events in the entire data set.
for leg 03to10. After replacing these points with the mean bin speed, the overall mean haul time
decreased from by 2 seconds from 393 to 391, or 0.5%. This is a relatively large decrease,
considering that this change is caused only by the effects of five data points. The results from the
other route in Group 1, leg 08to10, were similar. Nine data points were flagged as stop events,
and their removal caused a mean haul time decrease of 11 seconds or 6.3%. Again, only a small
fraction of truck stop events were observed, yet they seem have a significant impact on the mean
total haul time. Although stop events were observed less frequently than speeding events, the
results here show that they have a greater impact on overall mean haul times.

5.1.3 Discussion

The removal of the data falling in the first and fourth quartiles represents a general
reduction in haul time variability, with equal amounts of data excluded from both extremes. The
results of the analysis showed an overall decrease in mean haul times, suggesting that in this case,
slower vehicles have a greater effect on overall mean haul times than faster ones. This result was
further tested in the analysis of speeding vehicles. Approximately one-third of all trucks were
observed to travel in excess of the mine’s posted speed limit of 40 km/h. Of these, 24% were
found to be travelling at instantaneous speeds greater than 44 km/h, or 10% above the maximum
allowable speed. This is a significant observation, since these speed limits are meant to be strictly
enforced for safety purposes. One might be interested in the impacts of speeding. For example,
does speeding translate into a significant reduction in overall haul times? In other words, might
there be a benefit to increasing the speed limit to allow for faster travel times in the open pit? The
results suggest that this is not the case, since the removal of speeding trucks had a negligible
impact on overall haul times. Furthermore, it is important to note that this analysis does not
account for safety, which is obviously a concern at higher speeds. Generally speaking, one might
conclude that the current speed limit of 40 km/h is not a major constraint on productivity, and should be set and enforced with respect to safety.

The analysis of trucks stopping or slowing down considerably along this haul leg reaffirm the observations from the first experiment, showing that these events have a higher impact on overall mean haul times. Based on field observations, it is known that these events are usually caused by traffic interactions: usually, haul trucks yielding right-of-way to other vehicles. A small number of such events can increase the overall mean haul times significantly. This suggests that it is beneficial to ensure proper right-of-way practices within the open pit. Haul trucks, whether loaded or empty, should be consistently given right-of-way by other vehicles as per the mine’s traffic regulations. This will minimize unexpected truck stoppages and in turn, overall haul times.

5.2 Group 2 (Loaded, Incline/Flat)

5.2.1 Analysis

Group 2 consist of haul legs that were travelled by loaded trucks originating at the southwest shovels (polygon #10). The routes were both predominantly inclined along the south route. These haul legs are similar to those of Group 1 in that they are the reverse trip, using the same routes in the opposite directions (see Figure 5-3). As discussed in Section 4.4, the speed and time data for these routes contain less variability due to the combination of haulage parameters: loaded trucks on inclined roads tend to travel at relatively constant speed due to mechanical constraints. The bin distribution plots for leg 10to03 are shown in Figure 5-9 as an example. High levels of variability are observed in the initial flat section of road ending at approximately 1200 m. The rest of the route is inclined, limiting the speeds and resulting in relatively constant travel times. Here, the switchback section spans the road section from 1300 – 1900 m. It is interesting to note that in comparison to the Group 1 results, there is much less variability along this section.
This is likely due to the priority given to these loaded trucks travelling uphill—the Group 1 traffic travelling in the opposite direction yields to them by slowing down when necessary and giving them ample room in the hairpin turns. Overall, it was anticipated that the data presented in Figure 5-9 would be less sensitive to reductions in variability as compared to Group 1. Its predominantly inclined profile contains much less inherent variability in terms of both speeds travelled and incremental travel times.

Figure 5-9- Box plot distributions of incremental speeds (top) and times (bottom) over haul leg 10to03. Relative elevation profile is superimposed in red.
5.2.2 Results

The first experiment applied a general reduction in variability to the entire data set by once again removing all data lying in the 1st and 4th quadrants of each bin distribution. The results (Figure 5-10) show a reduction in mean total haul time with the exclusion of these data, from 766 to 757 seconds. This corresponds to a decrease of 1.2 %. When compared to the results from Group 11, the result is similar but at a lesser magnitude.

![Graph showing simulated bin times vs. distance along haul route for leg 10to03, Group 2, before and after application of variability exclusion rule](image)

**Figure 5-10**- Simulated bin times vs. Distance along haul route- leg 10to03, Group 2- before (top) and after (bottom) application of variability exclusion rule (removal of data in Q1 and Q4)
To further study the hypothesis that variability has a lesser impact on overall mean haul times in inclined/loaded hauls, Group 2 is subjected to the same speeder/stopper analysis as performed for Group 1. Results are shown in Figure 5-11 and Figure 5-12.

**Figure 5-11**- Simulated bin times vs. Distance along haul route- leg 10to03, Group 2- before (top) and after (bottom) correction for speeding trucks (>40 km/h)
These results show very little effect of variability removal for both speeding and stopping analyses. Speeding events were frequent as in Group 1, but in this case constrained to the flat portion of the haul; in fact, speeding was observed in all of the first 27 bins (0-1215 m) at varying frequencies (391 total events, or 19% of all data points in these bins). Their removal had a negligible effect, resulting in an increase in mean total haul time of 2 seconds or 0.26%. Only one stopping event was recorded along this route, yet its removal caused the mean total haul times to drop by 2 seconds. The results of this analysis show that in this case, the effect of one
stopping event has the same influence as 391 speeding events in terms of magnitude—+/- 2 seconds.

5.2.3 Discussion

The haul routes represented in Group 2 differ from those in Group 1 in two fundamental ways: the trucks are loaded, and they are mostly travelling uphill. The bin distribution of incremental haul speeds and times shown in Figure 5-9 display a clear contrast between the initial flat section of the route (0-1200 m) and the inclined portion (1200-3500 m). A contrast in levels of variability is also apparent between these sections. Overall, a lower level of variability in speeds and haul times is present in the Group 2 hauls, mostly due to the relatively constant speeds travelled on the incline. Therefore, as hypothesized in section 5.2, removal of variability both generally (Q1 & Q4) and specifically (speeders & stoppers) had less of an effect than was seen in Group 1. Although the effects were less pronounced, similarities were observed in terms of which events were the most influential on overall haul times. In both cases, stopping events were less common than speeding events but were shown to have a significant effect despite their low frequencies. In fact, the 391 speeding events in the Group 2 analysis had the same magnitude impact as one solitary stopping event, each changing the mean total haul time by 2 seconds. This reinforces the theory proposed in the Group 1 analysis: stopping trucks are shown to have much greater impact on overall haul times than speeding trucks. However, due to the higher consistency of travel speeds and times in loaded/inclined travel, it seems more beneficial to focus on variability reduction in empty and/or declined hauls rather than loaded and/or inclined hauls.

5.3 Group 3 (Crossing Intersections)

The legs in Group 3 are both loaded hauls travelling on predominantly inclined routes with a stop sign along the route at the Q3INT intersection (introduced in section 4.3.1, shown in Figure 103
4-3). Both of these routes approach this intersection from the West before continuing on to the
dumps defined at Polygon 3 to the East or those at Polygon 4 to the South. The focus of this
analysis is the effect of the intersection on the overall haul times. First, the observed behaviour at
the intersection is presented and discussed to gain a better understanding of stopping patterns.
Second, the effects of removing variability in travel times across the intersection are analyzed.
Finally, a theoretical case is presented to investigate the benefits of removing the intersection
effect entirely.

5.3.1 Stopping Behaviour at Intersections

Before proceeding with variability manipulation and haul time simulation, a better
understanding of truck behaviour at intersections was desired. Consider the intersection crossings
at Q3INT for both legs in Group 3: 10to04ALT and 10to03ALT, as circled in Figure 5-13 and
zoomed in Figure 5-14. At first glance, there seems to be some local variation in behaviour, with
some trucks coming to complete stops (defined by a truck observed to travel at less than 1 km/h
in the intersection area) while others are observed to slow down and roll through. Just how many
trucks actually came to a full stop as per the mine’s traffic regulations? For those that didn’t, how
fast did they travel through the intersection? Did they slow down at least? The data were analyzed
in an attempt to answer these questions, with results shown in Figure 5-15.
Figure 5-13- Speed vs. Cumulative Distance Travelled along leg 10to03ALT, with intersection circled in black

Figure 5-14- Speed vs. Cumulative Distance Travelled along leg 10to03ALT, zoomed in on the intersection area (~2450-2600 m)
The data show that 41% of the trucks followed the correct procedure of coming to a full stop before crossing the intersection; 74% slowed considerably to a speed lower than 3 km/h; and 88% slowed to less than 5 km/h. These results indicate that the mine’s posted stop sign at this intersection is not consistently respected, since the majority of trucks (80 of 136 or 59%) failed to stop completely at this location. These delinquent trucks do slow down considerably, however, with roughly 75% recorded at less than 3 km/h within the intersection and 90% below 5 km/h. On the other extreme, eight trucks were observed to have stopped at the intersection for an extended period of time (defined here as greater than 20 seconds). This represents 6% of the Group 3 data. Of these, five trucks sat for longer than 40 seconds. Although these events do not contradict the mine’s traffic rules, they do represent unplanned stops along a haul route that occur unpredictably. Ideally, these trucks would come to a complete stop at the stop sign, and immediately continue through unimpeded, while cross traffic yields. In cases of prolonged delays.
at the stop sign, it is clear that there is either a vehicle travelling through the intersection, or a string of multiple vehicles being waved through by the truck operator. This scenario contradicts the right-of-way guidelines of the mine, which reserve the right-of-way for loaded trucks before any other (with the exception of emergency vehicles). Therefore, this scenario is avoidable. Although these delays do not occur frequently, it is possible they have an impact on overall haul times, as discussed in the following chapter.

Another important observation is that these routes were predominantly used during night shift during the data collection period: 116 of the 136 total hauls in Group 3 crossed this intersection during night shift. Of these, 59% were found to have not come to a complete stop at the intersection. In contrast, not a single truck failed to stop at this same location during the day. This contrast in behaviour is noteworthy—it sheds light (pun intended) on the perceived mentality shift of truck operators during night shift. For instance, it is often presumed that this behavioural change arises due to the lower level of supervision during the night hours. The numbers here would support such a claim. But is this the only cause? Perhaps the powerful headlights used by trucks travelling at night make for better intersection visibility, and in turn better traffic awareness resulting in lower stopping frequency? If this is the case, perhaps there is a benefit to enforcing stop signs only during the day and not at night? These are all valid and potentially valuable questions that could initiate future research in this field.

5.3.2 Variability Reduction

It was desired to investigate the effects of a general reduction in the extreme values associated with intersections for both short and long intersection delays. The data lying in the first and fourth quadrants of every bin within the intersection area was removed. By only applying this filter to the data within the intersection, we are able to investigate its effect independently of other
factors described earlier in this chapter (i.e., road grades, loaded vs. empty, etc.) The before-and-after result of removing said data is illustrated in Figure 5-16.

![Box-and-whisker plots](image)

**Figure 5-16** - Before (left) and after (right) box-and-whisker plots of Q3INT intersection crossing times. Span: 2450-2600 m.

The box-and-whisker plot on the left shows the actual time distributions recorded across the 150 m section centered at the intersection: 68 observations ranging from 29-79 seconds with a mean crossing time of 36 seconds. On the right, we see the same data set with the first and fourth quartiles removed so that we are left with the middle fifty percent of the data. Note that the scales are dissimilar (for improved visual effect). Here, the data ranges from 33-37 seconds, with a mean
of 34 seconds. Haul times are simulated as before, using actual, unmodified data everywhere except at the intersection in question. The results are shown in Figure 5-17.

![Figure 5-17](image)

**Figure 5-17- Simulated bin times vs. Distance along haul route- leg 10to04ALT, Group 3-before (top) and after (bottom) application of variability exclusion rule. Removal of data in Q1 & Q4 for section 3- intersection (blue) only**

Recall that in this case the data in Q4 represents the slowest trucks to pass through the intersection (top 25% crossing times) while the data in Q1 represents the fastest trucks to pass through the intersection (bottom 25% crossing times). It is observed that the removal of this data leaves a data subset that represents an idealized scenario in the vicinity of the intersection. Firstly, all trucks that stopped for extended durations are excluded. Essentially, this removes the effects of cross traffic interference at this intersection. Secondly, trucks that failed to come to a complete
stop at the intersection are omitted. Therefore, the remaining hauls are fully compliant of the posted stop sign, coming to a full stop before continuing through the intersection unimpeded. The removal of extreme data at the intersection causes the incremental times across the intersection (circled in Figure 5-17) to decrease from 39 seconds to 35 seconds. Overall, the comparison of simulated haul times does not reveal a significant change in total haul times.

5.3.3 Effects of Intersection Removal

The final step in the analysis is to investigate the effects on the theoretical case of removing the intersection altogether. This analysis aims to show the potential benefits of coordinating traffic in such a way that the intersection has no impact on haul times and in turn, productivity. The discussion here uses route 10to04ALT for graphical representations and references supporting results from the 10to03ALT route.

The box and whisker plots for both the instantaneous speeds and incremental bin times are shown below in Figure 5-18. Here, it is noted that speeds gradually reduce and increase on either side of the intersection. The total overall effect of the intersection is observed to span 630 m section of haul road, from 2280-2910 m (refer to the x-axes in Figure 5-18). Therefore, this area becomes the focus for analysis.
Figure 5-18- Box plot distributions of incremental speeds (top) and times (bottom) over haul leg 10to04ALT. Relative elevation profile is superimposed in red.

In order to effectively remove the intersection effect from this data set, the incremental travel times from the bins within the area of influence (2280 – 2910 m) are overwritten with observed values from adjacent bins with similar elevation profiles. This process is informally comparable to ‘cutting and pasting over the intersection’ with real time data sourced from adjacent time bins, before and after the intersection area. The resulting incremental travel times for this modified data set are shown in Figure 5-19. Note that variability is included at observed levels in the place of the intersection to maintain a realistic representation.
Haul times were simulated using this hybrid data set using the same procedure as seen in earlier analyses from this chapter (outlined in Figure 5-2). The resulting mean haul times are compared to the unmodified times for each route in Group 3 in Table 5-3. Both routes are significantly affected, with decreases in travel times of roughly 6-7%.
Table 5-3- Comparison of simulated mean haul times for Group 3 haul routes before and after intersection "removal"

<table>
<thead>
<tr>
<th>Route</th>
<th>Mean haul time- unmodified data (with intersection) (s)</th>
<th>Mean haul time- hybrid data (intersection “removed”) (s)</th>
<th>Change (s)</th>
<th>% change</th>
</tr>
</thead>
<tbody>
<tr>
<td>10to03ALT</td>
<td>606</td>
<td>572</td>
<td>-34</td>
<td>-5.9 %</td>
</tr>
<tr>
<td>10to04ALT</td>
<td>640</td>
<td>596</td>
<td>-44</td>
<td>-6.9 %</td>
</tr>
</tbody>
</table>

5.3.4 Discussion

As previously discussed, this chapter gives an idea of the overall effect of a typical intersection crossing on haul times. The impact is significant, with an average delay for both one-way cases of 39 seconds (based on the assumptions used during analysis). This change would certainly have some effect on productivity. But, to what extent? The purpose of the following section is to address this question, linking variability in travel times to changes in productivity and in turn, economic impacts.

5.4 Impacts on Productivity and Revenue

5.4.1 Simulator

A study of the potential impacts of haul time variability on productivity and, in turn, revenue was carried out. This was done using simple dispatch simulations, based on a program written by Christopher Cameron [69], a colleague at Queen’s University. The goal of his simulator which was developed in MATLAB® is to recreate realistic operating conditions and track significant operating parameters. The program is initialized using a series of inputs and initial conditions, and outputs optimal flow rates as well as corresponding truck assignments for each path. The simulator is based on discrete events, to enable a realistic model of queuing times and variable path and loading times [69].

113
The simulator comprises two components: a linear optimization module, and a discrete-event simulator. The linear optimization step calculates the minimum fleet requirements by selecting path flow rates that meet specified constraints. It is based on a simplified version of the dispatching linear programming logic (LP) discussed in Section 2.1 as well as examples presented in [11] and [17]. The discrete-event simulator is designed to match trucks to paths and drive the queuing time calculations. Again, it is functionally similar to the logic seen in Section 2.1 (i.e., the DP module), but is more intimately described in Figure 5-20.
Figure 5-20- Flowchart of the discrete event simulator basic structure [69]
The drawbacks of this simulator are that it does not address traffic control, topographical layout, non-uniform equipment fleets, maintenance delays, and truck speeds. Despite these shortcomings, it was deemed useful as a tool to roughly analyze the effects of haul time variability at Goldstrike; at the very least, to give an ‘order of magnitude’ approximation of potential impacts.

5.4.2 Experiment Design

The goal of this exercise was to feed the simulator with numbers obtained throughout the research presented herein to investigate the effects of haul time variability on overall productivity. To this end, it was necessary to provide the model with a representation of the Goldstrike environment. Due to the simplified nature of the program and limited flexibility of its design, it was observed that the most meaningful results were obtained by only inputting the nine most common hauls as described in Table 5-4. This avoids complications due to low frequency hauls that were statistically insignificant (in this case, those with less than 50 recorded haul times) as well as non-productive hauls (those not travelling between shovel and dump or vice versa). For an example of productive haul time data listed in Table 5-4, refer Figure 3-9 (Histogram plot for haul leg 03to10) in Section 3.4.1.
Table 5-4- Most common hauls used as inputs for simulations

<table>
<thead>
<tr>
<th>Rank (#)</th>
<th>EdgeID</th>
<th>Node origin</th>
<th>Node destination</th>
<th>Desc.</th>
<th>Topo profile</th>
<th>Leg count</th>
<th>Cumul. count</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>03to10</td>
<td>03</td>
<td>10</td>
<td>Empty</td>
<td>Decline</td>
<td>106</td>
<td>106</td>
</tr>
<tr>
<td>2</td>
<td>10to03</td>
<td>10</td>
<td>03</td>
<td>Loaded</td>
<td>Incline</td>
<td>87</td>
<td>193</td>
</tr>
<tr>
<td>3</td>
<td>08to10</td>
<td>08</td>
<td>10</td>
<td>Empty</td>
<td>Decline</td>
<td>72</td>
<td>265</td>
</tr>
<tr>
<td>4</td>
<td>10to04ALT</td>
<td>10</td>
<td>04</td>
<td>Loaded</td>
<td>Incline</td>
<td>71</td>
<td>336</td>
</tr>
<tr>
<td>5</td>
<td>10to03ALT</td>
<td>10</td>
<td>03</td>
<td>Loaded</td>
<td>Incline</td>
<td>69</td>
<td>405</td>
</tr>
<tr>
<td>6</td>
<td>10to08</td>
<td>10</td>
<td>08</td>
<td>Loaded</td>
<td>Incline</td>
<td>63</td>
<td>468</td>
</tr>
<tr>
<td>7</td>
<td>01to10ALT</td>
<td>01</td>
<td>10</td>
<td>Empty</td>
<td>Decline</td>
<td>52</td>
<td>520</td>
</tr>
<tr>
<td>8</td>
<td>04to10</td>
<td>04</td>
<td>10</td>
<td>Empty</td>
<td>Decline</td>
<td>51</td>
<td>571</td>
</tr>
<tr>
<td>9</td>
<td>10to04</td>
<td>10</td>
<td>04</td>
<td>Loaded</td>
<td>Incline</td>
<td>51</td>
<td>622</td>
</tr>
</tbody>
</table>

These hauls represent 70\% of all productive hauls recorded during the field trip. This subset consists of 341 loaded haul legs (55\%) and 281 empty haul legs (45\%). In order to implement a simulation of these haul legs, a five-node simplified network of the Goldstrike scenario is required. The network diagram is shown in Figure 5-21. It is important to note that the shovel node (#10) includes two operating shovels, both outputting material at a rate 5000 tonnes per hour. It was necessary to group them into one node due to their close proximity and occasional overlapping that was observed during the data collection period.
Three experiments were designed and conducted according to the parameters presented in Table 5-5, Table 5-6 and Table 5-7, respectively. Experiment 1 only uses the mean haul times as inputs to the simulation, without the presence of any variability. In other words, each path has a unique travel time associated to it; every time a truck is assigned to a given path, it is assigned the same corresponding travel time. This is a theoretical situation, conducted in simulation to establish a theoretical baseline for productivity.

Experiment 2 furthered the analysis by randomly sampling the observed travel times every time a truck-path assignment takes place. This set-up more closely resembled the real-life situation observed at Goldstrike: truck travel times vary according to the observed distributions.
Different levels of travel time variability were inputted and the results were compared to investigate the effects on productivity within the simulation. Ultimately, the goal of Experiment 2 was to quantify the benefits of speed control and/or other dispatch controls aimed at reducing travel time variability.

Finally, experiment 3 was formulated to investigate the effects of intersection removal. It was motivated by the results found in section 5.3.3 (Effects of Intersection Removal). The input data set used in Experiment 2 was manually modified to remove any delays associated with the Q3INT intersection, as if it did not exist. Trucks were also locked to the North road haul legs (10to03ALT and 10to04ALT), both of which cross a stop sign at the Q3INT intersection. This was necessary to narrow the focus of the experiment to intersection effects. Experiment 3 was meant to investigate the potential benefits of intersection removal, whether it is by design modification and/or traffic coordination.

All three experiments were run using a simulation run time of 100 hours. This time was chosen based on trial and error; it was deemed a suitable duration to minimize calculation time without sacrificing a statistically significant result in terms of productivity. Other notable parameters that remained constant in the three experiments were shovel production rates of 5000 tonnes per hour, and a nominal truck payload of 304 tonnes (320 t @ 95 % fill factor). These values were furnished by authorities at Goldstrike, and were deemed acceptable average values for the purpose of this exercise [29].
### Table 5-5: Simulation Experiment 1 - “Theoretical upper limit on productivity”

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>Description</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input:</td>
<td>Mean travel times found from Goldstrike</td>
<td>Means taken directly from data</td>
</tr>
<tr>
<td>Variability:</td>
<td><strong>None</strong></td>
<td>Travel times are constant and do not change during simulation</td>
</tr>
<tr>
<td>LP:</td>
<td>Calculated once at start, based on input travel times</td>
<td>Path flow rates remain constant during simulation</td>
</tr>
</tbody>
</table>

### Table 5-6: Simulation Experiment 2 - "Introduction of variability based on observed levels at Goldstrike"

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>Description</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input:</td>
<td>Simulated haul times from speed distributions</td>
<td><strong>Use different levels of variability</strong>-i.e., &quot;all data&quot; and &quot;middle fifty&quot;</td>
</tr>
<tr>
<td>Variability:</td>
<td><strong>Present</strong></td>
<td>According to observed distributions from Goldstrike data</td>
</tr>
<tr>
<td>LP:</td>
<td>Calculated once at start, based on input travel times</td>
<td>Path flow rates remain constant during simulation</td>
</tr>
</tbody>
</table>
Table 5-7- Simulation experiment 3- "Effects of intersection removal"

<table>
<thead>
<tr>
<th>Characteristics</th>
<th>Description</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input:</td>
<td>Simulated haul times from speed distributions</td>
<td>Use different levels of variability — i.e., “all data” and “middle fifty”</td>
</tr>
<tr>
<td>Variability:</td>
<td>Present; <strong>also, intersection effects removed at Q3INT</strong></td>
<td>According to observed levels from Goldstrike data</td>
</tr>
<tr>
<td>LP:</td>
<td>Calculated once at start, based on input travel times</td>
<td>Path flow rates remain constant during simulation</td>
</tr>
</tbody>
</table>

5.5 Results

5.5.1 Experiment 1

The results from Experiment 1, displayed in Figure 5-22, show a varying instantaneous production rate (blue) over the 100 hours of simulated run time. Values range from 7600 – 8500 tph. Since the inputted haul times for each haul leg are constant—simply the average of the recorded travel times for each leg—the observed variability in instantaneous production rate is caused only by shovel queuing in the discrete-event simulator. The mean production rate for the simulation was 8101 tonnes per hour (green). This production rate refers to the tonnes of material moved and placed at the various dumps. Based on previous results linking travel time variability with an overall increase in mean travel times, it is hypothesized that this production rate will be the highest possible within the simulator, due to the absence of variability altogether. This hypothesis is tested in Experiments 2 and 3, where variability is introduced into the system.
5.5.2 Experiment 2

The results from Experiment 2 are displayed in Figure 5-23. Here, two simulation runs are plotted together for comparison, each using a different level of travel time variability. This was done by having the simulator randomly sample the travel time distributions for each given leg. Both plots (blue and red) represent instantaneous production rates: The plot in blue is based on real, observed levels of travel time variability (i.e., sampling from all recorded haul times for each leg), while the red plot is based on the middle fifty percent of each travel time distribution (i.e., sampling haul times in quartile 2 and 3 only for each leg). The blue plot (high variability) is
observed to fluctuate between 7300 – 9400 tph, spanning 2100 tph, while the red plot (decreased variability) varies between from 7900 – 9100 tph, a span of 1200 tph. This is as expected, simply showing that as production rates become more variable with increased levels of travel time variability.

Interestingly, there is a marked shift in mean production rate for the two simulation runs. The mean production rate for the high variability case is 8086 tph, while the decreased variability case shows an increase in mean production rate at 8287 tph. This productivity increase of 2.5 % is likely due to the positive effects of travel time variability reduction on overall mean haul times (for example see section 5.1). It was also observed that the simulated truck queues were more congested in the high variability case, due to bunching effects caused by an inconsistent (i.e., variable) rate of truck arrivals/departures.
The 2.5 % increase in productivity translates into 37 more ounces of gold produced per day and an overall increase in revenues of $22M per year, as shown in Table 5-8. The material grade used of 0.00835 is an average for all material moved (i.e., ore and waste) was based on a previous study provided by Barrick[70]. A gold price of $1500/oz was assumed. It is important to stress that these potential economic benefits are based on many assumptions, average values and a simplified simulation scenario; therefore they are only meant to give an ‘order of magnitude’ estimate.

Figure 5-23- Experiment 2 results
Table 5-8- Economic analysis of Experiment 2 results

<table>
<thead>
<tr>
<th></th>
<th>&quot;high variability&quot;</th>
<th>&quot;decreased variability&quot;</th>
<th>change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Production rate, hourly</td>
<td>8,086</td>
<td>8,287</td>
<td>201</td>
</tr>
<tr>
<td>Production rate, daily</td>
<td>194,064</td>
<td>198,888</td>
<td>4,824</td>
</tr>
<tr>
<td>Avg. grade of material (oz/t)</td>
<td>0.00835</td>
<td>0.00835</td>
<td>-</td>
</tr>
<tr>
<td>Gold ounces moved per day (oz)</td>
<td>1,620</td>
<td>1,661</td>
<td>40</td>
</tr>
<tr>
<td>Gold price ($)</td>
<td>$1,500</td>
<td>$1,500</td>
<td>-</td>
</tr>
<tr>
<td>Revenue, per day ($)</td>
<td>$2,430,652</td>
<td>$2,491,072</td>
<td>$60,421</td>
</tr>
<tr>
<td>Revenue, per year ($)</td>
<td>$887,187,834</td>
<td>$909,241,353</td>
<td>$22,053,519</td>
</tr>
<tr>
<td>% improvement</td>
<td>-</td>
<td>-</td>
<td>2.50 %</td>
</tr>
</tbody>
</table>

5.5.3 Experiment 3

The results for Experiment 3 are displayed in Figure 5-24. Again, two simulation runs are plotted together for comparison. The only difference between runs is the inputted haul times for both haul legs travelling along the North haul road and crossing the Q3INT intersection from West to East (see section 4.3.1, Figure 4-3 for intersection layout). The unmodified run (shown in blue on Figure 5-24) uses observed travel times, while the modified run (red) uses data with the intersection effects removed. This modified data set was presented in section 5.3.3. The productivity results for both data sets range from 7600-9200 tph (spanning 1500 tph). Although the ranges are identical, their mean productivities for the simulation are different: the unmodified run has an average productivity of 8308 tph while the modified run with the intersection removed is slightly more productive at 8411 tph. This is an increase of 1.3 %. Using the same approach and assumptions used in the previous section, this improvement was analyzed and is presented in Table 5-9.
Figure 5-24- Experiment 3 results

Table 5-9- Economic analysis of Experiment 3 results

<table>
<thead>
<tr>
<th></th>
<th>With intersection</th>
<th>Without intersection</th>
<th>change</th>
</tr>
</thead>
<tbody>
<tr>
<td>Production rate, hourly</td>
<td>8,308</td>
<td>8,411</td>
<td>103</td>
</tr>
<tr>
<td>Production rate, daily</td>
<td>182,776</td>
<td>185,042</td>
<td>2,266</td>
</tr>
<tr>
<td>Avg. grade of material (oz/t)</td>
<td>0.00835</td>
<td>0.00835</td>
<td>-</td>
</tr>
<tr>
<td>Gold ounces moved per day (oz)</td>
<td>$1,526</td>
<td>1,545</td>
<td>19</td>
</tr>
<tr>
<td>Gold price ($)</td>
<td>$1,500</td>
<td>$1,500</td>
<td>-</td>
</tr>
<tr>
<td>Revenue, per day ($)</td>
<td>$2,289,269</td>
<td>$2,317,651</td>
<td>$28,382</td>
</tr>
<tr>
<td>Revenue, per year ($)</td>
<td>$835,583,331</td>
<td>$845,942,633</td>
<td>$10,359,302</td>
</tr>
<tr>
<td>% improvement</td>
<td>-</td>
<td>-</td>
<td>1.25 %</td>
</tr>
</tbody>
</table>

The results in Table 5-9 show that removing the effects of an intersection could yield an increase in gold production of 19 oz per day or $10.3M in revenue per year. Again, as a
cautionary note, this estimate is based on a simplified dispatch scenario, meant only to give a rough productivity estimate of this scenario. Specifically, this analysis only included one intersection, Q3INT, which was both the busiest and most travelled during the data collection period. Since the physical configuration and actual usage of every intersection are unique, this result is therefore unique to the given scenario. It would be imprudent to 'scale up' this result: instead, more complex simulations should be conducted with a higher level of detail. However, the net result shows a clear improvement in productivity related to intersection removal, and also gives a useful estimate of the potential benefits. Hopefully this is useful to future researchers that will look at road network configurations for semi- or fully-autonomous vehicles that can be dispatched in such a way as to eliminate intersection effects altogether. It could also prove useful in future cost-benefit analyses of alternative intersection configurations such as under/over passes, roundabouts, etc.
Chapter 6

Summary and Conclusions

Haul time variability is an observed reality of current open pit mining operations. The main goals of this thesis were to measure this variability, identify its sources, and explore its impact on haulage systems and in turn overall productivity. These goals were achieved by structuring the research in five main chapters that individually tackle specific areas of the problem. The order in which they are collectively presented closely resembles the actual project timeline, each building on the results of the previous. In the interest of consistency, the next section summarizes the research and contributions from each chapter, individually.

Chapter 1- Introduction

The motivation for this research was presented, mostly related to productivity increases, cost savings and reduced emissions. In order to effectively and realistically investigate haul time variability, a suitable test site was deemed necessary for data collection. The Goldstrike Open Pit mine, owned and operated by Barrick Gold Inc., was selected for its large scale, complex road network and ease of access. Certain key concepts such as productivity and material flow were defined, as was the scope of work and thesis structure.

Chapter 2- Dispatch Systems

A background of fleet management systems, and more specifically dispatch systems, was presented. The fundamentals are well summarized in [12] and [17]. Relevant dispatching strategies (such as those in [21]) along with a history of commercialized systems provide an overview of the state-of-the-art. The functioning of the MMS dispatching system is discussed in
depth, mostly based on information seen in [11] and [17]. Certain observations of how variability might affect the dispatching algorithms are introduced. The contribution of this chapter is a thorough literature review focusing on the functioning of MMS DISPATCH® software, including its advantages as well as its possible shortcomings. The significance of this contribution is emphasized due to the lack of published, unbiased information related to this subject.

Chapter 3- Measuring Variability in Haul Times

This Chapter includes a literature review that provides a summary of techniques and motivations for variability measurements in a variety of industries and applications, such as passenger vehicle flow [56] transport and air traffic control [55]. The experiment design for the data collection period carried out at Goldstrike is described, followed by the data requirements, methodologies and the reasons for using standalone GPS units. The raw data is presented and discussed at length along with graphics, such as Figure 3-3, to exemplify its depth and usefulness. Then, the logic behind the MATLAB® data analysis modules is presented, showing how the data were imported, parsed, sorted, visualized and manipulated. Finally, the variability in a selection of Goldstrike haul legs is discussed, showing that data and analysis techniques were successful to this end.

The raw data set collected in the field and presented in this section is a significant contribution to this field of research. Despite the exhaustive literature review for this section, no similar data sets were found. It is likely that similar data has been produced, but it is virtually non-existent in the public domain. In this context, the characteristics that make this positioning data unique are as follows:

- The quality of the positioning data set is very high, due to excellent satellite coverage and virtually no signal loss observed during the data collection period;
• The data set provides very high resolution for position tracking and in turn an accurate estimation tool for speeds and even acceleration, due to the use of a relatively high sampling rate of 1 Hz;

• The data set is diverse, representing a cross-section of real life operation at a large scale open pit mine, including a complex road network and a variety of haulage scenarios;

• In addition to its positioning and temporal components, the data set contains a wealth of other ‘minable’ information such as operator, shift, truck number, etc. This provides the opportunity for a wide range of potential research.

Although the raw data set is highly versatile and certainly adaptable to different fields of research, the original approach designed and implemented in this thesis can be considered as another significant contribution to this field. The code was written from scratch with future users in mind, formulated to be accessible and customizable. This code (see Appendix C) along with the original data, will remain available to future researchers at Queen’s as well as those at Barrick Gold Inc.

Chapter 4- Sources of Haul Time Variability

An extensive literature review was prepared to demonstrate the relevance of travel time variability and its sources in various industries. Precedents in air traffic [49] and in passenger vehicle traffic [56] discuss a wide variety of variability sources, from such phenomena as airport ground delays to driver aggressiveness. This research from other fields, combined with real life observations at the open pit mine, led to the definition of several parameters affecting haul times shown in Section 4.2 - Table 4-1. Of these parameters, several were identified as having significant influence on haul time variability and subsequently analyzed: for example, trucks
travelling along haul legs with intersections were shown to exhibit higher levels of travel time variability than others.

The contributions from this chapter relate to the definition of specific sources of haul time variability in haul times and their identification based on statistical analyses of the data. Furthermore, the results are annotated using the Candidate’s real-life observations from the data collection period, adding a realistic perspective to the discussion.

Chapter 5- Effects of Haul Time Variability

Various analyses were performed on different types of hauls to investigate the relationship between travel time variability and sets of overall mean haul times. Variability was generally observed to negatively affect the overall haul times in all cases, to different degrees. Empty, non-speed limited (i.e., decline and/or flat profile) hauls were found to not only contain the highest levels of variability, but were also found to be most sensitive to variability reduction: removing variability from these haul times led to significant improvements in overall haul times. This important observation led to these empty/decline/flat hauls being identified as prime candidates for speed control practices, which if implemented could lead to immediate improvements in cycle times.

In Section 5.3, valuable information pertaining to truck behaviour at and near intersections was extracted from the raw data and presented. A wide range of intersection crossing times was observed, showing that trucks not only stopped for extended periods of time at these locations, but they were also observed to ‘roll through’ on occasion, in violation of mine regulations. Furthermore, analyses of intersection variability reduction and theoretical intersection removal were performed. Each was shown to have positive effects on overall mean haul times, which were
quantified in the discussion. This is valuable information to not only the current mine operators but future designers and researchers that might want to explore different road network configurations and/or dispatch strategies.

Finally, the resulting data sets from these analyses were used as inputs for a simple dispatch simulator with the goal of providing order-of-magnitude estimates of both productivity improvements and revenue increases. The simulator was adapted to resemble the Goldstrike operational layout as closely as possible, using typical production levels, haul distances, and queuing times. The results from these dispatch simulations yielded quantifiable estimates of potential benefits in terms of tons, ounces and dollars per unit time. These estimates represent a significant contribution to the field since nothing similar was found to exist in the public domain.

6.1 Concluding Remarks

As open pit mines get larger and their road networks more complex, haul traffic is an area of increasing concern for mine operators. In particular, engineers from Barrick Gold Inc. have observed haul truck interactions in some of their larger open pit mines that are clearly disruptive to the haulage cycle. Such traffic effects cause uncertainty in haul times over identical routes; commercial dispatch systems do not account for this variability. Although there are many studies addressing these problems in other industries, there are very few specific to the mining industry. Current commercial dispatch systems are proprietary in nature, making it difficult for researchers to suggest possible improvements in this technology.

In the near term, the data generated for this research has the potential to support a number of related studies in this field such as:

- Investigating new road network configurations for open pit mines;
• More in-depth dispatch simulation to either improve the performance of current systems or develop new dispatching algorithms and/or strategies.

As the global economy shifts from an antiquated, narrow-minded mentality to one that embraces information-sharing, there will be many more opportunities for meaningful research in the field of open pit mine traffic optimization. For now, one of the most significant challenges lies in the age-old struggle between researchers and profit driven proprietary commercialization. It is hoped that this barrier will be broken in the near future, thereby allowing a free exchange of information that will certainly benefit all parties involved.
Bibliography

http://www.nbmg.unr.edu/dox/b111/goldstrike.pdf


http://www.hitachimining.com/ex5500-6.html

http://www.minepro.com/MinePro/Literature/Brochures/XS43591_PH_Shovels_Brochure.pdf


137
http://www.komatsuamerica.com/trucks-930E-4

http://www.mathworks.com/products/matlab/

[49] Michael Ball et al., "Total Delay Impact Study: A Comprehensive Assessment of the Costs
and Impacts of Flight Delay in the U.S.,” NEXTOR, Washington, D.C., Academic research
sponsored by industry 2010. [Online].


[53] O. Richetta and A. Odoni, "Dynamic Solution to the Ground-Holding Policy Problem in Air


[58] L. Fu and B. Hellinga, "Delay Variability at Signalized Intersections," Transportation
Research Record: Journal of the Transportation Research Board, vol. 1710, pp. 215-221,
2000.


APPENDICES
Appendix A

Truck Specifications

MAXIMUM GVW
565755 kg 1,115,000 lb
GROSS HORSEPOWER
2611 kW 3,500 HP

KOMATSU
930E-4SE

ELECTRIC DRIVE TRUCK

Photo may include optional equipment.
930E-4SE Electric Drive Truck

**Specifications**

### Engine

<table>
<thead>
<tr>
<th>Make and model</th>
<th>Komatsu SSDA18V170</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fuel</td>
<td>Diesel</td>
</tr>
<tr>
<td>Number of cylinders</td>
<td>8</td>
</tr>
<tr>
<td>Operating cycle</td>
<td>4 cycle</td>
</tr>
<tr>
<td>Gross horsepower</td>
<td>201 kW / 3,000 HP @ 1,900 rpm</td>
</tr>
<tr>
<td>Net flywheel power</td>
<td>255 kW (3,425 HP @ 1,900 rpm)</td>
</tr>
<tr>
<td>Weight (air)</td>
<td>10,714 kg 23,166 lb</td>
</tr>
</tbody>
</table>

Note: Net flywheel power is the rated power at the engine flywheel minus the average accessory loads. Accessories include fan and charging alternator.

### Electric Drive

<table>
<thead>
<tr>
<th>AC Current</th>
<th>G4A-39</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dual impeller in-line blower</td>
<td>450 m³/min 16,000 cfm</td>
</tr>
<tr>
<td>Control</td>
<td>AC Torque Control System Motorized wheels</td>
</tr>
<tr>
<td>Speed (maximum)</td>
<td>64.5 km/h 40 mph</td>
</tr>
</tbody>
</table>

Note: Wheel motor application depends upon gross vehicle weight, haul road grade, haul road length, designer resistance and other parameters. Komatsu and O.E. must analyze each case condition to assure proper application.

### Tires and Rims

<table>
<thead>
<tr>
<th>Tires</th>
<th>530/80R26</th>
</tr>
</thead>
<tbody>
<tr>
<td>Taper Lock</td>
<td>914 mm x 1600 mm x 127 mm 36° x 63° x 5° rim assembly</td>
</tr>
<tr>
<td>Rated</td>
<td>758 kPa 110 psi cold inflation pressure for rim, 600 kPa 87 psi for tires</td>
</tr>
</tbody>
</table>

Note: Tire should meet application requirements for high speed, heat, compound inflation pressure, ply rating or equivalent, etc.

### Body

All-welded steel flat floor body with horizontal bolsters and full canopy, telescopable, rear week box, ejection, body up, and rubber mounts on frame are standard. Pivot exhaust heating optional.

| Floor sheet             | 16 mm 0.63” 1379 MPa 200,000 psi tension steel (two-piece) |
| Front side              | 9 mm 0.35” 1379 MPa 200,000 psi tension steel |
| Side sheet              | 8 mm 0.31” 1379 MPa 200,000 psi tension steel |
| Canopy sheet            | 5 mm 0.19” 660 MPa 100,000 psi tension steel |

Standard SAE heated 211 mm 276 yard | Komatsu body weight 3025 kg 66,986 lbs |

Note: Komatsu must approve all bodies through a Body Application Worksheet.

### CAB

Advanced Operator Environment with integral 4-post ROPS/FOPS structure (meets J1040 Apr88), adjustable air suspension seat with lumbar support and arm rests, passenger seat, maximum 7-value insulation, tilt and telescoping steering wheel, electric windshield wipers, windshield washers, tinted safety glass, power windows, Komatsu Fayhood Weighting System, 55,000 Btu/hr heater and defroster, 21,000 Btu/hr air conditioning (RFP - 134A Refrigerant).

### Frame

Advanced technology, full box-welded box sectional ladder-type frame with integral ROPS support, integral front bumper, rear Subframe members, steel castings at all critical stress transition areas, napped continuous hose ccollar.

| Plate material         | 482.6 MPa 70,000 psi tension steel |
| Castling material      | 620.5 MPa 90,000 psi tension steel |

| Rail width             | 305 mm 12” |
|                       | 25 mm 0.98” Drive axle mounting |

Note: Swing link between frame and axle.

### Braking System

Service brakes: oil-cooled, hydraulic-activated, multiple disc brakes at each wheel. Traction system wheel pad/disc control.

| Max. service apply pressure | 17237 kPa 2500 psi |
| Total friction area per brake | 9725 cm² 19586 in² |

Secondary brakes: Automatically applied prior to hydraulic system pressure dropping below level required to meet secondary stopping requirements.

| Wheel brake locks       | Switch activated Parking brakes |
|                       | Multiple disc, spring-applied, hydraulically-released, dry brakes on board end of each wheel motor rotor shaft. Rated to hold on at 15% grade @ maximum gross vehicle weight. |
| Electric dynamic retarder | Max 4506 kW 6400 hp |

Continuous 2500 kW 3900 hp

### Suspension

Variable rate hydro-pneumatic with integral rebound control.

| Max. front stroke       | 306 mm 12.82” |
| Max. rear stroke        | 230 mm 9.04” |
| Max. rear axle oscillation | ±6.5” |

Note: Rear suspension is inverted.

### Cooling System

L&M radiator assembly, split-flow, with de-aeration-type top tank measuring 3277 mm x 2154 105” x 71”.

### Service Capacities

| Cooling System         | 719L 190 U.S. gal |
| Crankcase              | 341L 90 U.S. gal |
| Hydraulic system       | 1325 L 350 U.S. gal |
| Motor gear box (each)  | 30 L 25 U.S. gal |
| Fuel                   | 5300 L 1400 U.S. gal |

Note: Includes type of filter.
**ELECTRIC DRIVE TRUCK**

**DIMENSIONS**

All dimensions are with 218 m³/285 yd³ body (optional body).

<table>
<thead>
<tr>
<th>Bodies</th>
<th>Struck</th>
<th>2:1 Heap</th>
<th><em>Loading Height</em></th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard</td>
<td>171 m³/224 yd³</td>
<td>211 m³/276 yd³</td>
<td>7.06 m/232&quot;</td>
</tr>
</tbody>
</table>

* Exact load height may vary due to tire make, type, and inflation pressure.

**HYDRAULIC SYSTEM**

Steering: SAE approved double-acting cylinders provide constant rate steering. Secondary steering automatically supplied by accumulator.

- Power-oil pressure: 25.7 m³/91°F
- Reservoir: 9.87 L/200 US gal
- Filtration: In-line replaceable elements
- Suction: Single, full-flow, 100 mesh
- Hoses: Dual, in-line, high pressure
- Brake component cabinet: Above deck, easily accessible with diagnostic test connections.

- Hoist: Two-stage dual-acting outboard cylinders, internal cushion valve, over-center damping.
- Hoist times:
  - Power-up loaded: 21 sec
  - Power-up unloaded: 23 sec
  - Float-down empty: 24 sec

- Pumps: Two pumps, single package, end of alternator
- Tandem gear pump with output of 1022 lpm 270 gpm at 1900 rpm and 1723 kPa 2,906 psi

- Steering and brake: Pressure-compensating axial piston pump with output of 244 lpm 65 gpm at 1900 rpm and 18961 kPa 2,750 psi
- System relief pressure: 1723 kPa 2,900 psi
- Steering and brake: 27279 kPa 4,000 psi

**ELECTRICAL SYSTEM**

- 4 x 108 and 2 x 30, 220-amp-hour batteries, bump-mounted with disconnect switch.
- Alternator: 24 volt, 140 amp
- Lighting: 24 volt
- Cranking motors: two 64 volt

**WEIGHT DISTRIBUTION**

- Empty Vehicle: 194459 kg, 230,293 lb, 48.5%
- Rear axle: 399549 kg, 478,769 lb, 72.7%
- Total (wet, 100% fuel): 595108 kg, 1,115,060 lb

**Note:** GVW shall not exceed 507500 kg, 1,115,060 lb including options, tires, fuel and payload, subject to applicable approval by Komatsu.

**Note:** Figures and illustrations may show optional equipment. Materials and specifications subject to change without notice.
STANDARD EQUIPMENT

- Air cleaners, Donaldson SFG
- Air filter screens (2)
- Alternator (24 volt/140A)
- Auto lubrication system (Lincoln)
- Windshield wiper system (level indicator)
- Battery (6 x 9D and 2 x 30H (1450CCA))
- Battery charging cable and socket
- Body over center device
- Brakes: oil-cooled, multiple disc front & rear
- Circuit breakers, 24 volt
- Control cabinet
- Electric start
- Eliminator system for engine lubrication
- Filters, high pressure hydraulic
- Gate valves on hydraulic tank
- Ground level radiator fill
- Komatsu Padded Weighing System-PLM-III
- Mirrors, LH flat and RH rectangular convex
- Mud flaps
- Mounted exhaust—deck-mounted
- Power supply, 24 volt and 12 volt DC
- Quick disconnects (hoist and diagnostics)
- Radiator sight gauge
- Removable power module unit (radiator, engine, alternator, blower)
- Retarder speed control
- Reverse retarder
- Rock ejectors
- Thermosetic Fan Clutch
- Wiggins Fast-Fill Fuel System (in tank and left side remote)
- Service Center-LH
- Body Impact Plate

OPERATOR ENVIRONMENT & CONTROL

- All hydraulic service brakes with secondary auto apply
- Battery disconnect switch
- Body on sling
- Brake lock and drive system interlock
- Circuit breakers, 24 volt
- Diagonal staircase across grille, L to R
- Dynamic rearming with continuous rated element grog
- Engine shutdown at ground level
- Hoist propulsion interlock
- Horns (electro-fonic and back-up)
- Integral ROPS/FOPS Cab Level 2

STANDARD HIGH VISIBILITY DELUXE CAB:

- AC Drive interface Display
- Air cleaner vacuum gauges
- Air conditioner HFC-134A
- Alarm system warning lights red
- Accumulator pre-charge
- Brake hydraulic oil temperature warning
- Coolant level warning
- Drive system temperature warning
- Electric system fault warning
- Fold-out rear view mirror
- Hydraulic oil level warning
- Low engine oil pressure warning
- Low brake pressure warning
- Side mirror power warning
- Stop engine no propeller warning
- Steering pressure warning
- Traction control warning
- Windshield level warning
- Windshield wiper (dual) and washer (electric)

LIGHTING:

- Back-up lights—rear mount (2) halogen
- Backup lights—R and L—deck mount (2) halogen
- Brake and retard lights on top of cab
- Clearance lights
- Control cabinet service light (LED)
- Dynamic rearming, rear (2) halogen
- Engine compartment service lights
- Fog lights (2) halogen
- Headlights (8) halogen
- Stairway lights
- Manual back-up light, switch and indicator
- Platform lights R, L and Center
- Step & tail lights (2) (LED)
- Turn signals halogen
- Under-hood service lights

OPTIONAL EQUIPMENT

(Optional equipment may change operating weight).

- Body Liners
- Fire extinguisher 9 kg 29 lb
- Heated Body
- Hot start engine coolant (200V 2-2500W)
- Hot start engine oil (200V 2-600W)
- Hydrometer (kgs or liters)
- Modular Mining Systems (MMS) ready
- Mufflers between frame rails
- Radiator shutters
- Service Center—PH
- Special language decals
- Suspension, cold weather

*Available factory installed on new-installed. All other options and accessories listed are available factory installed only.

AESS778-01 ©2009 Komatsu America Corp. Printed in USA D0113c 01/09 (EV-1)

Komatsu America Corp. is an authorized licensee of Komatsu Ltd. Materials and specifications are subject to change without notice.

www.KomatsuAmerica.com

144
Appendix B

Graph Representation of Haulage Network
Appendix C
MATLAB® Code

C-1 IMPORT

% IMPORT.m - A script for importing data from Qstarz GPS logger.
% the following code removes the header line, then imports & formats
% the entire data set.
clear all;
close all;

disp('***IMPORT.m - a script for importing data from Qstarz GPS
logger.***')
disp('Select file to import in the dialog box.');

[filename, pathname] = uigetfile('*.csv', 'Select the CSV file to
import');
path = [pathname, filename];
 fid = fopen(path);
 C = textscan(fid, '%f %s %s %s %s %s %f %s %f %s %f %s %fM %fkm/h %f %f
  %f %f %f %f %s %s %fM', 'delimiter', ',', 'HeaderLines', 1);
fclose(fid);

% Choose the desired data and store it in labelled arrays.
index = C{1};
lat = C{9};
lon = C{11};
ele = C{13};

% date and time manipulations to give EVERY data point a unique
identifier
 time_crap = C{6};
date_crap = C{5};
time_combined = strcat(date_crap, time_crap);
time = datenum(time_combined, 'yyyy/mm/ddHH:MM:SS');
starttime_serial = time(1);
stoptime_serial = time(size(index, 1));

% Create mat file for plotting
nametemp = regexprep(filename, '.csv', '')
newname1 = [nametemp, '_raw'];
save(newname1, 'index', 'lon', 'lat', 'ele', 'time', 'filename');

disp('Select a location to save the new .mat file!')
foldername = uigetdir('*.mat', 'Select a save location:');
f = fullfile(foldername, newname1);
save(f);
disp1 = sprintf('the unit STARTED recording at %s.', datestr(starttime_serial));
disp2 = sprintf('the unit STOPPED recording at %s.', datestr(stoptime_serial));
disp3 = sprintf('data successfully imported to %s.mat.', f);
disp(disp1);
disp(disp2);
disp(disp3);

C-2 POLYGONS

% POLYGONS.m - This script was used to define polygons for the
% complete set of Qstarz data. Polygons are used to define nodes such
% as shovels and dumps.
% Polygon files form the basis for each analysis- every unique set of
% polygons will have a unique result.
% This is a first step towards parsing the data.
close all;
clear all;

% Enter desired number of split points (i.e., nodes)
nsplits = input('Enter number of desired nodes: ');

% Plot of Latitude-Longitude for ginput- see combinedplot.m
f = fullfile('C:', 'Users', 'Andrew.adm', 'Dropbox', 'Public', 'MTO',
'Matlab files', '_Qstarz MATLAB files', 'VISUALIZATION',
'combinedplot.m');
run(f);

disp('Click on vetices to define desired polygon of exclusion: [enter
to close]');
disp('');
disp('Make sure the polygons do not overlap or the program will not
work!');

% nsplits is the number of nodes- z contains all vertices of user-
defined
% polygon. This loop runs from 1..nsplits, once for each polygon to be
% defined.
for i = 1 : nsplits

    % allow zooming before polygon definition for more accurate polygon
    % boundary definition
    disp('');
disp('Zoom to desired area of interest, press SPACEBAR to begin
polygon definition:');
zoom on;
waitfor(gcf,'CurrentCharacter');

    % user inputs polygon vertices by clicking in MATLAB axes
[x,y] = ginput;
patch(x,y,'y');
str = sprintf('Give split point %d of %d a name. No spaces.' ,i, nsplits);
disp(str);

% allow user to label each polygon
splitname(i,1) = {input('Name: ' , 's')};
pause(1);
splitname(i,2) = {input('Is this a shovel (S), dump (D), queue (Q), intersection (I), or Flag (F): ' , 's')};
text(x(1), y(1), splitname(i,1));

z = [x y];
% save polygon vertices into separate variables from 1..nsplits
str1 = ['poly_' num2str(i) ' = z;'];
eval(str1);

leave = input('
Continue? Y/N [Y]: ' , 's');
if isempty(leave) || leave == 'y' || leave == 'Y'
    else
        nsplits = i;
        break
    end

% create temporary polygon file based on iteration for use in the inpolygon function in downstream programs.
str2 = ['poly_temp = poly_' num2str(i) ';'];
eval(str2);

% reset zoom
zoom out;
end

reply = input('Are you satisfied with your exclusion zones? Y/N [Y]: ' , 's');
if isempty(reply) || reply == 'y' || reply == 'Y'
    disp(' '); 

% name polygon file. A descriptive name is useful if many polygon % files will be created.
filename = input('Enter description for polygon file (i.e. Dumps, Shovels, etc.): ' , 's');
foldername = [datestr(now,1), ' ',filename, ' polygons'];
filename = [filename, '_polygons'];
timestamp = datestr(now);

% The next few lines define `runfolder`, a variable that plays a key role
% in downstream programs. It contains a path that points to the location of the polygon file.
mkdir('C:\Users\Andrew.adm\Dropbox\Public\MTO\Matlab files\_Qstarz MATLAB files\zzz_OUTPUT files\', foldername);
runfolder = fullfile('C:', 'Users', 'Andrew.adm', 'Dropbox', 'Public', 'MTO', 'Matlab files', '_Qstarz MATLAB files', 'zzz_OUTPUT files', foldername);
f = fullfile(runfolder, filename);
save(f, 'poly_*', '*split*', 'timestamp', 'runfolder');

disp(' ');
disp(' ');
disp4 = sprintf('polygon data saved to %s.mat.', filename);
disp(disp4)
return;

% If user replies 'N', the program is restarted—useful if you make a mistake.
elseif reply == 'N' || 'n'

disp('Try again!');
disp(' ');
disp(' ');
disp(' ');
POLYGONS;

C-3 PARSE
% PARSE.m - parse data using polygon exclusion zones
clearvars -except runfolder;
close all;

disp(' * * * * *POLYGON FILE SELECTION* * * * *');
polynametemp = input('Enter polygon filename to be processed: ', 's');
polynamel = [polynametemp, '_polygons.mat'];

% From here on in, most programs are set up to run single files, a batch of
% files or all files, as seen in the if statement below.
batch = input('Single File (S), Batch (B), or All (A) run?: ', 's');
if batch == 'S' || batch == 's'
count = 1;
elseif batch == 'B' || batch == 'b'
count = input('enter number of files to be run: ');
elseif batch == 'A' || batch == 'a'
clearvars -except runfolder polyname;
rawdatafolder = fullfile('C:', 'Users', 'Andrew.adm', 'Dropbox', 'Public', 'MTO', 'Matlab files', '_Qstarz MATLAB files', '0-DATA', 'mat');

% find all files and list
files = getAllFiles(rawdatafolder);
allfiles = length(files);
cnt = 0;
runfiles = {};
for a = 1 : allfiles
    if length(char(files(a))) ~= 125
        cnt = cnt + 1;
        runfiles(cnt,1) = files(a);
        allfiles = allfiles-1;
    end
end

% count is the number of files to be run, and also the counter for the main program loop.
count = numel(runfiles);
end

% Main program loop
for i = 1 : count
    clearvars -except runfolder runfiles polyname count i retainpolygon;
    close all;
tic;

    % load polygon file
    f = fullfile(runfolder, polyname);
    load(f);

    disp('  *  *  *  *DATA FILE SELECTION*  *  *  *');
    filecount = sprintf('  *  *  *  *file %d of %d*  *  *  *', i, count);
    disp(filecount);
    disp('  ');
    exist = exist('runfiles','var');

    if exist == 1
        filetemp = char(runfiles(i));
        load(filetemp);
    else
        rawnamein = input('Enter raw data filename to be processed (no _raw): ', 's');
        rawname = [rawnamein, '_raw.mat'];
f1 = fullfile('C:', 'Users', 'Andrew.adm', 'Dropbox', 'Public', 'MTO', 'Matlab files', '_Qstarz MATLAB files', '0-DATA', 'mat', rawname);
load(f1);
end

% run gpshaversine.m to calculate distances, velocities, etc.
% See it for details.
f2 = fullfile('C:', 'Users', 'Andrew.adm', 'Dropbox', 'Public', 'MTO', 'Matlab files', '_Qstarz MATLAB files', '5a- CALCS', 'HAVERSINE.m');
run(f2);

% Next loop checks all points and evaluates whether or not they fall within a defined polygon. If so, the data point is flagged with a number corresponding to the polygon # and stored in a column of matrix split. The size of matrix split is (index,nsplits).
for w = 1 : nsplits
    clearvars counttemp;
    str1 = ['poly_temp = poly_ ' num2str(w) ';'];
    eval(str1);
    IN = inpolygon(lon, lat, poly_temp(:,1), poly_temp(:,2));
    str2 = ['split(:, ' num2str(w) ') = IN*w;'];
    eval(str2);
end

% this adds the rows of matrix split so that we are left with a column vector with values of [0:nsplits].
split_combined = sum(split')';
disp('* * * * * * * * Step 3 - parse data * * * * * * * * ');

% this loop pads each end of the split_combined segments with the split_combined-1, so that after parsing we can still identify where the splits occurred.
splitflag = split_combined;

% initialize the temporary variables to that will store the parsed data- their length is equal to the number of elements == 0 in split_combined.
for m = 2:(numel(index))
    if split_combined(m-1,1) == 0 && split_combined(m,1) > 0
        splitflag(m-1,1) = split_combined(m,1)-1;
    end
    if split_combined(m-1,1) > 0 && split_combined(m,1) == 0
        splitflag(m,1) = split_combined(m-1,1)-1;
    end
end
% this loop stores the value in the nth row of the corresponding
temp variable if its split_combined is == 0. (i.e. not in exclusion
zones)
nsplitscount = nsplits;
for z = 1 : nsplitscount
    
    index_temp =
zeros(numel(split_combined(split_combined==z)),1);
    lon_temp =
zeros(numel(split_combined(split_combined==z)),1);
    lat_temp =
zeros(numel(split_combined(split_combined==z)),1);
    ele_temp =
zeros(numel(split_combined(split_combined==z)),1);
    time_temp =
zeros(numel(split_combined(split_combined==z)),1);
    vel_temp =
zeros(numel(split_combined(split_combined==z)),1);
    dist_temp =
zeros(numel(split_combined(split_combined==z)),1);
    
    for n = 1 : numel(index)
        if split_combined(n,1) == z
            index_temp(n,1) = index(n);
            lon_temp(n,1) = lon(n);
            lat_temp(n,1) = lat(n);
            ele_temp(n,1) = ele(n);
            time_temp(n,1) = time(n);
            vel_temp(n,1) = vel(n);
            dist_temp(n,1) = dist(n);
        end
    end
    
    Stemp = [ index_temp lon_temp lat_temp ele_temp time_temp
    vel_temp dist_temp ];
    Stemp(~any(Stemp,2),:) = [];
    str5 = ['S' num2str(z) ' = Stemp;'];
    eval(str5);
    clearvars Stemp index_temp lon_temp lat_temp ele_temp
    time_temp vel_temp dist_temp
    % z is displayed while running to monitor program progress.
    z
end

% this loop stores the value in the nth row of the corresponding
temp variable if its split_combined is == 0. (i.e. not in exclusion
zones)
index_temp = zeros(numel(split_combined(split_combined==0)),1);
lon_temp = zeros(numel(split_combined(split_combined==0)),1);
lat_temp = zeros(numel(split_combined(split_combined==0)),1);
ele_temp = zeros(numel(split_combined(split_combined==0)),1);
time_temp = zeros(numel(split_combined(split_combined==0)),1);
vel_temp =  zeros(numel(split_combined(split_combined==0)),1);
dist_temp =  zeros(numel(split_combined(split_combined==0)),1);

for v = 2 : numel(index)
    if split_combined(v,1) == 0
        index_temp(v,1) = index(v);
        lon_temp(v,1) = lon(v);
        lat_temp(v,1) = lat(v);
        ele_temp(v,1) = ele(v);
        time_temp(v,1) = time(v);
        vel_temp(v,1) = vel(v);
        dist_temp(v,1) = dist(v);
    end
end

% This code checks the first line that is missed in the
% preceding for loop that runs from 2 : numel(index).
if split_combined(1,1) == 0
    index_temp(1,1) = index(1);
    lon_temp(1,1) = lon(1);
    lat_temp(1,1) = lat(1);
    ele_temp(1,1) = ele(1);
    time_temp(1,1) = time(1);
    vel_temp(1,1) = vel(1);
    dist_temp(1,1) = dist(1);
end

%S0 is a grouping of all the temp variables to facilitate
% exporting out of the loop.
S0 = [ index_temp lon_temp lat_temp ele_temp time_temp vel_temp dist_temp ];
S0(~any(S0,2),:) = [];
clearvars Stemp index_temp lon_temp lat_temp ele_temp time_temp
vel_temp dist_temp;

% remove all values from splitid that are within the exclusion
% zones, so it will match the parsed data later. Since the splitid's
% are NEGATIVE numbers, they will remain for future reference.
splitid = splitflag(splitflag<0.5);
splitidinpoly = splitflag(splitflag~=0);

% clear variables containing raw data, to be replaced with the
% parsed data
clear index lon lat ele time vel dist;

% Make a directory in current runfolder to store parsed data
mkdir(runfolder, 'NodeData');
parentsavedir = fullfile(runfolder, 'NodeData');
%This next loop extracts data from the temporary variables from above
for y = 0 : nsplits
    if y == 0
        % assign parsed data into variables. zero rows are removed (same names)
        index = S0(:,1);
        lon = S0(:,2);
        lat = S0(:,3);
        ele = S0(:,4);
        time = S0(:,5);
        vel = S0(:,6);
        dist = S0(:,7);

        % save new .mat file with '_parsed' appended
        extst1 = exist('filetemp','var');
        if extst1 == 1
            rawname = filetemp;
            [pathstr, filenametemp, ext] = fileparts(rawname);
            filenametemp1 = regexprep(filenametemp, 'GSpit_', '');
            filenametemp2 = regexprep(filenametemp1, '_raw', '');
            parsename = [filenametemp2, '_parsed.mat'];
            mkdir(runfolder,'ParsedData');
            legsavedir = fullfile(runfolder, 'ParsedData', parsename);
            save(legsavedir, 'runfolder', 'rawname', 'polyname', 'parsename', 'index', 'time', 'lon', 'lat', 'ele', 'splitid', 'dist', 'vel', 'speedlimit', 'splitname', 'nsplits');
        else
            nametemp = regexprep(rawname, '_raw.mat', ' '); parsename = [nametemp, '_parsed.mat']; mkdir(runfolder,'ParsedData'); legsavedir = fullfile(runfolder, 'ParsedData', parsename);
            save(legsavedir, 'runfolder', 'rawname', 'polyname', 'parsename', 'index', 'time', 'lon', 'lat', 'ele', 'splitid', 'dist', 'vel', 'speedlimit', 'splitname', 'nsplits');
        end
    end

    %The following code, until line 206, plots the polygons along with the raw data, and a plot for speeds. hold all makes each polygon remain on plot until the end
    Xpoly = [ ];
    Ypoly = [ ];
    figure(20);
    hold all

    for j = 1 : nsplits
        strX = ['Xpolytemp = poly_' num2str(j) '(:,1); '];
eval(strX);
strY = ['Ypolytemp = poly_ ' num2str(j) '(:,2);'];
eval(strY);
patch(Xpolytemp,Ypolytemp,'y');
text(Xpolytemp(1), Ypolytemp(1), splitname(j,1));

Xpoly = [Xpoly; Xpolytemp];
Ypoly = [Ypoly; Ypolytemp];
end

% P
lot of X-Y
plot(lon,lat,'.-')
axis equal;
set(gca,'XDir','Reverse')
title('X-Y View of Data')
xlabel('Longitude')
ylabel('Latitude')

% Plot speeds
figure(2);
plot(time, vel,'.-')
title('speed vs. time')
xlabel('time')
ylabel('velocity')

% display information about parsed data
disp(' ');
disp(' ');
disp5 = sprintf('%d data points of %d were removed
(within exclusion zones).',
numel(split_combined(split_combined==0)), numel(split_combined));
disp6 = sprintf('%d data points of %d were retained.',
numel(split_combined(split_combined==0)), numel(split_combined));
disp7 = sprintf('Data successfully parsed. Saved to
%s.', parsename);
disp(disp5);
disp(disp6); 
disp(' ');
disp(disp7);
disp(' ');
clear index lon lat ele time vel dist nametemp nametemp1

splitid f;
else
  savenamedir = ['split' num2str(y,'%02d')];
  mkdir(parentsavedir, savenamedir);
  savedir = fullfile(parentsavedir, savenamedir);

  str6 = ['index = S' num2str(y) '(:,1);'];
  str7 = ['lon = S' num2str(y) '(:,2);'];
  str8 = ['lat = S' num2str(y) '(:,3);'];
  str9 = ['ele = S' num2str(y) '(:,4);'];
str10 = ['time = S' num2str(y) '(:,5);'];
str11 = ['vel = S' num2str(y) '(:,6);'];
str12 = ['dist = S' num2str(y) '(:,7);'];
str13 = ['splitid = splitidinpoly(splitidinpoly==abs(' num2str(y) '));'];

eval(str6);
eval(str7);
eval(str8);
eval(str9);
eval(str10);
eval(str11);
eval(str12);
eval(str13);

% save new .mat file with '_parsed' appended
exst1 = exist('filetemp', 'var');
if exst1 == 1
    rawname = filetemp;
parsename2 = regexprep(rawname, '_ raw.mat', '');
parsename3 = [parsename2, '_split' num2str(y, '%02d') '_parsed.mat'];
parsename = parsename3(1, 86:121);
end

f = fullfile(savedir, parsename);
save(f, 'savedir', 'parentsavedir', 'runfolder', 'rawname', 'polyname', 'parsename', 'index', 'time', 'lon', 'lat', 'ele', 'splitid', 'dist', 'vel', 'speedlimit', 'splitname', 'n_splits');
clear savenamedir index lon lat ele time vel dist nametemp nametemp1 splitid f;
end
end
toc;

C-4 HAVERSINE (function called in PARSE)

% HAVERSINE.m - Script for converting lat, lon & ele GPS data to metres
% using the haversine formula. Reference:
% http://en.wikipedia.org/wiki/Haversine_formula
%%%%%%%%%%%%%%%%%%%%%%%% This is currently called in the PARSE.m script %%%%%%%%%%%%%%%%%

% convert lat/lon data into radians, calculate incremental elevation changes
lonrad = lon * pi() / 180;
latrad = lat * pi() / 180;

d_ele = zeros(numel(index), 1);
d_lon = zeros(numel(index), 1);
d_lat = zeros(numel(index),1);
h = zeros(numel(index),1);
dist_crow = zeros(numel(index),1);
dist = zeros(numel(index),1);
dist_cumul = zeros(numel(index),1);
vel = zeros(numel(index),1);

% the following for loop calculates incremental real distances travelled, cumulative distance travelled and velocity.

% Max speed for 930E's is 40 MPH/64 kph (source: http://www.komatsuamerica.com/trucks-930E-4SE)
speedlimit = 64

for i = 1:(numel(index)-1)
    % incremental elevation, longitudes, latitudes:
    d_ele(i+1) = ele(i+1) - ele(i);
d_lon(i+1) = lon(i+1) - lon(i);
d_lat(i+1) = lat(i+1) - lat(i);

    % haversine formula (d/R):
    h(i+1) = (sin((latrad(i+1)-latrad(i))/2))^2 + cos(latrad(i)) * cos(latrad(i+1)) * (sin((lonrad(i+1)-lonrad(i))/2))^2;

    % distance travelled, as the crow flies, in metres:
    dist_crow(i+1) = 2 * 6371 * asin(sqrt(h(i+1))) * 1000;

    % real distance travelled, taking elevation change into account:
    dist(i+1) = sqrt((d_ele(i+1))^2 + (dist_crow(i+1))^2);

    % cumulative distance travelled, in metres:
    dist_cumul(i+1) = dist(i+1) + dist_cumul(i);

    % Test using a Haversine formula found online:
    % dist_crow(i+1) = haversinefun( [latrad(i+1) lonrad(i+1)] , [latrad(i) lonrad(i)] ) * 1000;

    % calculated velocity in km/h:
    vel(i+1) = dist(i+1) / 1000 * 3600;
end

C-5       LEGSSORT
% LEGSSORT.m – This program sorts and groups individual haul legs into folders.

clearvars -except runfolder
close all

haultimesfolder = fullfile(runfolder,'HaulTimes');
legfolder = fullfile(runfolder,'HaulTimes','_raw','LEGS');
sortedfolder = fullfile(haultimesfolder,'_sorted');

% find all files and list
files = getAllFiles(legfolder);
allfiles = length(files);

% split full file paths into paths, filenames, and extensions for sorting
for i = 1 : allfiles
    [pathstr, name, ext] = fileparts(char(files(i)));
    paths(i,:) = {pathstr};
    filenamestemp(i,:) = {name};
    exts(i,:) = {ext};
end

filenames = char(filenamestemp);
legs = filenames(:,1:6);
legs1 = [legs,filenames(:,14:16)];

% This ensures all leg names have the same length in terms of characters to
% allow for easy manipulation/ID later on.
for a = 1 : length(legs1)
    if legs1(a,7) == '
        legs1(a,7:9) = '___';
    end
end

disp('');
disp('Sorting haul leg files into folders...');

% Find unique legs (origin-destination pairs).  Au = legfolders
% (i.e. 01to02, etc etc)
[Au I J] = unique(legs1,'rows');
Count = accumarray(J,1);
tic
for b = 1 : length(Au)
    mkdir(sortedfolder,Au(b,:));
end
for c = 1 : length(Au)
    cntr = 0;
    for d = 1 : all files
        if strcmp(legs1(d,:),Au(c,:)) == 1
            cntr = cntr + 1;
            cntrfile = sprintf('%02d.mat',cntr);
            sortpath = fullfile(sortedfolder,Au(c,:),cntrfile);
            copyfile(char(files(d)),sortpath);
        end
    end
end
toc
cd(haultimesfolder);

C-6 LEGANALYSIS

% LEGANALYSIS.m – A program for generating time and/or speed vs.
cumulative distance along the haul route plots

clearvars -except runfolder
close all
OF = cd;

% define parent folders
haultimesfolder = fullfile(runfolder,'HaulTimes');
legfolder = fullfile(runfolder,'HaulTimes', '_raw','LEGS');
sortedfolder = fullfile(haultimesfolder, '_sorted');

% choose desired folder to analyze
reply = input('Enter leg to be analyzed (ie. 10to03 or 01to10ALT): ','s');
if length(reply) == 6
    reply = [reply,'___'];
end
folder = fullfile(runfolder,'HaulTimes', '_sorted',reply);

% uncomment if you want to choose suplot or not, otherwise sp = y so it
% always subplots

% sp = input('Subplot? [y/n]', 's');
sp = 'y';

% call histogram generation & speed filtering function. Only needs
% 'folder' as an input.
[filterhist, spfilterflag, spfilters, maxspeeds, loadlist] = histgen(folder);
% call velocity vs. distance function. This NEEDS TO RUN AFTER
histgen.m if PRELIMINARY FILTERS FOR OUTLYING HAULISTS AND HAULTIMES
are desired.
[haulspeedcell, spdvsdistfilt, ax99, xax99] = veldist(folder, loadlist);

% call cumul haul time vs. distance function for plotting;
[datehaultimes, indexhaultimes, elevscell, Y, ax100] = timedist(folder, loadlist, sp, xax99);

% subplot the raw speed/time vs. cumulative distance plots
if sp == 'Y' || sp == 'y'
    axcopy = subplot(211);
    copyobj(allchild(ax99),axcopy);
    set(gca,'yminortick','on','xminortick','on');
end

% get list of preliminarily filtered leg files from appropriate folder,
outputted by histgen.m function
OF = cd(folder);
allcount = length(loadlist);

% load variables for each haul leg from indiv_haul variable
for q = 1 : allcount
    load(loadlist(q).name);
    haultimes(q,1) = length(indiv_haul(:,7));
    hauldists(q,1) = sum(indiv_haul(:,9));
    maxspeeds(q,1) = max(indiv_haul(:,10));
    instspds(q,1) = {indiv_haul(:,10)};
    reltimes(q,1) = {indiv_haul(:,3)};
    reldists(q,1) = {indiv_haul(:,9)};
    elevs(q,1) = {indiv_haul(:,6)};
    hauldistcell(:,q) = {cumsum(indiv_haul(:,9))};
    hauldatecell(:,q) = {indiv_haul(:,11)};
end

meanhauldist = mean(hauldists);
stddevhauldist = std(hauldists);

% This next for loop is for counting intersection stops, if desired.
replyints = input('Count intersection stops [Y/N]? ','s');

if replyints == 'y' || replyints == 'Y'
    % define intersection for stop counting, if desired
    replystart = input('Intersection start: ');
    replyend = input('Intersection end: ');
    for A = 1 : allcount
        intcountarray = [cell2mat(hauldistcell(1,A))
        cell2mat(haulspeedcell(1,A)) cell2mat(hauldatecell(1,A))];
        intidx1 = find(intcountarray(:,1) > replystart);
intcountarray1 = intcountarray(intidx1(1):intidx1(end),:);
intidx2 = find(intcountarray1(:,1) < replyend);
intcountarray2 = intcountarray1(intidx2(1):intidx2(end),:);
intcounter = intcountarray2(:,2) < 1; % CHOOSE SPEED THRESHOLD

FOR STOP DEFINITION
    dateavg(A,:) = datestr(datevec(mean(intcountarray2(:,3)))); % date to do night shift/day shift analysis
    intcount(A,1) = sum(intcounter);
end

nstoptrucks = sum(intcount~=0);
percentstoptrucks = nstoptrucks/allcount*100;
intsstring = sprintf('%d of %d stopped',nstoptrucks,allcount);
figure(100);
text(replyend,5,intsstring);

% choice added for analysis using bins, sections, or both
reply1 = input('Split using [b]ins, [s]ections or b[o]th? ', 's');

% user chooses desired road segment bin sizes for velocity vs. distance
% along road plots

% for bins only
if reply1 == 'b'
    distbinsize = input('Enter desired bin size for road segments (m): ');
    distbins = 0:distbinsize:mean(hauldists);
    nBINS = length(distbins(1,:))-1;
    for b = 1 : nBINS
        BINname(b,:) = {num2str(b)};
    end
    BINends = distbins';
    BINends = BINends(BINends~=0);
end

% for sections only
elseif reply1 == 's'
    % Section definition.  Use speed plot to decide on split locations.
    nsections = input('Enter desired number of road sections: ');
    distbins = zeros(1, nsections + 1);
    for a = 1 : nsections
        fprintf('\nsection %d of %d - total haul distance: %0.0f\n', a, nsections, meanhauldist);
        sectionname(a,:) = input('Enter section descriptor
        sectionends(a,1) = input('Enter section end (m): ');
        if sectionname(a,:) == 'i' || sectionname(a,:) == 'I'
            sectionname(a,:) = char('Incline');
        elseif sectionname(a,:) == 'd' || sectionname(a,:) == 'D'
            sectionname(a,:) = char('Decline');
        elseif sectionname(a,:) == 'f' || sectionname(a,:) == 'F'
            sectionname(a,:) = char('Flat ');
        end
    end
end
elseif sectionname(a,:) == 'X' || sectionname(a,:) == 'x'
    sectionname1(a,:) = char('Xsection');
end
distbins(1,a+1) = sectionends(a,1);
BINends = distbins';
BINends = BINends(BINends~=0);
end

% for both bins and sections
elseif reply1 == 'o'
distbinsize = input('Enter desired bin size for road segments (m): ');
distbins = 0:distbinsize:mean(hauldists);
nBINS = length(distbins(1,:))-1;
for b = 1 : nBINS
    BINname(b,:) = {num2str(b)};
end
BINends = distbins';
BINends = BINends(BINends~=0);

% Section definition.  Use speed plot to decide on split locations.
nsections = input('Enter desired number of road sections: ');
for a = 1 : nsections
    fprintf('
Section %d of %d - total haul distance: %0.0f m
', ...
        a, nsections, meanhauldist);
    sectionname(a,:) = input('Enter section descriptor [I]ncline,[D]ecline,[F]lat,[X]intersection: ', 's');
    sectionends(a,1) = input('Enter section end (m): ');
    if sectionname(a,:) == 'i' || sectionname(a,:) == 'I'
        sectionname1(a,:) = char('Incline');
    elseif sectionname(a,:) == 'd' || sectionname(a,:) == 'D'
        sectionname1(a,:) = char('Decline');
    elseif sectionname(a,:) == 'f' || sectionname(a,:) == 'F'
        sectionname1(a,:) = char('Flat ');
    elseif sectionname(a,:) == 'X' || sectionname(a,:) == 'x'
        sectionname1(a,:) = char('Xsection');
    end
end

% find section lengths and store in array 'sectionlength'
sectionlength = zeros(nsections,1);
sectionlength(1,1) = sectionends(1,1);
for e = 1 : nsections-1
    sectionlength(e+1,1) = sectionends(e+1) - sectionends(e);
end

% find section lengths if using section only analysis
if reply1 == 's'
    for d = 1 : nsections
        sectionlength(d,1) = distbins(d+1) - distbins(d);
    end
end

% initialize matrix D to store unfiltered spd and dist data extracted from cell variables.
D = [];

for j = 1 : allcount
    C = [cell2mat(hauldistcell(j)) cell2mat(elevs(j))];
    D = [D; C];
end

% this hist function sorts the data into bins of user chosen size % (distbins) and returns the frequency (n) and bin ID (bin) for each data point.
[n1, bin] = histc(D(:,1), distbins);

for r = 1 : allcount
    % sort inst. speeds into bins. # points per bin = # haul legs
cumdists(r,1) = {cumsum(cell2mat(reldists(r,1)))};
cumdisttemp = cell2mat(cumdists(r,1)); % cumulative distances temp variable
    [n, xout] = hist(cumdisttemp, distbins); % sort cumul. dists. into bins and index
    bintimes(:,r) = n; % the count of points per bin, n, is the time spent in each bin for each haul.

    % interpolate cumulative travel time at bin intervals. Use Yi as an input for plots, etc. if this method is desired
    x = [0; cell2mat(Y(1,r))];

    x0 = (diff(x) == 0); % this is a bug fix for if the truck is at a dead stop and has two identical distances travelled between time increments.
    ix = find(x0 == 1); % find the trouble point, return index, and make it a very small number (i.e. a few centimetres) so interpolation works.
    x(ix+1) = x(ix) + 0.0005; % adding 0.0005 metres - negligible, but fixes the problem discussed above.

    y = [0 cell2mat(Y(2,r))];
    yi(:,r) = interp1(x, y, distbins, 'linear', 'extrap'); % this is the interpolation function (linear)

    for u = 2 : length(yi(:,1))
Yitemp(u,r) = yi(u,r) - yi(u-1,r); % this converts the cumulative travel times to incremental travel times per bin.

% sort inst. speeds into bins. # points per bin = # haul legs
instspdstemp = cell2mat(instspds(r,1)); % instantaneous speeds temp variable
nspds = cumsum(n);
nspds = [1;nspds'];

for t = 1 : length(nspds)-1
    % [nspds, xoutspds] = hist(instspdstemp, distbins);
    if nspds(t) == 0
        nspds(t) = nspds(t-1); % when bin sizes are small, there is a risk of zero speeds being in the second bin
    end

    temp99 = instspdstemp(nspds(t):nspds(t+1)); % using nspds as index to isolate the speed data in each bin
    binspds(t,r) = mean(temp99); % take the mean of the bin speeds and record a single point for each haul for each bin.
    clearvars temp99
end

Yi = Yitemp';
Yi = Yi(:,2:end);
bintimesplot = bintimes';

% calc speeds by dividing bin size (m) by speed (km/h), making sure units are ok
if reply1 == 'S' || reply1 == 's'
    for l = 1 : nsections
        SPEEDSfromTIMES(:,l) = sectionlength(l,l)./Yi(:,l)*(3600/1000);
    end
else
    SPEEDSfromTIMES = distbinsize./Yi.*(3600/1000);
end

binspdsplot = SPEEDSfromTIMES;

% comment below if you want to use SPEEDSfromTIMES
% binspdsplot = binspds';

% binelevs are cell arrays containing elevation information for each bin
for z = 1 : length(distbins)-1
    for k = 1 : length(bin)
        if bin(k,1) == z
            elevstemp(k,z) = D(k,2);
binelevscols = elevtemp(:,z);
binelevs(z) = {binelevscols(binelevscols~=0)};

% ELEVS are mean bin elevations
ELEVS(z,1) = mean(cell2mat(binelevs(z)));

% extract and save the information from each bin/section for use later.
% binsectmaster is a cell array of matrices. 1st row contains the raw
% speed data in km/h for each bin/section inputted from VELVSDIST.m.
% 2nd row contains the quartile limits of the data for min (0%), Q1
% (25%), median (50%), Q3 (75%) , Q4 max.
binsectmaster = cell(1,length(BINends));

for w = 1 : length(BINends)

% Calculate quartile ranges & means for BIN TIMES, and store along
% with individual bin data in binsectmaster for use in vartune***.m
    temp = Yi(:,w);
    meantemp = {mean(temp)};
    datasettemp = dataset(temp);
    s(w,1) = summary(datasettemp);
    probstemp= [s(w).Variables.Data.Probabilities
               s(w).Variables.Data.Quantiles];

% Calculate quartile ranges & means for BIN SPEEDS, and store along
% with individual bin data in binsectmaster for use in vartune***.m
    tempspds = binspdsplot(:,w);
    meantempspds = {mean(tempspds)};
    datasettempspds = dataset(tempspds);
    s1(w,1) = summary(datasettempspds);
    probstempspds= [s1(w).Variables.Data.Probabilities
                    s1(w).Variables.Data.Quantiles];

% alternate manual input of desired quantile thresholds for BIN
% TIMES
    quant5 = quantile(temp,0.05);
    quant10 = quantile(temp,0.1);
    quant90 = quantile(temp,0.9);
    quant95 = quantile(temp,0.95);
    probstempl = [0.05 quant5; 0.1 quant(10); 0.9 quant(0.9) ; 0.95
                   quant(0.95)];

% alternate manual input of desired quantile thresholds for BIN
% SPEEDS
    quant5spds = quantile(tempspds,0.05);
    quant10spds = quantile(tempspds,0.1);
    quant90spds = quantile(tempspds,0.9);
quant95spds = quantile(tempspds,0.95);
probstemplspds = [0.05 quant5; 0.1 quant(10); 0.9 quant(0.9); 0.95 quant(0.95)];

% populate the cell array binsectmaster for use in vartune****.m
binsectmaster(1,w) = {temp};  % bin times
binsectmaster(2,w) = {probstem};  % quartile probabilities for times (Q1,Q2,Q3,Q4)
binsectmaster(3,w) = {meantemp};  % mean bin time across all hauls
binsectmaster(4,w) = {probstem1};  % additional quantile probabilities for BIN TIMES (top/bottom 10%)
binsectmaster(5,w) = {tempspds};  % bin speeds
binsectmaster(6,w) = {probstempspds};  % quartile probabilities for speeds (Q1,Q2,Q3,Q4)
binsectmaster(7,w) = {meantempspds};  % mean bin speed across all hauls
binsectmaster(8,w) = {probstemp1spds};  % additional quantile probabilities for BIN SPEEDS (top/bottom 10%)

end

% Code for plotting true instantaneous speeds calculated from raw data
binspeedsplot = binspdsplot;

% Boxplot and formatting for SPEEDS
% Create boxplot
hold all;
figure(3);
if sp == 'Y' || sp == 'y'
    subpboxplot = subplot(2,1,1);
    close 99 % close figure 99 which was generated and copied over to a subplot
else subpboxplot = gca;
end

boxplot(binspeedsplot, BINends, 'plotstyle','compact','whisker',100);
ax1 = gca;
xboxlim = XLim;
if reply1 == 'b' || reply1 == 'o'
    if distbinsize <= 50
        binlabels1 = BINends(1:4:end)';
        set(subpboxplot,'XTick', 1:4:nBINS,'XTickLabel', binlabels1,'XMinorTick', 'on');
    end
end

% Hide outliers (simplifies plot)
% h = findobj(gca,'tag','Outliers');
% set(h,'Visible','off');
% Labels and Titles
xlabel('Bins of incremental distance along haul route (m)');
ylabel('Distributions of instantaneous speeds per bin (km/h)');
title1 = sprintf('Box Plot distributions of instantaneous speeds over
incremental distances of Haul Route\nROUTE: %s --- HAUL ROUTE
DISTANCE: %0.0f m>>>', reply, mean(hauldists));
title(title1);

% Plot elevations
ax2 = axes('Position',get(ax1,'Position'),'XTicklabel',[],'XLim',xboxlim,'YAxisLocation','right','XAxisLocation','top','visible','off','Color','none ','XColor','k','YColor','k');
hold all;
elevsplot = plot(0.5:numel(BINends), ELEVS,'color','r','linewidth',1);

% text(distbins(1:length(elevs)), elevs,{elevs},'horizontalalignment','center','color','r','fontweight','bold');
ylabel(ax2,'Elevation (m)');

% use the variable removeint for intersection removal analysis
removeint = 0;
if removeint == 1
    rintstart = zeros(1,2);
    rintend = zeros(1,2);
    rintstart(1,2) = input('Enter starting bin (in metres along route):
');
    rintend(1,2) = input('Enter ending bin (in metres along route):
');
    rintstart(1,1) = find(BINends==rintstart(1,2));
    rintend(1,1) = find(BINends==rintend(1,2));
    rintrange = rintend(1,1)-rintstart(1,1);
    prebinsectmaster = binsectmaster;

    for zz = rintstart(1,1) : rintend(1,1)
        Yi(:,zz) = Yi(:,rintstart(1,1)-1);
        binsectmaster(1,zz) = {Yi(:,zz)};
    end
    sp = 'n';
end

% Boxplot and formatting for TIMES
% Create boxplot
hold all;

if sp == 'Y' || sp == 'y'
    subpboxplot1 = subplot(2,1,2);
else
    figure(4);
subpboxplot1 = axes;
end

boxplot(Yi, BINends, 'plotstyle','compact','whisker',100);
ax1 = gca;

if reply1 == 'b' || reply1 == 'o'
    if distbinsize <= 50
        binlabels2 = BINends(1:4:end);
        set(subpboxplot1,'XTick', 1:4:nBINS,'XTickLabel',
            binlabels2,'XMinorTick', 'on');
    end
end

% Hide outliers (simplifies plot)
% h = findobj(gca,'tag','Outliers');
% set(h,'Visible','off');

% Labels and Titles
xlabel('Bins of incremental distance along haul route (m)');
ylabel('incremental travel times times per bin (seconds)');
title1 = sprintf('Box Plot distributions of incremental times over
incremental distances of Haul Route
--- HAUL ROUTE DISTANCE: %0.0f m>>>', reply, mean(hauldists));
title(title1);

% Plot elevations
ax2 = axes('Position',get(ax1,'Position'),'XTicklabel',[],'XLim',xboxlim,'YAxisLocation','right','Xaxislocation','top','visible','off','Color','none', 'XColor','k','YColor','k');
hold all;
figure(3);
elevsplot = plot(0.5:numel(BINends), ELEVS,'color','r','linewidth',1);

% text(distbins(1:length(elevs)), elevs, elevs,'horizontalalignment','center','color','r','fontweight','bold')
ylabel(ax2,'Elevation (m)');

C-7 HISTGEN & VELDIST (functions called in LEGANALYSIS)

function [haulspeedcell, spdvsdistfilt, ax99, xax99] = veldist(folder, loadlist)
%this function inputs desired speed filter thresholds, and outputs a cell array containing the appropriate haultimes for each filter, ready to be plotted. Also performs median filtering and stores this data separately.
get list of preliminarily filtered leg files from appropriate folder, 
outputted by histgen.m function
OF = cd(folder);
allcount = length(loadlist);

% generate speed vs. distance along the route plot
figure(99);
ax99 = axes('Parent',99);

% choose desired level for median filtering:
% medfilt = input('Choose level for median filtering: ');
medfilt = 3;

% initialize spdvsdistfilt for use as a storage variable of median
% filtered data
spdvsdistfilt = [];

% initialize temp variables using NaN's since each column will have
% a different length, filter out NaN's after populating these
% variables.
haulspeedtemp = NaN(10000,200);
elevstemp = NaN(10000,200);

for q = 1 : allcount
    load(loadlist(q).name);
    haultimes(q,1) = numel(indiv_haul(:,7));
    hauldists(q,1) = sum(indiv_haul(:,9));
    maxspeeds(q,1) = max(indiv_haul(:,10));
    hauldistcell(:,q) = {cumsum(indiv_haul(:,9))};
    for r = 1 : numel(indiv_haul(:,1))
        haulspeedtemp(r,q) = indiv_haul(r,10);
        elevstemp(r,q) = indiv_haul(r,6);
    end
    x = cell2mat(hauldistcell(:,q));
    y = haulspeedtemp(:,q);
    y = y(isnan(y)==0);
    haulspeedcell(:,q) = {y};
    z = elevstemp(:,q);
    z = z(isnan(z)==0);
    elevscell(:,q) = {z};

    % perform median filtering
    xl = medfilt1(x,medfilt);
yl = medfilt1(y,medfilt);

    hold all
    plot(xl,yl);
xlabel('Distance along route (m)');
ylabel('Instantaneous speed (km/h)');
titledisp = sprintf('Speed vs. Cumulative Distance Travelled\nMEDIAN FILTER WINDOW: %d data points',medfilt);
title(titledisp);
set(gca,'yminortick','on','xminortick','on');

temp = [x1 y1];
temp1 = [spdvsdistfilt; temp];
spdvsdistfilt = sortrows(temp1,1);

end

xax99 = XLim;

% Return to original working folder
cd(OF);
end

function [filterhist, spfilterflag, spfilters, maxspeeds, loadlist] = histgen(folder)

%this function inputs desired speed filter thresholds, and outputs a cell array containing the appropriate haultimes for each filter, ready to be plotted.

% get list of leg files from appropriate folder
OF = cd(folder);
loadlist = dir('*\*.mat');
allcount = length(loadlist);

% filter out wandering trucks and outliers
for q = 1 : allcount
    load(loadlist(q).name);
    haultimes(q,1) = length(indiv_haul(:,7));
    hauldists(q,1) = sum(indiv_haul(:,9));
    maxspeeds(q,1) = max(indiv_haul(:,10));
end

meanhauldist = mean(hauldists);
stddevhauldist = std(hauldists);

strdist = sprintf('\n\nMean haul distance: %0.2f m\nStd. Dev.: %0.2f',meanhauldist, stddevhauldist);

170
disp(strdist);

% enter filter sensitivities to be used in histfilter function
% distinput = input('Enter desired sensitivity for distance filter in #''s of SD''s: ');
distinput = 2;

% distrange = input('
Input desired haul distance range for filtering
out wandering trucks\n(domain = [mean-range, mean+range]): ');
distrange = distinput * stddevhauldist;
hauldistmax = meanhauldist + distrange;
hauldistmin = meanhauldist - distrange;

haulfilter = numel(haultimes);
distancefiltercount = 0;

% Filter our OUTLIERS, i.e. any data point greater than 3 standard
% deviations from the haultime mean
outliermean = mean(haultimes);
outlierstd = std(haultimes);

strdist = sprintf('
Mean haul time:  %0.2f m
Std. Dev.: %0.2f', outliermean, outlierstd);
disp(strdist);

% timeinput = input('Enter desired sensitivity for time filter in #''s
of SD''s: ');
timeinput = 5;

outlierlimit = outliermean + timeinput * outlierstd;
outliercount = 0;
overspeedcount = 0;

for a = 1 : haulfilter
    if hauldists(a,1) < hauldistmin || hauldists(a,1) > hauldistmax
        hauldists(a,1) = -99;
distancefiltercount = distancefiltercount + 1;
    elseif haultimes(a,1) > outlierlimit
        haultimes(a,1) = -88;
        outliercount = outliercount + 1;
    elseif maxspeeds(a,1) > 65
        maxspeeds(a,1) = -77;
        overspeedcount = overspeedcount + 1;
    end
end

% remove hauls with discrepancies in HAULDIST:
removal = any(hauldists == -99,2);
hauldists(removal,:) = [];
loadlist(removal,:) = [];
% remove hauls with OUTLIERS:
remove2 = any(haultimes == -88,2);
haultimes(remove2,:) = [];
loadlist(remove2,:) = [];

% remove hauls with speeds greater than maximum possible speed (GPS error):
remove3 = any(maxspeeds == -77,2);
maxspeeds(remove3,:) = [];
loadlist(remove3,:) = [];

% Display filtering results
dispfilter = sprintf('n%d total hauls to be analyzed; n%d haultimes were filtered as OUTLIERS, > %d SD's from mean haultime (haultimes > %0.0f seconds); n%d haultimes were filtered due to HAULDIST, > %d SD's from mean distance (hauldist > %0.0f metres); n%d haultimes were filtered due to MAXSPEEDS, > 64 km/h. n%d remain for stat analysis. n', haulfilter, outliercount, timeinput, outlierlimit, timeinput, distancefiltercount, distinput, hauldistmax, overspeedcount, length(loadlist(:,1)));
disp(dispfilter);

% input number of non-zero speed filters
filtnum = input('Input number of desired speed filters (enter 0 for 1 histogram with all data): ');

if filtnum ~= 0
    for l = 1 : filtnum
        disp1 = sprintf('Filter %d of %d', l, filtnum);
disp(disp1);
        spfilters(l,1) = input('Enter filter speed in km/h: ');
    end

    % insert a speed filter = 0 to get ALL data points for histogram plot
    spfilters(end+1,1) = 0;
    spfilters = sort(spfilters);
    filtnum = filtnum + 1;
else
    filtnum = 1;
    spfilters(1,1) = 0;
end

% initialize filterhist as a cell array, and a flag variable so that % individual legs can be identified later
filterhist = cell(1,filtnum);
spfilterflag = zeros(length(loadlist),filtnum);

% filter leg data according to inputted spfilters
for n = 1 : filtnum

    filtertemp = [];

    for m = 1 : length(loadlist)

        load(loadlist(m).name);

        if all(indiv_haul(:,10) > spfilters(n,1))
            ht = etime(datevec(indiv_haul(end,7)),datevec(indiv_haul(1,7)));
            hd = sum(indiv_haul(:,9));
            delev = indiv_haul(end,6) - indiv_haul(1,6);
            avgvel = mean(indiv_haul(:,10));
            leginfo = [ht hd delev avgvel];
            filtertemp = [filtertemp ; leginfo];
            spfilterflag(m,n) = m;
        end
    end

    filterhist(n) = {filtertemp};

end

% return to original working folder
cd(OF);
end

C-8

VELVSDIST

% VELVSDIST.m - a program for plotting instantaneous speeds vs. distance along a given route

clearvars -except runfolder
close all
% remember working directory, functions are used herein...
OF = cd;

% define parent folders
haultimesfolder = fullfile(runfolder,'HaulTimes');
legfolder = fullfile(runfolder,'HaulTimes','_raw','LEGS');
sortedfolder = fullfile(haultimesfolder,'_sorted');

% choose desired folder to analyze
reply = input('Enter leg to be analyzed (ie. 10to03 or 01to10ALT): ', 's');
if length(reply) == 6
    reply = [reply,'___'];
end
folder = fullfile(runfolder,'HaulTimes','_sorted',reply);
% call histogram generation & speed filtering function. Only needs
% 'folder' as an input.
[filterhist, spfilterflag, spfilters, maxspeeds, loadlist] =
histgen(folder);

% call velocity vs. distance function. This NEEDS TO RUN AFTER
histgen.m, if PRELIMINARY FILTERS FOR OUTLYING HaulDISTs AND HaulTIMES
are desired.
[hauldistcell, haulspeedcell, haultimes, hauldists, elevscell,
allcount, spdvsdistfilt, x, y] = veldist(folder, loadlist);

% choice added for analysis using bins, sections, or both
reply1 = input('Split using [b]ins, [s]ections or b[o]th? ', 's');

% user chooses desired road segment bin sizes for velocity vs. distance
% along road plots

% for bins only
if reply1 == 'b'
    distbinsize = input('Enter desired bin size for road segments (m): ');
    distbins = 0:distbinsize:mean(hauldists);
    nBINS = length(distbins(1,:))-1;
    for b = 1 : nBINS
        BINname(b,:) = {num2str(b)};
    end
    BINends = distbins';
    BINends = BINends(BINends~=0);

    % for sections only
    elseif reply1 == 's'
        % Section definition. Use speed plot to decide on split locations.
        nsections = input('Enter desired number of road sections: ');
        distbins = zeros(1, nsections + 1);
        for a = 1 : nsections
            fprintf('
\section %d of %d\n', a, nsections);
            sectionname(a,:) = input('Enter section descriptor [I]ncline,[D]ecline,[F]lat,[X]intersection: ', 's');
            sectionends(a,1) = input('Enter section end (m): ');
            distbins(1,a+1) = sectionends(a,1);
        end

        % for both bins and sections only
    elseif reply1 == 'o'
        distbinsize = input('Enter desired bin size for road segments (m): ');
        distbins = 0:distbinsize:mean(hauldists);
nBINS = length(distbins(1,:))-1;
for b = 1 : nBINS
    BINname(b,:) = {num2str(b)};
end
BINends = distbins';
BINends = BINends(BINends~=0);

% Section definition. Use speed plot to decide on split locations.
nsections = input('Enter desired number of road sections: ');

for a = 1 : nsections
    fprintf('
section %d of %d
', a , nsections);
    sectionname(a,:) = input('Enter section descriptor
    sectionends(a,1) = input('Enter section end (m): ');
    if sectionname(a,:) == 'i' || sectionname(a,:) == 'I'
        sectionname1(a,:) = char('Incline');
    elseif sectionname(a,:) == 'd' || sectionname(a,:) == 'D'
        sectionname1(a,:) = char('Decline');
    elseif sectionname(a,:) == 'f' || sectionname(a,:) == 'F'
        sectionname1(a,:) = char('Flat   ');
    elseif sectionname(a,:) == 'X' || sectionname(a,:) == 'x'
        sectionname1(a,:) = char('Xsection');
    end
end

% find section lengths and store in array 'sectionlength'
sectionlength = zeros(nsections,1);
sectionlength(1,1) = sectionends(1,1);
for e = 1 : nsections-1
    sectionlength(e+1,1) = sectionends(e+1) - sectionends(e);
end

% find section lengths if using section only analysis
if reply1 == 's'
    for d = 1 : nsections
        sectionlength(d,1) = distbins(d+1) - distbins(d);
    end
end

% initialize matrix D to store unfililtered spd and dist data extracted from
% cell variables.
D = [];
for j = 1 : allcount
C = [cell2mat(hauldistcell(j)) cell2mat(haulspeedcell(j))
    cell2mat(elevscell(j))];
D = [D; C];
end

% this histc function sorts the data into bins of user chosen size
% (distbins) and returns the frequency (n) and bin ID (bin) for each
data point.
[n,bin] = histc(D(:,1), distbins);

% binspds and binelevs are cell arrays containing information for each
bin
for z = 1 : length(distbins)-1
    for k = 1 : length(bin)
        if bin(k,1) == z
            binspds(temp(k),z) = D(k,2);
            elevs(temp(k),z) = D(k,3);
        end
    end
endspdscols = binspdscol(:,z);
binspds(z) = {binspdscols(binspdscols~=0)};

elevscols = elevs(:,z);
binelevs(z) = {binelevscols(binelevscols~=0)};
end

% initialize the variable used for boxplot- since there are a different
% number of data points in each bin, it needs to be initialized with
NaN values.
bartemp = NaN(max(n), length(distbins)-1);

% This plots the data as a boxplot in a separate figure.
for y = 1 : length(distbins)-1
    bartemp(1:length(cell2mat(binspds(y))),y) = cell2mat(binspds(y));
elevs(y,1) = mean(cell2mat(binelevs(y)));

    % bt = {bartemp(:,y)};
    % remove = isnan(cell2mat(bt));
    % bt(remove,y) = [];
    %
    % meanspdtemp(1,y) = mean(bt);
end

% extract and save the information from each bin/section for use later.
% binsectmaster is a cell array of matrices. 1st row contains the raw
% speed data in km/h for each bin/section inputted from VELVSDIST.m.
2nd row contains the quartile limits of the data for min (0%), Q1
(25%), median (50%), Q3 (75%) , Q4 max.

binsectmaster = cell(1,length(bartemp(1,:)));
for w = 1 : length(bartemp(1,:))
    temp = bartemp(:,w);
    temp(isnan(temp(:,1)),:) = [];

    meantemp = {mean(temp)};
    datasettemp = dataset(temp);
    s(w,1) = summary(datasettemp);
    binsectmaster(1,w) = {temp};

    probstemp= [s(w).Variables.Data.Probabilities
                s(w).Variables.Data.Quantiles];

    binsectmaster(2,w) = {probstemp};
    binsectmaster(3,w) = meantemp;
end

% Boxplot and formatting
% this next loop labels the x-axis with bin ranges in seconds
for j = 1 : numel(distbins)-1
    binlabel(j,:) = {sprintf('%0.0f-%0.0f',distbins(j),distbins(j+1))};
end

% Create boxplot
hold all;
figure(3);
boxplot(bartemp, binlabel, 'plotstyle','compact','whisker',4);
ax1 = gca;
set(ax1,'XColor','b','YColor','b');

% Labels and Titles
xlabel('Bins of incremental distance along haul route (m)');
ylabel('instantaneous speed distributions per bin (km/h)');
title1 = sprintf('Box Plot distributions of speeds over incremental
distances of Haul Route\n<<ROUTE: %s --- HAUL ROUTE DISTANCE: %0.0f m>>\n', reply, mean(hauldists));
title(title1);

% Plot elevations
if reply1 == 'b' || reply1 == 'o'
    xlim = [0 BINends(end)];
else
    xlim = distbins';
    xlim = [0 xlim(end)];
end
ax2 = axes('Position',get(ax1,'Position'), 'XLim', xlim, 'YAxisLocation','right','YAxisLocation','top','visible','off','Color','none', 'XColor','k','YColor','k');
hold all;
figure(3);
C-9 VARTUNETIMES

% VARTUNETIMES.m - a script used to input TIME data from sections/bins of
% haul routes, and incrementally decrease variability for random
% number generation.  This program is meant to be run immediately after
% VELVSDIST.m as it uses variables outputted from that program.

% if statement to determine what type of analysis was specified in
% VELVSDIST.m
if reply1 == 'b'
    clearvars -except runfolder binsectmaster sectionname sectionnamel
    sectionends sectionlength nsections binlabel reply reply1 nBINS BINends
    BINname distbinsize removeint Y1
    tunereply = input('Remove [W]hiskers (Q1 & Q4), [L]ower 25% (Q1),
    [U]pper 25% (Q4), [M]iddle 50%? (Q2 & Q3), [S]peeders (>40 km/h),
    Sto[P]pers, or [N]one? ', 's');
    counter = nBINS;
    dividelength = repmat(distbinsize,nBINS,1);
elseif reply1 == 's'
    clearvars -except runfolder binsectmaster sectionname sectionnamel
    sectionends sectionlength nsections binlabel reply reply1 removeint Y1
    counter = nsections;
    nBINS = nsections;
    BINends = sectionends;
    dividelength = sectionlength;
elseif reply1 == 'o'
    clearvars -except runfolder binsectmaster sectionname sectionnamel
    sectionends sectionlength nsections binlabel reply reply1 nBINS BINends
    BINname distbinsize removeint Y1
    tunereply = input('Remove [W]hiskers (Q1 & Q4), [L]ower 25% (Q1),
    [U]pper 25% (Q4), [M]iddle 50%? (Q2 & Q3),\n[S]peeders (>40 km/h),
    Sto[P]pers, or [N]one? ', 's');
    counter = nBINS;
    dividelength = repmat(distbinsize,nBINS,1);
end

spdcntr = 0;
stopcntr = 0;
for i = 1 : counter
    clearvars temp1 temp2 temp3 temp4 temp5
    % This is cut&pasted from LEGANALYSIS.m- choose your
desired inputs
    % binsectmaster(1,w) = {temp};  % bin times
binsectmaster(2,w) = {probstemp}; % quartile probabilities for times (Q1,Q2,Q3,Q4)
binsectmaster(3,w) = meantemp; % mean bin time across all hauls
binsectmaster(4,w) = {probstemp1}; % additional quantile probabilities for BIN TIMES (top/bottom 10%)
binsectmaster(5,w) = {tempspds}; % bin speeds
binsectmaster(6,w) = {probstempspds}; % quartile probabilities for speeds (Q1,Q2,Q3,Q4)
binsectmaster(7,w) = meantempspds; % mean bin speed across all hauls
binsectmaster(8,w) = {probstemp1spds}; % additional quantile probabilities for BIN SPEEDS (top/bottom 10%)

currentsection = i;
temp1 = cell2mat(binsectmaster(1,i)); % assign the desired time or speed data to temp

temp2 = cell2mat(binsectmaster(2,i)); % quartile probabilities - make sure they are consistent with the data you chose for temp1

temp3 = cell2mat(binsectmaster(3,i)); % bin means - make sure they are consistent with the data you chose for temp1

temp4 = cell2mat(binsectmaster(4,i)); % additional probabilities (top/bottom 10%) - make sure they are consistent with the data you chose for temp1

temp5 = cell2mat(binsectmaster(5,i)); % bins speeds

% For section analysis ONLY, 2 box and whisker plots are displayed for each section iteratively, one for before variability modification and one after.
if reply1 == 's'|| reply1 == 'S'
savename(i,:) = {sprintf('S%d_%s_%d',i, sectionname(i,1), sectionends(i,1))};
titletemp = sprintf('Section %d of %d --- Count = %d
Section Type: %s --- Ending at: %d m', i, nsections, numel(temp1), sectionname(i,1), sectionends(i,1));
else
savename(i,:) = {sprintf('S%d_%s_%d',i, cell2mat(BINname(i,1)), BINends(i,1))};
titletemp = sprintf('Section %d of %d --- Count = %d
Section Type: %s --- Ending at: %d m', i, nBINS, numel(temp1), cell2mat(BINname(i,1)), BINends(i,1));
end
disp(titletemp);
disp(' ');

if reply1 == 's' || reply1 == 'S'
figure(i);
fig1 = subplot(1,2,1);
hold on;
boxplot(temp1,sectionends(i,1),'whisker',4);
sectylim = YLim;
title(titletemp);
str1 = sprintf('Median: %0.0f s', temp2(3,2));
str2 = sprintf('Q1: %0.0f s', temp2(2,2));
str3 = sprintf('Q3: %0.0f s', temp2(4,2));
str4 = sprintf('Max: %0.0f s', temp2(5,2));
str5 = sprintf('Min: %0.0f s', temp2(1,2));
str6 = sprintf('Mean: %0.0f s', temp3);
str7 = sprintf('StdDev: %0.0f s', std(temp1));
text(1.1,temp2(3,2),str1,'Fontsize',8);
text(1.1,temp2(2,2),str2,'Fontsize',8);
text(1.1,temp2(4,2),str3,'Fontsize',8);
text(1.1,temp2(5,2),str4,'Fontsize',8);
text(1.1,temp2(1,2),str5,'Fontsize',8);
plot(1,temp3,'*','markersize',10);
text(0.6,temp3,str6,'Fontsize',8);
text(0.6,temp2(2,2),str7,'Fontsize',8);
end

% Choose desired variability to exclude
if reply1 == 's' || reply1 == 'S'
clearvars tunereply desc temp10 titletemp1;
tunereply = input('Remove [W]hiskers (Q1 & Q4), [L]ower 25% (Q1), [U]pper 25% (Q4), [M]iddle 50% (Q2 & Q3), [S]peeders (>40 km/h), Sto[P]pers, or [N]one? ', 's');
end

% Remove variability based on user input 'tunereply' and Quartile limits defined in 'temp2'
if tunereply == 'W'
desc = 'Q1 & Q4';
    temp10 = temp1(temp1>temp2(2,2));
    temp10 = temp10(temp10<temp2(4,2));
elseif tunereply == 'L'
desc = 'Q1';
    temp10 = temp1(temp1>temp2(2,2));
elseif tunereply == 'U'
desc = 'Q4';
    temp10 = temp1(temp1<temp2(2,2));
elseif tunereply == 'M'
desc = 'Q2 & Q3';
    temp10 = temp1(temp1<temp2(2,2));
    temp100 = temp1(temp1>temp2(4,2));
    temp10 = [temp10;temp100];
elseif tunereply == 'T'
desc = 'Top/Bottom 10%';
    temp10 = temp1(temp1>temp4(2,2));
    temp10 = temp10(temp10<temp4(4,2));
elseif tunereply == 'T'
desc = 'Top/Bottom 10%';
    temp10 = temp1(temp1>temp4(2,2));
end
elseif tunerreply == 'S'
    temp10 = temp1; \% copy times to temp10, filter array. Below is
code to find speeders and fix at max speed limit
    desc = 'Speeders (>40 km/h)';
    speeders(1,i) = { find(temp1< 45/(40 *1000/3600)) }; \% find all
avg. bin speeds greater than speed limit and flag
    speeders(2,i) = { temp5(cell2mat(speeders(1,i))) };
    speeders(3,i) = { temp1(cell2mat(speeders(1,i))) };
    nspeeders = numel(cell2mat(speeders(1,i))); \% Count number of
speeders per bin and display
    if nspeeders > 0
        spdcntr = spdcntr+1;
        binspeeders(spdcntr,1) = i; \% bin containing speeders
        binspeeders(spdcntr,2) = BINends(i); \% location of speeder
        containing bin
        binspeeders(spdcntr,3) = nspeeders; \% number of speeders in
this bin
        binspeeders(spdcntr,4) = numel(BINends);
        binspeeders(spdcntr,5) = nspeeders/numel(BINends); \%
percentage of speeding trucks in bin
        \% temp10 = temp1(legalspeeds); \% this simply
removes all times associated with speeding
        \% trucks
        binspeeders(spdcntr,6) =
        mean(temp10(cell2mat(speeders(1,i))));
        temp10(cell2mat(speeders(1,i))) = 45/(40*1000/3600); \% set
all speeders to travel at maximum in-pit speed limit
        binspeeders(spdcntr,7) = temp10(i,1);
    end
elseif tunerreply == 'P'
    temp10 = temp1; \% copy times to temp10, filter array. Below is
code to find stoppers and fix at median bin time
    desc = 'Stoppers (<5 km/h)';
    stoppers(1,i) = { find(temp1 > 45/( 8 *1000/3600)) }; \% find
all avg. bin speeds lower than x km/hr and flag
    stoppers(2,i) = { temp5(cell2mat(stoppers(1,i))) };
    stoppers(3,i) = { temp1(cell2mat(stoppers(1,i))) };
    nstoppers = numel(cell2mat(stoppers(1,i))); \% Count number of
stoppers per bin and display
    if nstoppers > 0
        stopcntr = stopcntr+1;
        binstoppers(stopcntr,1) = i; \% bin containing stoppers
        binstoppers(stopcntr,2) = BINends(i); \% location of stopper
        containing bin
        binstoppers(stopcntr,3) = nstoppers; \% number of stoppers
in this bin
        binstoppers(stopcntr,4) = numel(BINends);
        binstoppers(stopcntr,5) = nstoppers/numel(BINends); \%
percentage of stopping trucks in bin
        \% temp10 = temp1(legalspeeds); \% this simply
removes all times associated with stopping trucks
binstoppers(stopcntr,6) =
mean(temp10(cell2mat(stoppers(1,i))));  % mean before value to be
replaced

temp10(cell2mat(stoppers(1,i))) = mean(temp1);  % set all
stoppers to travel at median bin speed (time taken to travel the bin
distance at that speed)

binstoppers(stopcntr,7) = temp10(i,1);  % after value

else if tunereply == 'N'
    desc = 'None';
    temp10 = temp1;
else if tunereply == 'R'
    desc = 'Remove Intersection';
    inttravelspeed = input('Select desired speed for travel across
intersection: ');
    temp10 = temp1;
    counter
    temp10(:, :) = sectionlength(i,1)/(inttravelspeed * 1000 / 3600)
end

% Box and Whisker plot after variability excluded
if reply1 == 's'
    titletemp1 = sprintf('Section %d of %d --- %s REMOVED --- count
= %d', i, nsections, desc, numel(temp10));

    fig10 = subplot(1,2,2);
    hold on;
    boxplot(temp10,sectionends(i,1), 'whisker', 4);
    title(titletemp1);
    %    set(gca,'YLim',sectylim);
end

% calculate new stats & quartiles for 'tuned' data set, saved in
'temp10'
tMed = median(temp10);
tMean = mean(temp10);
tQ1 = quantile(temp10,0.25);
tQ3 = quantile(temp10,0.75);
tMin = min(temp10);
tMax = max(temp10);
tStddev = std(temp10);

% only plotted for section-only analysis
if reply1 == 's'
    str10 = sprintf('Median:  %0.0f s',tMed);
    str20 = sprintf('Q1:  %0.0f s', tQ1);
    str30 = sprintf('Q3:  %0.0f s', tQ3);
    str40 = sprintf('Max:  %0.0f s', tMax);
    str50 = sprintf('Min:  %0.0f s', tMin);
    str60 = sprintf('Mean:  %0.0f s', tMean);
end
str70 = sprintf('StdDev: %0.0f s', tStddev);

text(1.1,tMed,str10,'FontSize',8);
text(1.1,tQ1,str20,'FontSize',8);
text(1.1,tQ3,str30,'FontSize',8);
text(1.1,tMax,str40,'FontSize',8);
text(1.1,tMin,str50,'FontSize',8);
plot(1,tMean,'*', 'MarkerSize', 10);
text(0.6,tMean,str60,'FontSize',8);
text(0.6,tQ1,str70,'FontSize',8);
end

aftermaster10(1,i) = {temp10};
end

% generate random speeds using distributions: 2 for each section, one
% before filtering and
% one after.

its = input('how many random haul times would you like to generate? ');

htbefore = zeros(its,1);
htafter = zeros(its,1);
timesbefore = zeros(counter,its);
timesafter = zeros(counter,its);
storetime1 = zeros(counter,1);
storetime10 = zeros(counter,1);
rndht1 = zeros(counter,1);
rndht10 = zeros(counter,1);

for p = 1 : its
    for q = 1 : counter
        % generate random speeds
        incrtime1 = randsample(cell2mat(binsectmaster(1,q)),1);
        if isempty(cell2mat(aftermaster10(1,q))) == 1
            incrtime10 = cell2mat(binsectmaster(3,q));
        else
            incrtime10 = randsample(cell2mat(aftermaster10(1,q)),1);
        end

        storetime1(q,1) = incrtime1;
        storetime10(q,1) = incrtime10;
    end

timesbefore(:,p) = storetime1;
timesafter(:,p) = storetime10;
htbefore(p,1) = sum(storetime1);
htafter(p,1) = sum(storetime10);
clearvars incrtimel incrtimel0 storetimel storetimel0
end

beforemean = sprintf('
Mean HT BEFORE variability decrease: %0.0f
', mean(htbefore));
disp(beforemean);
aftermean = sprintf('Mean HT AFTER variability decrease: %0.0f
', mean(htafter));
disp(aftermean);
beforestdev = sprintf('Std. Dev. BEFORE variability decrease: %0.2f
', std(htbefore));
disp(beforestdev);
afterstdev = sprintf('Std. Dev. AFTER variability decrease: %0.2f
', std(htafter));
disp(afterstdev);

if reply1 == 's' || reply1 == 'o'
    % SECTIONS or BOTH ONLY
    if reply1 == 's'
        plotbysection = 'c';
    else
        plotbysection = input('Plot sections [i]ndividually, [c]omplete route, or [b]oth? ', 's');
    end

    % This splits up the speeds and ht's calculated previously into
    % sections for individual analysis

    for r = 1 : nsections
        cntr = 0;
        cumdivide = cumsum(dividelength);
        while cumdivide(cntr+1) <= sectionlength(r,1)
            cntr = cntr+1;
            indextemp(q,r) = cntr;
        end
        index = indextemp(indextemp~=0);
        csumindex = cumsum(index);
        indexends = csumindex;
        indexstarts = [1; csumindex+1];
        indexstarts(end,:) = [];
        if indexends(end) > nBINS
            indexends(end) = nBINS;
        end
        if reply1 == 'o'
            sectiontimesbefore(1,r) = {timesbefore(indexstarts(r):indexends(r),:)};
            sectiontimesafter(1,r) = {timesafter(indexstarts(r):indexends(r),:)};
            sectionbinhtsbefore(1,r) = {cell2mat(sectiontimesbefore(1,r))}
            sectionbinhtsafter(1,r) = {cell2mat(sectiontimesafter(1,r))};
```
sectionbinends(r,1) = {BINends(indexstarts(r):indexends(r),:)};
else
    sectiontimesbefore(1,r) = { timesbefore(r,:) };
    sectiontimesafter(1,r) = { timesafter(r,:) };
    sectionbinhtsbefore(1,r) = { cell2mat(sectiontimesbefore(1,r)) };
    sectionbinhtsafter(1,r) = { cell2mat(sectiontimesafter(1,r))};
    sectionbinends(r,1) = { BINends };
end
if plotbysection == 'i' || plotbysection == 'b'
    figure(r+10);
    subplot(211);
    plottemp1 = plot(cell2mat(sectionbinends(r,1)),cell2mat(sectiontimesbefore(1,r)), '.');
    scale = YLim;
    title10 = sprintf('Section %d - Incremental Times BEFORE removing %s
Mean Haul time: %0.0f' , r, desc, mean(sum(cell2mat(sectionbinhtsbefore(1,r)))); % Std dev. display was removed May23 2012
    title(title10);
    xlabel('Distance along haul route (m)');
    ylabel('Simulated bin times (s)');
    subplot(212);
    plottemp2 = plot(cell2mat(sectionbinends(r,1)),cell2mat(sectiontimesafter(1,r)), '.');
    title20 = sprintf('Section %d - Incremental Times AFTER removing %s
Mean Haul time: %0.0f' , r, desc, mean(sum(cell2mat(sectionbinhtsafter(1,r)))); % Std dev. display was removed May23 2012
    title(title20);
    xlabel('Distance along haul route (m)');
    ylabel('Simulated bin times (s)');
    set(subplot1,'YLim',scale); % ensures plots are all scaled similarly
end
end
elseif reply1 == 'b' % BINS ONLY
    sectionends = BINends;
end
if removeint == 1 % if removeint = 1, plot a special plot showing this info
    figure(20);
    intremoved = plot(sectionends,timesbefore, '.');
    title1 = sprintf('Route: %s
Mean Haul time: %0.0f',reply, mean(htbefore)); %
Std dev. display was removed May23 2012
    title(title1);
    scale10 = YLim;
    xlabel('Distance along haul route (m)');
    ylabel('Simulated bin times (s)');

    if distbinsize <= 50
        binlabels = BINends(1:4:end)';
        set(intremoved,'XTick', BINends(1:4:end),'XTickLabel',
        binlabels,'XMinorTick', 'on','XLim', [0 BINends(end)+distbinsize]);
    end
    else

    figure(10);
    subplot(211);
    plot(sectionends,timesbefore, '.')
    title1 = sprintf('Route: %s
Mean Haul time: %0.0f',reply, desc, mean(htbefore)); % Std dev.
    display was removed May23 2012
    title(title1);
    scale10 = YLim;
    xlabel('Distance along haul route (m)');
    ylabel('Simulated bin times (s)');

    if distbinsize <= 50
        binlabels = BINends(1:4:end)';
        set(subp1,'XTick', BINends(1:4:end),'XTickLabel',
        binlabels,'XMinorTick', 'on','XLim', [0 BINends(end)+distbinsize]);
    end

    subp = subplot(212);
    plot(sectionends,timesafter, '.')
    title1 = sprintf('Route: %s
Mean Haul time: %0.0f',reply, desc, mean(htafter)); % Std dev.
    display was removed May23 2012
    title(title1);
    xlabel('Distance along haul route (m)');
    ylabel('Simulated bin times (s)');
    set(subp,'YLim',scale10); % ensures plots are all scaled
    similarly

    if distbinsize <= 50
        binlabels = BINends(1:4:end)';

    if removeint == 1 % if removeint = 1, plot a special plot showing this info
    figure(20);
    intremoved = plot(sectionends,timesbefore, '.');
    title1 = sprintf('Route: %s
Mean Haul time: %0.0f',reply, mean(htbefore)); %
Std dev. display was removed May23 2012
    title(title1);
    scale10 = YLim;
    xlabel('Distance along haul route (m)');
    ylabel('Simulated bin times (s)');

    if distbinsize <= 50
        binlabels = BINends(1:4:end)';
        set(intremoved,'XTick', BINends(1:4:end),'XTickLabel',
        binlabels,'XMinorTick', 'on','XLim', [0 BINends(end)+distbinsize]);
    end
    else

    figure(10);
    subplot(211);
    plot(sectionends,timesbefore, '.')
    title1 = sprintf('Route: %s
Mean Haul time: %0.0f',reply, desc, mean(htbefore)); % Std dev.
    display was removed May23 2012
    title(title1);
    scale10 = YLim;
    xlabel('Distance along haul route (m)');
    ylabel('Simulated bin times (s)');

    if distbinsize <= 50
        binlabels = BINends(1:4:end)';
        set(subp1,'XTick', BINends(1:4:end),'XTickLabel',
        binlabels,'XMinorTick', 'on','XLim', [0 BINends(end)+distbinsize]);
    end

    subp = subplot(212);
    plot(sectionends,timesafter, '.')
    title1 = sprintf('Route: %s
Mean Haul time: %0.0f',reply, desc, mean(htafter)); % Std dev.
    display was removed May23 2012
    title(title1);
    xlabel('Distance along haul route (m)');
    ylabel('Simulated bin times (s)');
    set(subp,'YLim',scale10); % ensures plots are all scaled
    similarly

    if distbinsize <= 50
        binlabels = BINends(1:4:end)';
if replyl == 'o' || replyl == 's'
    if plotbysection == 'c' || plotbysection == 'b'

        % Plot lines marking the section splits on the plots and other
        info
        figure(30);
        subplot(211);
        if replyl == 's'
            for c = 1 : nsections
                if c == 1
                    plotdividelength = sectionlength(c,1)/2;
                else
                    plotdividelength(c,1) = sum(sectionlength((1:c-1),1)) + sectionlength(c,1)/2 ; % try to center the plots in each
                    section for section only...
                end
            end
        else
            plotdividelength = cumdividelength;
        end

        plot(plotdividelength,timesbefore, '.');
        title1 = sprintf('Route: %s - Incremental Times BEFORE removing
%s
Mean Haul time: %0.0f', reply, desc, mean(htbefore)); % Std dev.
        display was removed May23 2012
title(title1);
        xlabel('Distance along haul route (m)');
        ylabel('Simulated bin times (s)');
        set(gca,'XLim',[0 cumdividelength(end)+100]);
        xscale30 = XLim;
        yscale30 = YLim;

        for g = 1 : nsections
            yL = get(gca,'YLim');
            line([sectionends(g)],yL,'Color','r','linestyle','--');
            if replyl == 'o'
                meandisp = mean(sum(cell2mat(sectionbinhtsbefore(1,g))));
                stddisp = std(sum(cell2mat(sectionbinhtsbefore(1,g))));
            else
                meandisp = mean(cell2mat(sectionbinhtsbefore(1,g)));
                stddisp = std(cell2mat(sectionbinhtsbefore(1,g)));
            end
        end

    end
end
sectiondisp = sprintf('Section %d\n%snEnd: %d m\nMean: %0.0f s',g,sectionname1(g,:),sectionends(g),meandisp); % Std dev.
display was removed May23 2012
text(sectionends(g),max(max(timesbefore))-5,sectiondisp,'FontSize',8,'horizontalalignment','center');
end

hold on;
subp2 = subplot(212);
plot(plotdividelength,timesafter,'.');
title2 = sprintf('Route: %s - Incremental Times AFTER removing %s\nMean Haul time: %0.0f', reply, desc, mean(htafter)); % Std dev.
display was removed May23 2012
title(title2);
xlabel('Distance along haul route (m)');
ylabel('Simulated bin times (s)');
set(subp2,'XLim',xscale30,'YLim',yscale30); % ensures plots are all scaled similarly

for f = 1 : nsections
    sectionends(f),yL,'Color','r','linestyle','--');
    if reply1 == 'o'
        meandisp1 = mean(sum(cell2mat(sectionbinhtsafter(1,f))));
        stddisp1 = std(sum(cell2mat(sectionbinhtsafter(1,f))));
    else
        meandisp1 = mean(cell2mat(sectionbinhtsafter(1,f)));
        stddisp1 = std(cell2mat(sectionbinhtsafter(1,f)));
    end
    sectiondisp = sprintf('Section %d\n%snEnd: %d m\nMean: %0.0f s',f, sectionname1(f,:), sectionends(f), meandisp1);
    display was removed May23 2012
text(sectionends(f),max(max(timesafter))-5,sectiondisp,'FontSize',8,'horizontalalignment','center');
end
end

% information about speeders
if tunereply == 'S'
    disp('')
    disp('
BinSpeedersReport')
    format shortG
    binspeeders
elseif tunereply == 'P'
    disp('')
    disp('BinStoppersReport')
    format shortG
    binstoppers
end